

Editor Dana Simian

Proceedings of
The Fourth International
Students Conference on Informatics

ICDI

IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT

Romania, Sibiu, May 15-17, 2014

Editor DANA SIMIAN

IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT

Proceedings of International Students
Conference on Informatics
ICDD

May 15th – 17th, 2014
Sibiu, Romania

“Lucian Blaga” University of Sibiu

“Lucian Blaga” University, Sibiu, 2014

Editor Dana Simian

All papers in this volume were peer review by two independent reviewers

ISSN 2069-964X

Associated Editor Laura Florentina Stoica

Proceedings of International Students Conference on
Informatics ICDD

May 15th – 17th, 2014, Sibiu, Romania

Copyright @ 2014 All rights reserved to editors and authors

Preface

This volume contains refereed papers presented within the Fourth International Students Conference on Informatics - ICDD – 2014, which was held between May 15th – 17th, at the Faculty of Sciences, University ”Lucian Blaga” of Sibiu, Romania.

The conference is addressed to bachelor and master level students. Conference aim is to bring together students from different universities from all over the world to discuss and present their researches on informatics and related fields (like computational algebra, numerical calculus, bioinformatics, etc) and their original results. Specific topics of the conference included but were not restricted to: algorithms and data structures, graph theory and applications, formal languages and compilers, cryptography, modeling and simulation, computer programming, computer vision, computer graphics, game design, data mining, distributed computing, artificial intelligence, service oriented applications, networking, grid computing, mobile operating systems, scientific computing, software engineering, bioinformatics, robotics, computer architecture, evolutionary computing, multimedia systems, internet communication and technologies, web applications.

The talks were delivered by students from 7 countries (Bulgaria, Romania, Rusia, Serbia, Ukraine and United States).

We thank all the participants for their interesting talks and discussions. We also thank the members of the scientific committee for their help in reviewing the submitted papers and for their contributions to the scientific success of the conference and to the quality of this proceedings volume.

The conference was organized with the support of Romanian Ministry of National Education and of IQUEST, EBS, GSD, ROPARDO, TOP TECH and PanFood.

May 2014

Dana Simian
Conference chairman

Motto:

“There are no limits, only your imagination”

Proceedings of International Students Conference on Informatics, ICDD – 2014
IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT
Sibiu, Romania

Editor Prof. PhD. Dana Simian

Scientific committee

- Prof. PhD. Grigore Albeanu - University of Bucharest, Romania
- Prof. PhD. Florian Boian - "Babes-Bolyai" University of Cluj-Napoca, Romania
- Prof. PhD. Alina Barbulescu - Ovidius University of Constanta, Romania
- Prof. PhD. Oleksandr Dorokhov - Kharkiv National University of Economics, Ukraine
- Prof. PhD. Heiner Gonska - Duissburg-Essen University, Germany
- Prof. PhD. Gheorghe Grigoras - "Alexandru Ioan Cuza" University of Iasi, Romania
- Prof. PhD. Katalina Grigorova - University of Ruse, Bulgaria
- Prof. PhD. Ion Iancu - University of Craiova, Romania
- Prof. PhD. János Karsai - University of Szeged, Hungary
- Prof. PhD. Milena Lazarova - Technical University of Sofia, Bulgaria
- Prof. PhD. Daniela Marinescu - Transilvania University of Brasov, Romania
- Prof. PhD. Ioana Moisil - "Lucian Blaga University" of Sibiu, Romania
- Prof. PhD. Mariana Nagy - "Aurel Vlaicu" University of Arad, Romania
- Prof. PhD. Bazil Parv - "Babes-Bolyai" University of Cluj-Napoca, Romania
- Prof. PhD. Dana Petcu - West University of Timisoara, Romania
- Prof. PhD. Valer Rosca - "Lucian Blaga" University of Sibiu, Romania
- Prof. PhD. Ernest Scheiber - Transilvania University of Brasov, Romania
- Prof. PhD. Klaus Bruno Schebesch, "Vasile Goldis" University, Arad, Romania
- Prof. PhD. Dana Simian - "Lucian Blaga" University of Sibiu, Romania
- Prof. PhD. Luminita State - University of Pitesti, Romania
- Prof. PhD. Mihai Talmaciu - University of Bacau, Romania
- Prof. PhD. Milan Tuba - Megatrend University of Belgrade, Serbia
- Prof. PhD. Dan Eugen Ulmet - University of Applied Sciences Esslingen, Germany
- Assoc. Prof. PhD. Kiril Alexiev - Bulgarian Academy of Sciences, Bulgaria
- Assoc. Prof. PhD. Vasile Aurel Căus - University of Oradea, Romania
- Assoc. Prof. PhD. Ioana Chiorean - "Babes-Bolyai" University of Cluj-Napoca, Romania
- Assoc. Prof. PhD Daniela Danciulescu, University of Craiova, Romania
- Assoc. Prof. PhD. Stefka Fidanova,- Bulgarian Academy of Sciences, Bulgaria
- Assoc. Prof. PhD. Antoanelia Naaji - "Vasile Goldis" Western University of Arad, Romania
- Assoc. Prof. PhD. Victoria Iordan - West University of Timisoara, Romania
- Assoc. Prof. PhD. Adrian Florea - "Lucian Blaga" University of Sibiu, Romania
- Assoc. Prof. PhD. Petrica Pop - North University of Baia Mare, Romania
- Lecturer PhD. Mihaela Ciortea - "1 December 1918" University of Alba Iulia, Romania
- Lecturer PhD. Ralf Fabian - "Lucian Blaga" University of Sibiu, Romania
- Lecturer PhD. Daniel Hunyadi - "Lucian Blaga" University of Sibiu, Romania
- Lecturer PhD. Gabriela Moise - Petroleum-Gas University of Ploiesti, Romania
- Lecturer PhD. Corina Rotar - "1 December 1918" University of Alba Iulia, Romania
- Lecturer PhD. Florin Stoica - "Lucian Blaga" University of Sibiu, Romania
- Lecturer PhD. Anca Vasilescu - Transilvania University of Brasov, Romania

Contents:

Shop4j	11
Cristian-Ionel Aldulea, Teodora-Vanessa Gligor, Delia-Elena Iuga	
Coordinator: Silviu Razvan Dumitrescu	
Genealogy Web Based System	20
Atanas Anastasov	
Coordinator: Galina Atanasova	
Implementing a modular, extensible, yet small and cross-platform HTTP daemon	26
Barbu Paul - Gheorghe	
Coordinator: Adrian Florea	
Kinect game	40
Robert Sandica, Petrica Bota	
Coordinator: Dana Simian	
Student+ - an Android Application	46
Andrei Călbăjos, Bărbos Adrian, Kim Velker	
Coordinator: Camelia-M. Pintea	
Drive Me - A easy way to orientation	50
Cosmin Constantin Calin , Diana Maria Duca	
Coordinator: Dana Simian, Daniel Hunyadi	
PCTVPlay – advanced media player	56
Georgi Chavdarov	
Coordinator: Katalina Grigorova	
Concepts and technical solutions for building scalable cloud applications ...	60
Bogdan Cimpoesu, Radu Cigmaian	
Coordinator: Mircea Vladutiu	
The Future of Augment Reality	67
Octavian-George Cimpu	
Coordinator: Anca Vasilescu	
Improving word learning experience using spaced repetition technique implemented in Windows Phone 8 application	74
Mykhailo Dorokhov, Mariia Honcharova	
Coordinator: Iryna Ushakova, Snezhana Lembik	

My Virtual Trainer. An application to get you fit in no time	87
Bogdan – Mihai Fieraru, Vasile – Sebastian Vidrea, Sorina – Gabriela Irimie	
Coordinator: Anca Vasilescu	
Plagiarism detector – Document Processing	92
Florea Andrei George, Marica Constantin Ionuț, Liță Alexandru Marian, Dragomir Ionuț	
Coordinator: Constantinescu Zoran, Nicoară Simona, Moise Gabriela	
QuickThink	103
Ilie Daniela, Hodina Ionut, Oprea Lucian	
Coordinator: Hunyadi Daniel	
Measuring the Similarity of Images Compressed Using JPEG Algorithm	108
Milos Jordanski, Nebojsa T. Peric, Radmila Sekulic	
Coordinator: Milan Tuba	
Computing Functions on Android	118
Gerhard Konnerth	
Coordinator: Delilah Florea, Adrian Florea	
Theoretical Bases of Building a Decision Support System (Dss) for eveloping Regional Innovation Program	128
Narine Manasyan	
Coordinator: Vladimir Chernov	
Alarm modules, a necessity for industrial software	132
Istvan Benjamin Pallo	
Coordinator: Dana Simian	
Real-time edge detection for video capture	141
Nebojsa T. Peric, Radmila Sekulic, Milos Jordanski	
Coordinator: Milan Tuba	
Probabilistic model for quantitative risk assessment of traffic accidents in Bulgaria	149
Kristina Petkova	
Coordinator: Lyudmila Dimitrova	
Screen Resolution Invariant Pattern Matching	161
Kristina Petkova	
Coordinator: Milena Lazarova, Hristo Aladjov	

Web Content Management System using ASP .NET, C#, SQL Server and Windows Azure	171
Petrescu Octavian Florin	
Coordinator: Maria Miroiu	
Real-time shape recognition using contours and Hough transform	175
Radmila Sekulic, Nebojsa T. Peric, Milos Jordanski	
Coordinator: Milan Tuba	
Tursib4iPhone. Public transport in Sibiu made easy	184
Emanuel Silivăsan	
Coordinator: Delilah Florea, Adrian Florea	
The logistics management a transportation system «point of departure – point of destination» in real time regime	190
Maria Sokolova	
Coordinator: Vladimir Chernov	
Highlight Image Filter Produces Outstanding Results in Optical Character Recognition	200
Iulia Știrb	
Coordinator: Horia Ciocârlie	
Comparison between the Optimization Techniques of Image. Filters Used to Improve Optical Character Recognition	211
Iulia Știrb	
Coordinator: Horia Ciocârlie	
3D Modelling of a Restaurant	221
Marius-Alin Țăpordei	
Coordinator: Dana Dânciulescu	
Real-time palm detection and animation.....	226
Vasil Tsolov	
Coordinator: Milena Lazarova	
Unlimited register machine simulator	238
Eva Tuba	
Coordinator: Eva Tuba	

On Investigating Promoter Areas for Regulatory Motifs in the Solanum Lycopersicum Genome	246
Dagmawi Zegeye	
Coordinator: Sofia Visa, Esther Van der Knaap	
Workflow management using QR codes	252
Victor Gheorghe Rusu	
Coordinator: Ralf Fabian	
J2XML Metro Converter.....	257
Gyunesh Shefkedov	
Coordinators: Senior Assistant Valentin Velikov	
List of authors	263
Sponsors	269

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Shop4j

**Cristian-Ionel Aldulea, Teodora-Vanessa Gligor, Delia-Elena Iuga
Teacher Coordinator: Silviu Razvan Dumitrescu**

Abstract

Our project is an online store that enables administrators to configure and organize their online business dynamically, and allows customers an easy way to navigate and purchase products. Therefore, the administrator can define the desired categories, add or remove products, sizes for products and can manage discounts. Customers can change personal data, security data, recover their password, send orders, view history of orders, return orders, add and remove products from the cart. To provide ideas to our clients, our project implements a recommendation system which provides the most suitable products with the customer profile.

1 Why we made our choice

Our team thinks that e-commerce is a booming industry, and in the future almost everything will be able to be sold online, so we decided to make an application that can solve both the problems of users and administrators. The main problem of this type of solution, which can make the difference between two applications in the eye of a client is the usability. From the perspective of an administrator the most important objective is that they can perform their task easier which after all is all about usability. So the big problem that we want to solve is the usability.

2 About our application

2.1 Technology that we used

Our application is designed exclusively in Java EE 6 and run in JBoss AS 7 server applications. Connection to the database (MySql) is done through the server as application security (done by the security module which implements API Java Authentication Authorization Service – JAAS). The web client interface is developed using the PrimeFaces 4.0 framework which is based on JSF 2.2. To facilitate the use bookmarks we used PrettyFaces 3.3.3 that maps in an elegant manner URLs. Image management, which requires storing and searching them effectively is done through JCR Jackrabbit 2.3.7. Accessing each image is achieved by using a REST service. Also, this service is used for the password recovery section, which consists of sending a URL, through email, to the client. The URL can be used to access a page that allows the client to change the password. To meet the needs of many customers, we provide an internationalization system (EN / RO). Managed beans are used to build view components which are displayed on the page. The service layout of our application is made using Enterprise Java Beans (EJB) which are managed by Jboss container as well as Managed beans. The database is mapped using JPA 2.1 API. To search for products we use indexes, produced and maintained using Hibernate Search 4.2.0 and Hibernate Search Analyzer 4.4.0. Because our goal is usability we implement search with autocomplete which help the customer when they want to find a product. Another feature available through our application is the PDF report generation (using itextpdf 5.4.3), XML (using

JAXB), CSV and JSON because we believe these are very useful for good management. The recommendation system is built using Mahout library.

2.2 System requirements

The system requirements for this software are low-cost because it is a web application and all the business is executed on the server side. So everything the client needs is a browser which can run in almost every OS.

2.3 How the application works

For our application we have defined five roles: visitor, client, operator, admin and superadmin.

- The visitor is the person who first enters the store and does not have an account yet. They have the possibility to create an account, search (by a keyword and price) and visualize products.
- Clients inherit all the rights of the visitor after they login using their account. In addition, they can add / delete / edit the desired products in the cart, complete the order, view order history and return orders that do not meet their desires. They also have the ability to edit their account, recover their password and communicate with operators.
- The operators can view the entire list of clients and delete them (unless the client has orders in progress); they also have a role in managing orders (delivery, communication with the client during the delivery process and return of orders). When there are not enough items on stock, the operator must supply stock. They can change the time interval in which the link sent by email to recover password is valid, and finally, like all other users, they can change personal data and security data associated with their account.
- Administrators inherit all rights that the operators have. In addition, they manage operators, products, photos, discounts, newsletters and can export database files into XML / CSV.
- The superadmins inherit all rights that the administrators have. In addition, they manage administrators, product categories, colors and sizes which depend on the category.

2.3.1 Visitor

2.3.1.1 Create account

Visitors can create an account accessing Create account page from menu. There, they must complete fields and press the create button. If the fields are not filled correctly, error messages will be shown (Fig. 1).

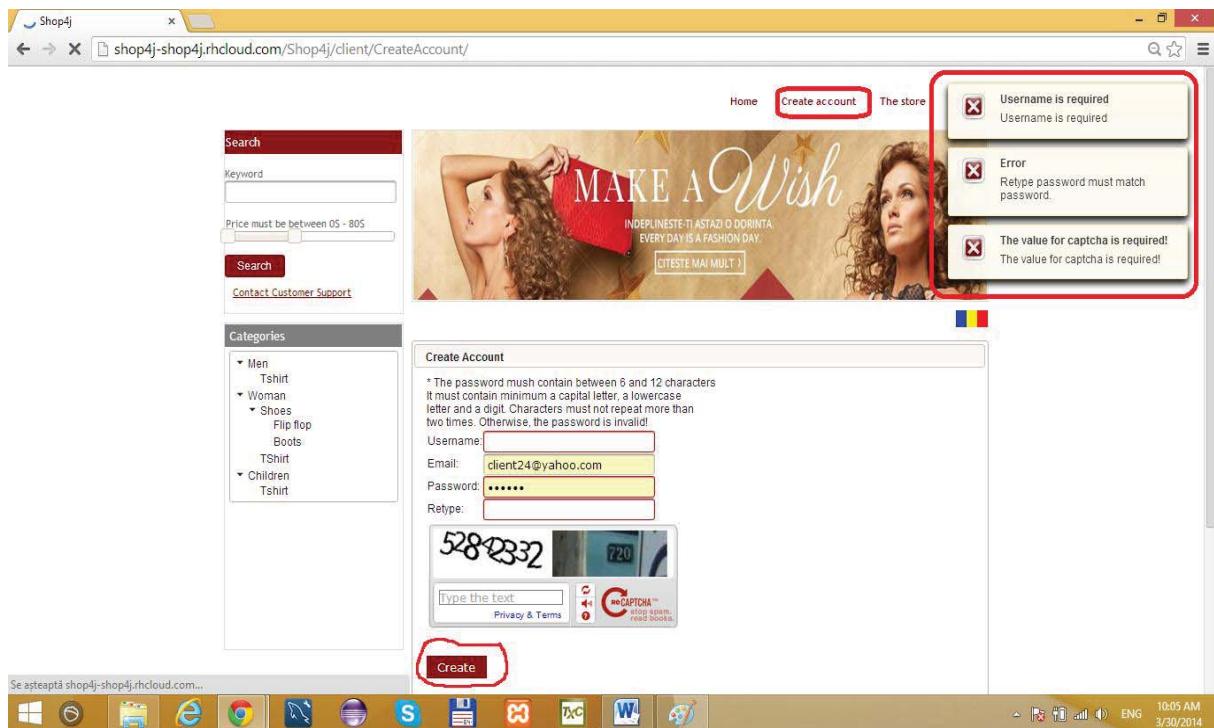


Fig. 1. Create account page

2.3.1.2 Search

Searching can be done using a keyword and the result can be filtered by price. We used hibernate search to implement this functionality and create a bridge between category and products, so the keyword refers at the same time to products and categories (when the user searches using a keyword, the software will look at the product's name and at the category name). If the user wants to use search, they must type a keyword, choose a price range (this is optional; otherwise minimum and maximum prices are selected by default) and press the Search button (Fig. 2).

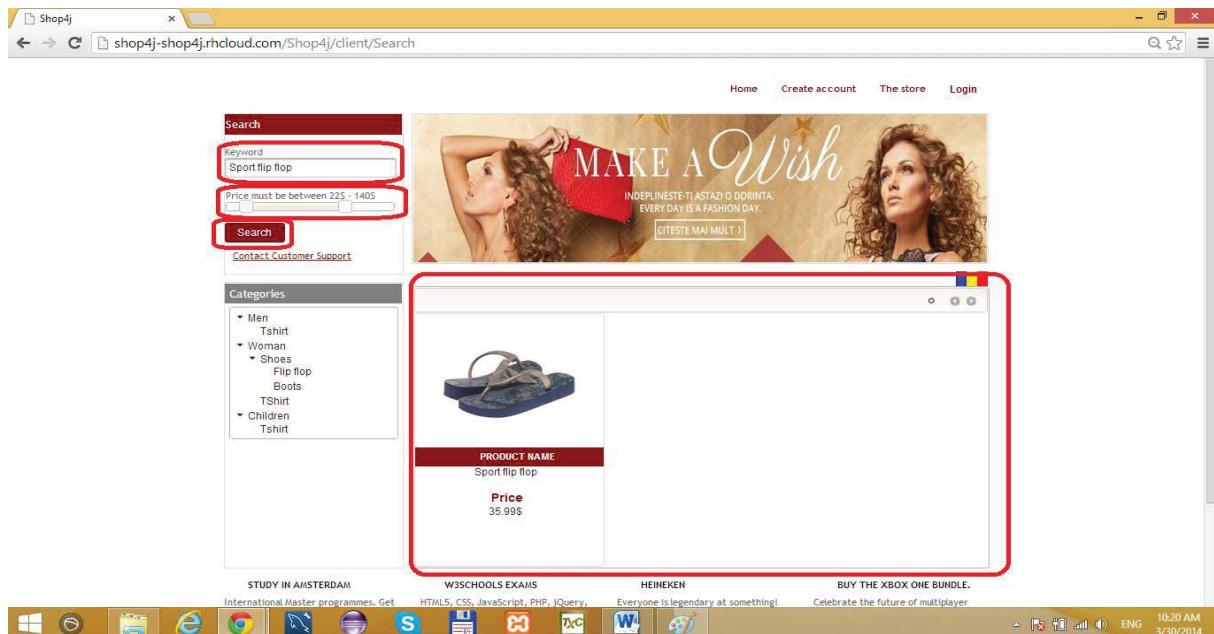


Fig. 2. Search form

Because we think usability is very important, we implemented an autocomplete feature to the search. So, if the user types at least 3 characters and the system detects some results, then a list with possible products or categories will be shown (Fig. 3).

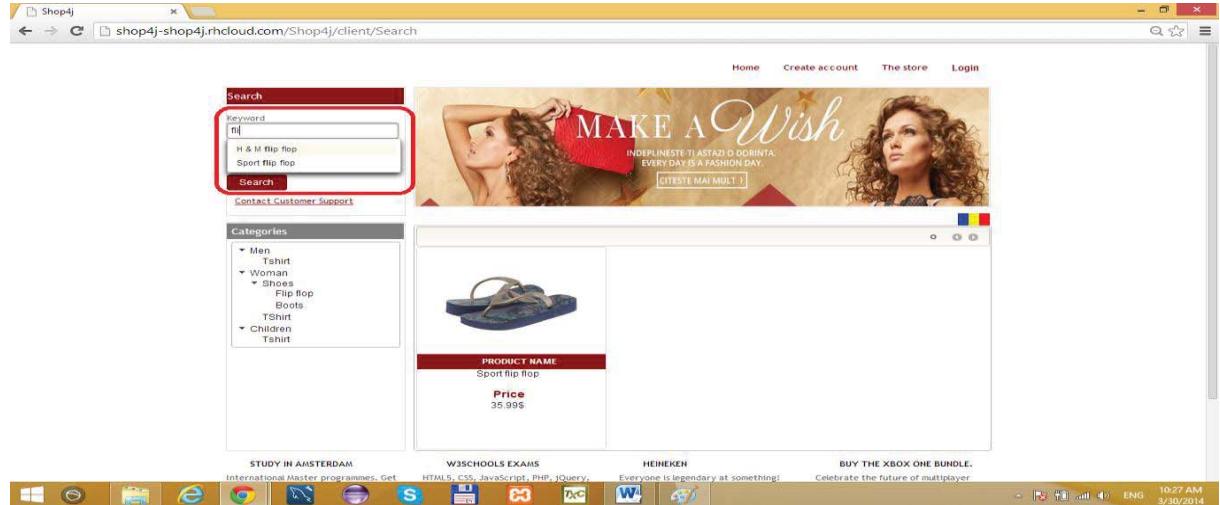


Fig. 3. Search autocomplete

2.3.1.2 View products

A visitor can view all the available products before creating an account and becoming a client. They can do this by pressing on the name of one of the categories. Afterwards, a page containing the results will be displayed. Our paging lets the client choose the number of products to be displayed on the page (Fig. 4).

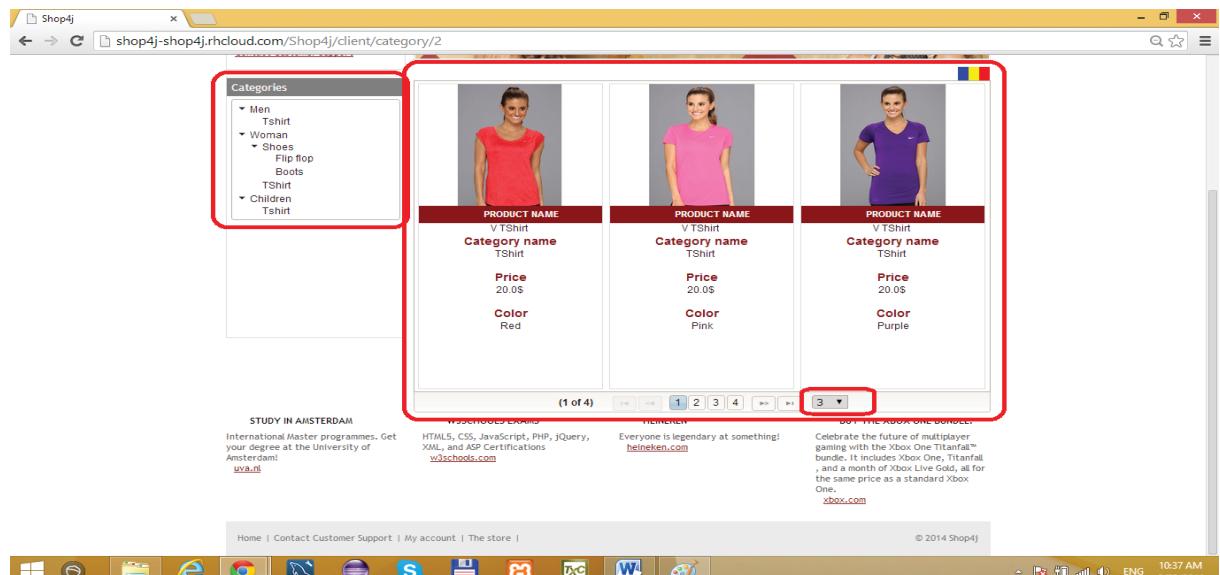


Fig. 4. View products page

2.3.1 Client

2.3.2.1 Change personal and security information

A client can modify their personal data by accessing the My account option from the main menu, then Personal data or Security data. Both options have validators, and if the data is incorrect, it signals this by error messages (Fig. 5).

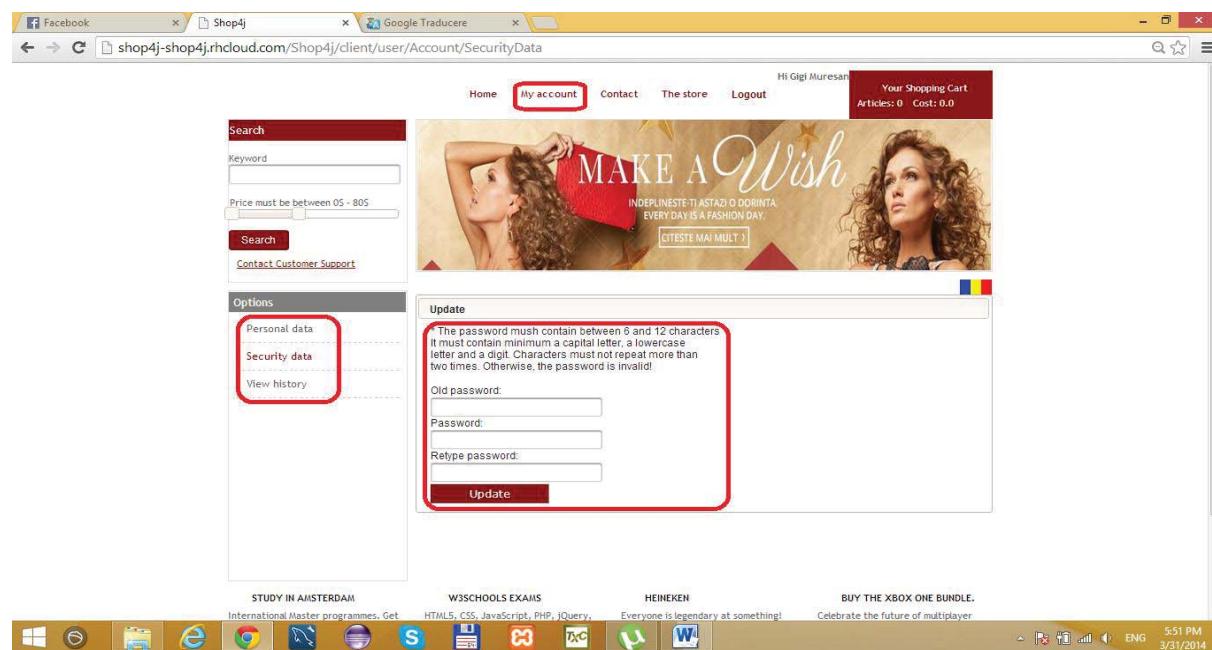


Fig. 5. Change security form

2.3.2.2 View command history

With View history on the left menu which appears when a client presses on the My account option, a client can view the entire history of commands and can return a command (Fig. 6).

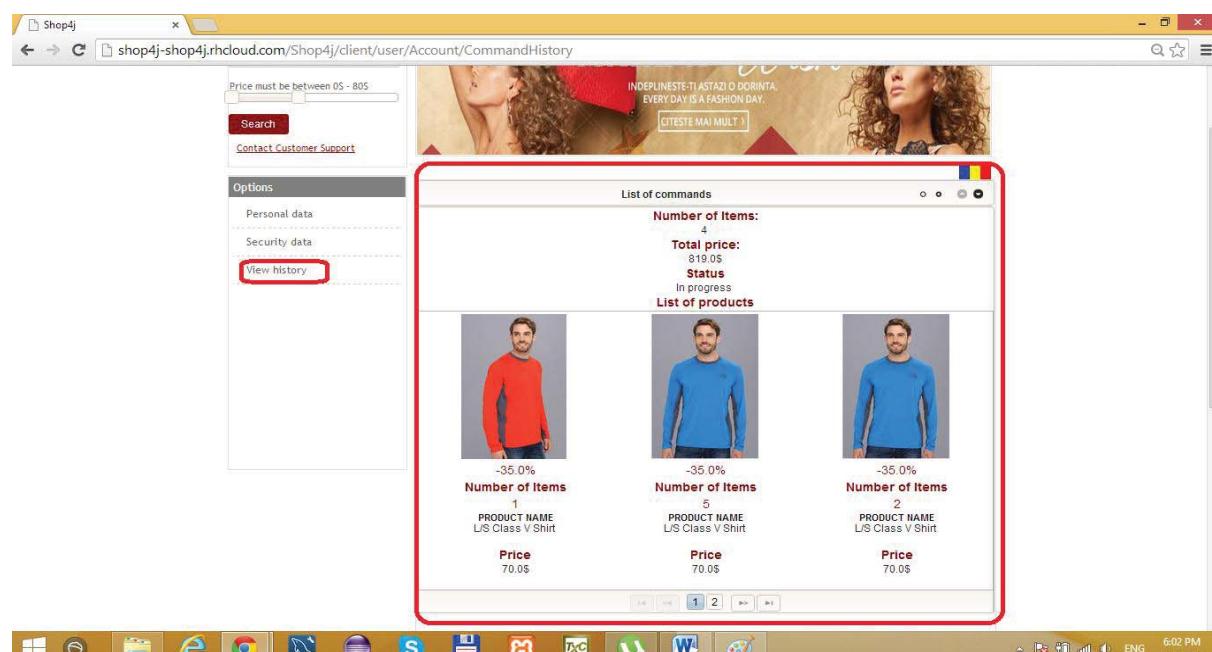


Fig. 6. Command history form

2.3.2.3 Product description

After the customer chooses a product, a page with more details will be opened. The picture with the product changes every time; the customer can change the color from the color list. Also this page is used to add products to the cart, choosing the number of items and pressing the Add to cart button. If the number of products is invalid or there are not enough products on stock, an error message will be shown (Fig. 7).

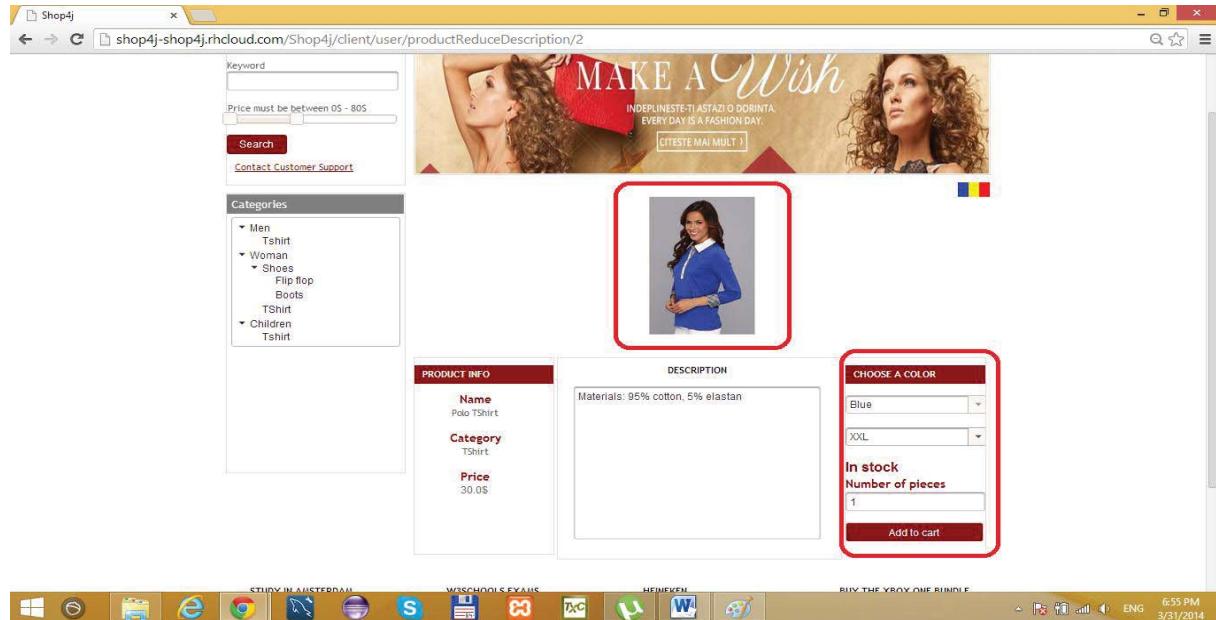


Fig. 7. Product description page

2.3.2.3 Operations on the cart

When the client moves the cursor on the cart, a dialog with a short description with the content of the cart opens. This dialog also allows the user to remove product from the cart and to access the page which contains more details about the cart (Fig. 8).

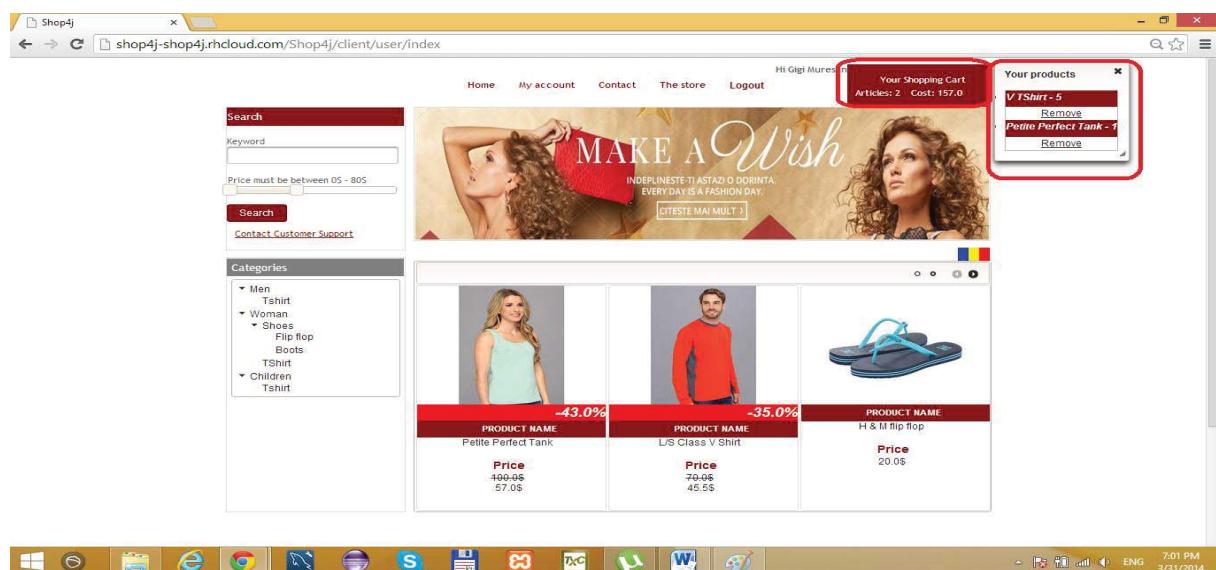


Fig. 8. Cart summary

The page containing the description of the cart, consists in two sections. The first section contains the description of the products which were added to the cart (Fig. 9).

The screenshot shows a shopping cart page with the following details:

- Header:** Hi Gigi Muresan, Your Shopping Cart, Articles: 4 Cost: 819.0
- Left Sidebar:**
 - Search bar with Keyword and Price range (05 - 805).
 - Search button.
 - Contact Customer Support link.
 - Categories tree:
 - Men: Tshirt
 - Woman: Shoes (Flip flop, Boots), TShirt
 - Children: Tshirt
- Banner:** MAKE A Wish INDEPLINESTE-TI ASTAZI O DORINTA. EVERY DAY IS A FASHION DAY. CITESTE MAI MULT!
- Cart Content:**

PRODUCT NAME	Price	PRODUCT NAME	Price	PRODUCT NAME	Price
L/S Class V Shirt	70.0\$ 45.5\$	L/S Class V Shirt	70.0\$ 45.5\$	L/S Class V Shirt	70.0\$ 45.5\$
- Right Sidebar:** Your products window showing two L/S Class V Shirts with Remove buttons.

Fig. 9. Cart page

The second section allows the client to send a command, containing the products from the cart. To do this he must choose an existing address or add a new one, and press send command button. As in other cases if the client fills in the forms with incorrect data, an error message will be shown (Fig.10).

The screenshot shows the 'Send command' form with the following fields:

- Address Selection:** Choose an existing address (checkbox) and Add new address (checkbox selected).
- New Address Fields:**
 - Country: Choose a country
 - County: Choose a county
 - Locality: Choose a locality
 - Street:
 - Number:
 - Block:
 - Staircase:
 - Flat:
- Message:** The field mark with * are mandatory.
- Buttons:** Send command, Your products window showing two L/S Class V Shirts with Remove buttons.

Fig. 10. Send command form from cart page

2.3.2.4 Forgot password

If a client forgets their password, they can enter their email on the “forgot password page”. Then an email will be sent to the client, with a link which can be used to change their password. The link will expire in one week.

2.3.3 Operator

An operator is a person which can see a list of clients, orders and reports. They can operate orders, returned orders and can communicate with clients.

2.3.4 Admin

Admin is the category of users that can add products to stock, make special offers, manage images and manage operators.

2.3.5 Superadmin

Superadmin is the category of users that can add or remove categories, colors and can manage operators and admins.

3 Technical aspects

Our application is designed exclusively in Java EE 6 and run in JBoss AS 7 server applications, which offers many facilities, and good performance. The main application is built on this platform and is compiled and archived using maven that allows adding new components (jar, war). The most challenging problems were created by the recommendation engine and the management system dealing with pictures.

3.1 Recommendation system

The entire system is build using Mahout, a library that was written in Java and offers support for data maintenance. We save data about user preferences in CSV files because in Mahout it is more efficient to work with files than to use a database. Communication between the database and application is more expensive than working with file because it provides the possibility to create updated files. (For example if we store the preferences in a file with name File.csv we can create files by following rule File.1.csv, File.2.csv,, File.n.csv with preferences expressed recently by users. Otherwise, files can be loaded to the recommender using a refresh method).

To make predictions, first of all, the engine loads the data through files. Then it calculates the distance between users. At the next step, the system groups clients in neighborhoods, applying a threshold to the previously calculated distance. (This is very similar with clustering). Finally, when an estimation is needed, it will look in the neighbourhood of the user and it will get the items with the best score.

4. Conclusions and future developments

This application was created to facilitate the work of both clients and administrators using some of the most powerful technologies for the web. So we think this software can add a major advantage for a new business. In the future we would like to develop system recommendations, create a cloud in which we can parallelize computations using Hadoop.

References

- [1] Sean Owen, Robin Anil, Ted Dunning, Ellen Friedman, *Mahout in action*, Shelter Island, 2012.
- [2] Documentation for Jboss, <https://docs.jboss.org/author/display/AS71/Documentation>
- [3] Documentation for primefaces, <http://www.primefaces.org/showcase/ui/home.jsf>

ALDULEA Cristian-Ionel
Transilvania University of Brasov
Departament of Mathematics and Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
Romania
alduleacristi@yahoo.com

GLIGOR Teodora-Vanessa
Transilvania University of Brasov
Departament of Mathematics and Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
ROMANIA

IUGA Delia-Elena
Transilvania University of Brasov
Departament of Mathematics and Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
ROMANIA

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Genealogy Web Based System

**Atanas Anastasov
Teacher Coordinator: Galina Atanasova**

Abstract

This paper presents a Genealogy Web Based System. This application allows its users to maintain the information of his/her family and the tracing of its lineages and history. There are researched some applications in this subject and their features are summarized. It is described the approach for Genealogy Web Based System creation.

1 Introduction

Genealogy comes from Greek ‘*genea*’ that means generation and ‘*logos*’ that means knowledge, also known as family history [2]. Actually, this is the study of families and the tracing of their lineages and history. Our Genealogy Web Based System, in brief called GWBS, is an application to help you to do genealogical research and to follow your own or someone else’s family tree and provenance. The program maintains its own database in which an user can enter and store a huge amount of data for all members, building a family tree, the relationships between them, and to create multiple flexible reports with summary information. Web page can be used for personal purposes and to get to know their own origin and to carry out detailed historical research on a person in your family.

The paper is organised as followed: 1 Introduction, 2 Genealogy Web Based System Review, 3 Description of decision, 4 Conclusions.

2 Genealogy Web Based System Review

2.1 Other Genealogy Web Based System Review

We are familiar with some other software applications such as “My Heritage” [4] which is the only genealogical page that supports Bulgarian language. This application realizes the most popular for that kind of software functions, but print function requires extra money payment. Positive aspects of the developed GWBS are accessible and money free to every its user.

2.2 Genealogy Web Based System Project

Every user of GWBS, that uses it for the first time, fills in a registration form and receives an user name and password for identification it the system. The user name and password allow access the

other application functions: information about him/her self entering, information about him/her family members entering, create tree, add a person, delete person, delete tree, tree visualization. Nowadays the UML [1] is accepted as a standard language for software project description due to the platform-, technology-, and program-independence of this graphical language. The described above functions are separated in such that require from user input information and such that application generates from stored data. The common use case diagram presenting the user depending and the application depending features of the GWBS are given on the following (Fig.1).

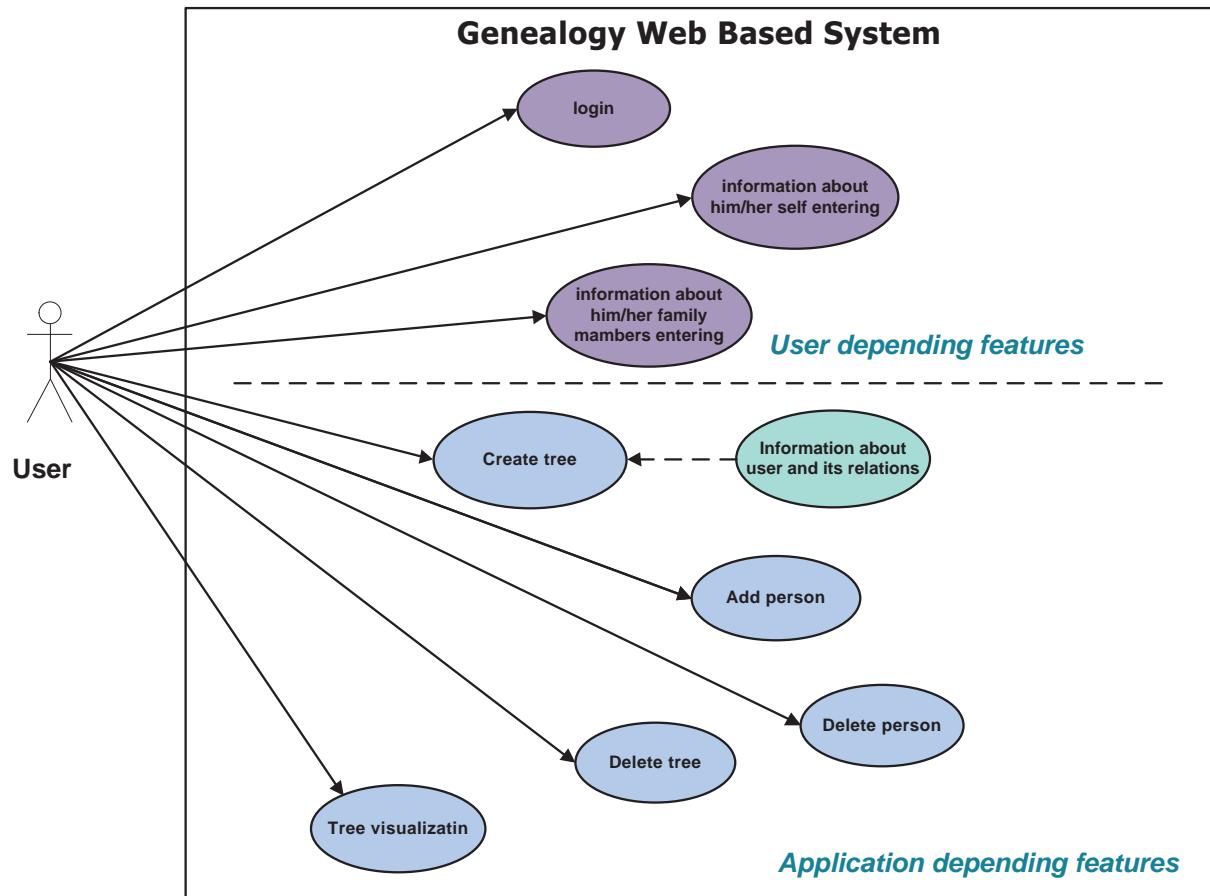


Fig. 1: GWBS common use case

3 Description of decision

3.1 Program code

It is used last of language HTML, namely HTML5 to construct the basic structure of the GWBS application [3]. For complete and design application pages there are used other programming languages and graphic editors such as Photo Shop, CSS3, PHP. On the program fragment shown below is demonstrated the pages' navigation:

```

...
<nav>
    <ul id="nav">
        <li class="current">
            <a href="index.html">Home</a>
        </li>
        <li>
            <a href="#">Form</a>
            <ul>
                <li>
                    <a href="New.html">New</a>
                </li>
                <li>
                    <a href="Add.html">Add</a>
                </li>
                <li>
                    <a href="#">Delete</a>
                </li>
            </ul>
        </li>
        <li>
            <a href="Tree.html">Tree</a>
        </li>
        <li>
            <a href="Contacts.html">Contacts</a>
        </li>
    </ul>
</nav>
...

```

3.2 Genealogy Web Based System page

This description can't fully demonstrate the capabilities of the GWBS. You can see designed application view on fig. 2. Instead, it will give you general guidelines on how to use the program so that you can get started in a Genealogy Research. When you first start you will need to create a new family tree, which will actually be used as a reference database next time you log into the program.

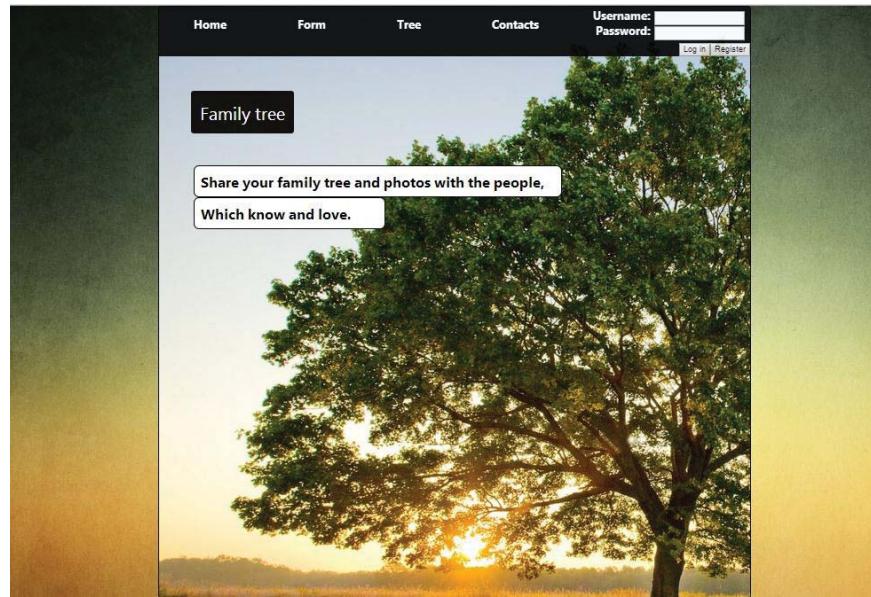


Fig. 2: GWBS main page

3.3 Genealogy Web Based System basic form

By pressing the 'New' button from drop-down menu shown on fig. 3 user can choose to create a new family tree as it is shown on fig 4.

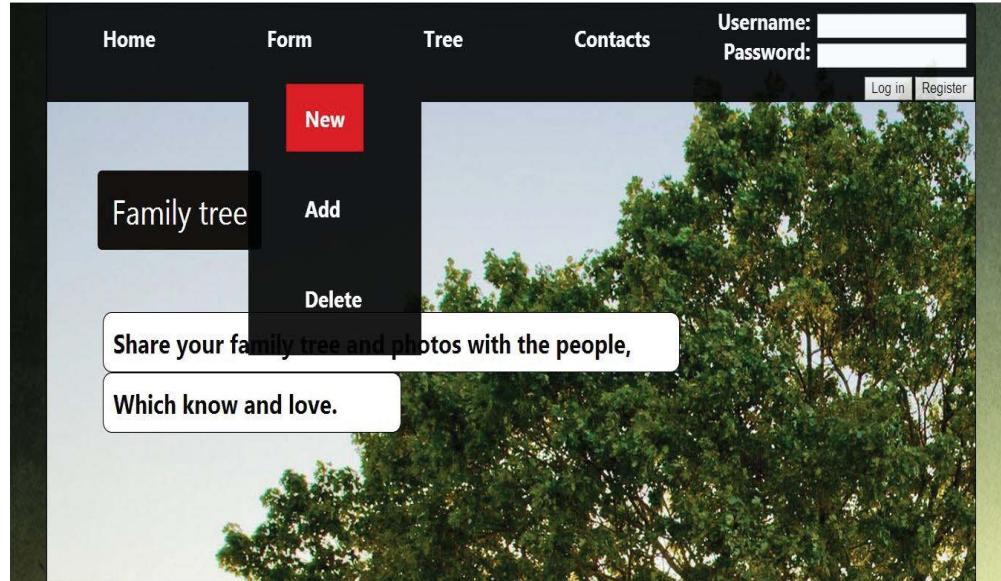


Fig. 3: Drop-down menu with 'New' button on the basic GWBS form

Create tree

Last name	<input type="text"/>		
First Name	<input type="text"/>		
Address	<input type="text"/>		
City	<input type="text"/>		State <input type="text"/>
Zip/Postal Code	<input type="text"/>		
Country	UK <input type="button" value="▼"/>		
Phone(country code,area code,umber)	<input type="text"/>	<input type="text"/>	<input type="text"/>
E-mail	<input type="text"/>		
Birth date	Month <input type="text"/>	Day <input type="text"/>	Year(4 digit) <input type="text"/>
Gender	Male <input type="button" value="▼"/> Male Female		
<input type="button" value="submit"/> <input type="button" value="Clear This Form"/>			

Fig. 4: GWBS create tree form

3.4 Genealogy Web Based System Add person form

By clicking the button ‘Add’ from the drop-down menu an user choose to append a member to the family tree user can see this in fig. 5. An user also can choose his/her relationship from other drop-down menu which is located in the lower right corner.

Add person

Last name	<input type="text"/>		
First Name	<input type="text"/>		
Address	<input type="text"/>		
City	<input type="text"/>		State <input type="text"/>
Zip/Postal Code	<input type="text"/>		
Country	Bulgaria <input type="button" value="▼"/>		
Phone(country code,area code,umber)	<input type="text"/>	<input type="text"/>	<input type="text"/>
E-mail	<input type="text"/>		
Birth date	Month <input type="text"/>	Day <input type="text"/>	Year(4 digit) <input type="text"/>
Gender	Male <input type="button" value="▼"/> Male Female		
family relationship	Mother <input type="button" value="▼"/> Mother Father Brother Sister Son daughter grandfather grandmother		
<input type="button" value="submit"/> <input type="button" value="Clear This Form"/>			

Fig. 5: GWBS add person form

3.5 Genealogy Web Based System Tree visualization

By clicking the button Tree it is appearing the family tree that an user have created as it is shown on fig 6.

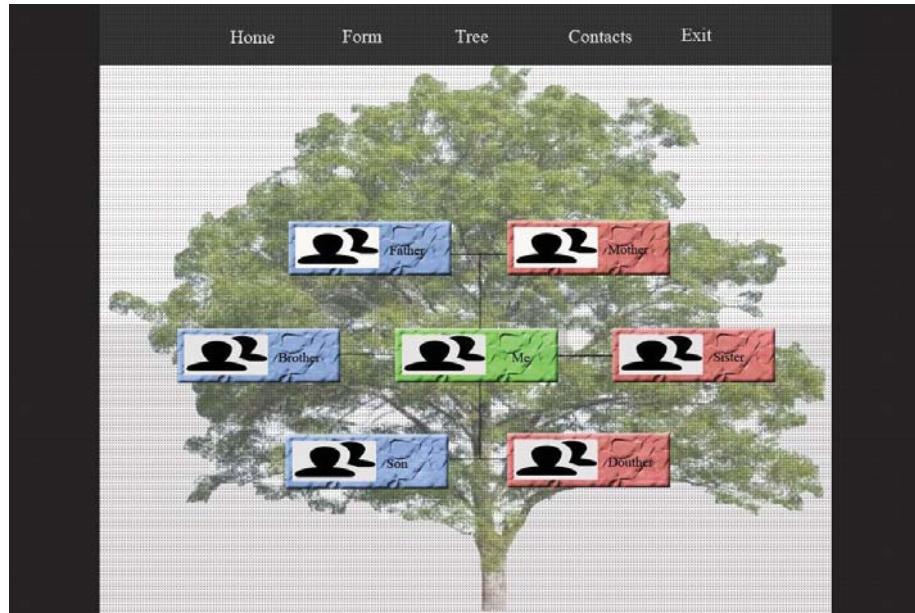


Fig. 6: Family Tree in GWBS window

4. Conclusions

The developed GWBS equip its users with abilities to store and to visualize information about its' family members and to keep this memories in the time without extra money.

References

- [1] Fowler, M., *UML Distilled: A brief guide to the standard object modeling language*, 3rd edition, Addison Wesley, 2004
- [2] <https://www.google.bg>.
- [3] <http://academy.telerik.com>.
- [4] <http://www.myheritage.com/index.php?lang=BG>

Atanas ANASTASOV
 University of Rousse
 Natural Sciences and Education Department
 Studentska Str. 8
 BULGARIA
 Atanas.Y.Anastaasov@gmail.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Implementing a modular, extensible, yet small and cross-platform HTTP daemon

**Barbu Paul - Gheorghe
Teacher Coordinator: Adrian Florea**

Abstract

HTTP daemons have become a fundamental part of the web as we know it. This paper presents from a high-level perspective how a HTTP daemon works and how one can be implemented in order to serve static files and even dynamic content. The HTTP daemon presented here is implemented in such a way that it allows programmers to extend its functionality and, due to the small size and modularity of the components, even to modify its inner mechanisms. In aid of that goal the Qt¹ framework is used, combined with the C++ programming language in order to keep the daemon's resource usage at a minimum and to maintain portability.

1 Introduction

Today's world is dominated by the internet, with over 70% of individuals using the internet in the developed countries [1]. Every online resource accessed through a web browser is served by applications running 24/7 on a server farm. HTTP daemons are the backbone of this complex infrastructure, every website is served to clients through such a piece of software. With this premise in mind this paper aims to introduce the concept of a HTTP daemon and its implementation (at a high level) to programmers.

One of the key features of the daemon is its extensibility, which means that anyone can add functionality to the daemon's core via plugins. This is the reason the application is split into three main parts, freely accessible under the Apache v2 license at the following locations:

Core: The kernel of the application that listens for and accepts requests, routes them to the plugins and then sends the response created by the plugins back to the client. <https://github.com/paullik/http-daemon>

Components: The interface that creates the link between the core and the plugins, also serves as an abstraction level for the daemon's implementation. <https://github.com/paullik/http-daemon-components>

¹<http://qt-project.org/>

Plugins: Usually 3rd party shared libraries that add functionality (through use of the components) otherwise unavailable in the core. Two examples of plugins will be described later on. <https://github.com/paullik/http-daemon-plugins>

In the following sections we will refer to these parts as **core**, **components** and **plugins**.

The following sections describe the current work in the field, explain similarities and differences between my daemon and some popular ones like Apache ² and Lighttpd ³. Afterwards I proceed to explain the architecture of the daemon and how it processes requests and how it sends responses.

2 Related work

2.1 Apache

According to their website this is “The Number One HTTP Server On The Internet” [2]. An open source and free daemon, available on a number of platforms, that was an inspiration source for my daemon which is also free, open source and cross-platform.

Apache is written in C, whereas my daemon is written in C++. A similarity is the use of the APR library by the Apache web server and the use of the Qt framework by me, both provide abstraction over the underlying operating system [3] allowing a programmer not to worry about a feature being implemented in the operating system he’s developing on.

Apache implements 3rd party functionality using the so called modules⁴, compared to the plugins that enhance the functionality of my daemon.

Also Apache is compliant with HTTP/1.1⁵, but my daemon has only partial compliance for HTTP/1.0 [4].

Similarly to Apache’s configuration⁶ my daemon’s configuration file allows for variable substitution by using the syntax {var} where var is the name of the variable to be substituted by its contents, an example of this may be seen in the configuration section for the **fileplugin** in **httpdaemon.conf**⁷.

2.2 Lighttpd

Lighttpd claims to provide security, speed, and compliance, it is renowned for its small memory footprint and advanced feature set. It allows modules to extend its core functionality.

Both Apache and Lighttpd provide conditional configuration.

Similarities between Lighttpd and my daemon include the event driven architecture, which for me is provided by the Qt Signals & Slots mechanism [5], this architecture provides a high level of flexibility because the core of the daemon listens to some signals emitted by other parts of the system and takes action when these signals are received. The advantage of this mechanism is that there is no coupling between system parts like it would be if a callback-based architecture was used, since the signal emitters mustn’t have any knowledge of which parts are listening for signals. The only required knowledge for the listening part is to know in advance what signals

²<https://httpd.apache.org/>

³<http://www.lighttpd.net/>

⁴<https://httpd.apache.org/modules/>

⁵<https://httpd.apache.org/docs/trunk/compliance.html>

⁶<https://httpd.apache.org/docs/2.4/configuring.html#syntax>

⁷<https://github.com/paullik/http-daemon/blob/master/httpdaemon.conf#L58>

may be emitted, and so the emitter part is required to declare what signals it may send out, this is simply done by declaring some methods for specific classes.

2.3 Others

Other daemons include IIS⁸, a HTTP daemon developed by Microsoft, nginx⁹ and the CERN httpd¹⁰, the first HTTP daemon, but no longer maintained. A more complete list may be found at https://en.wikipedia.org/wiki/Comparison_of_web_server_software.

3 Description of the implementation

The core of the daemon is composed of several classes responsible for accepting a request, parsing it into the proper structures, then routing it to the plugins.

The plugins are then capable of creating a response, wrapping it in the needed form so that it may be sent to the core. The plugins then signal the core that the processing has ended and that the response should be sent to the client. The core takes over once again and sends the HTTP formatted [4] response back to the requester. Figure 1 describes the design of the core.

⁸<http://www.iis.net/>

⁹<http://nginx.org/>

¹⁰<http://www.w3.org/Daemon/>

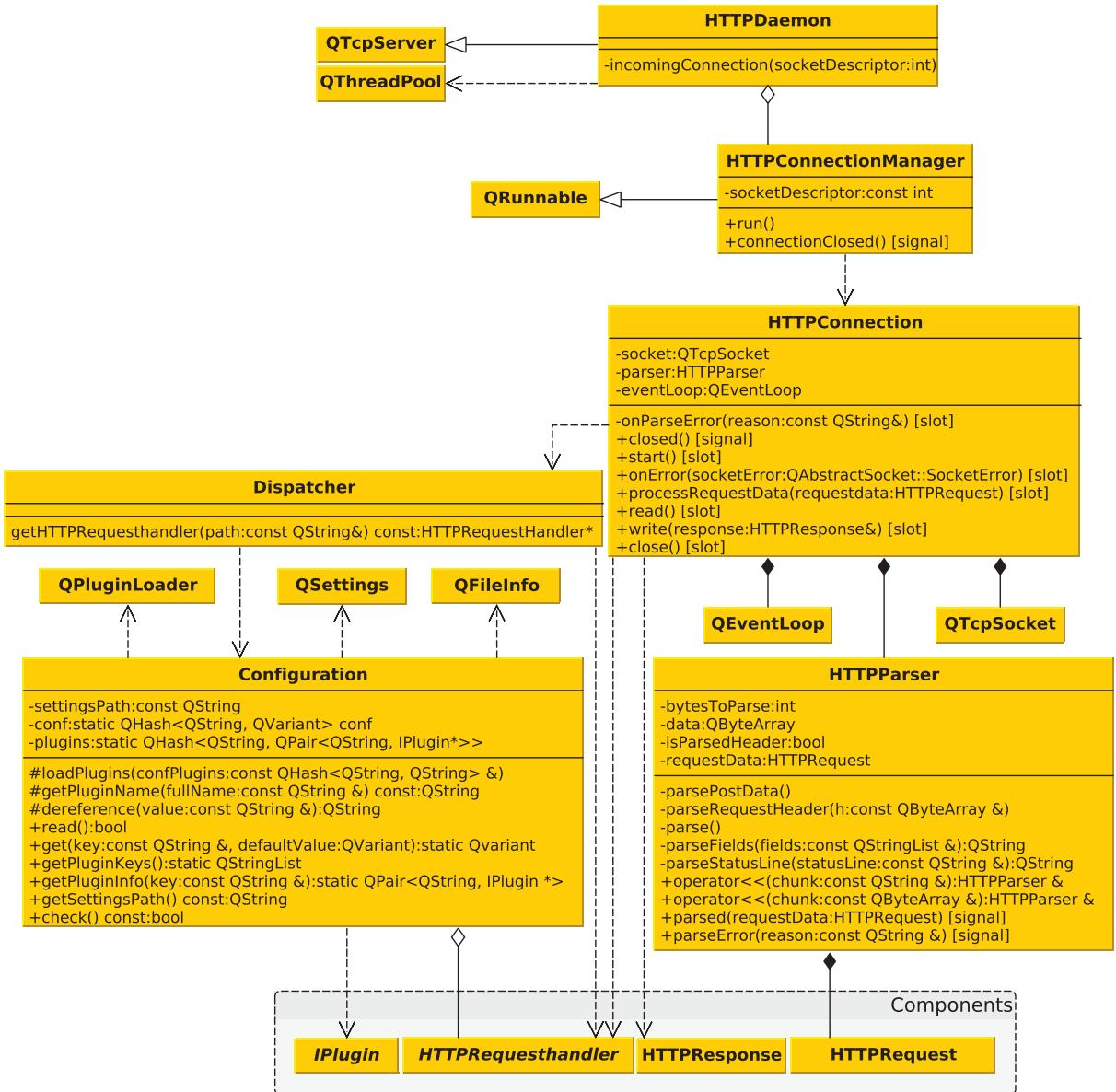


Figure 1: Core architecture

The entry point in the application is the `main()` function which is responsible for `fork()`-ing the daemon, changing directory, changing the user. In other words responsible for "transforming" the application into a daemon [6]. After being run as a daemon, the application instantiates the `HTTPDaemon` class.

The settings for what port the daemon should connect to, what user to change to, paths to plugins and the document root are all managed by the `Configuration` class which descends from `QSettings`¹¹.

`HTTPDaemon` inherits from `QTcpServer` and is in charge of accepting the requests via the overridden `incomingConnection()` method, here an object of type `HTTPConnectionManager` is

¹¹<http://qt-project.org/doc/qt-5/qsettings.html>

created and it's fed to the thread pool to be run. This way I accomplish concurrency while keeping a fair amount of threads running, since if all the threads in the pool have been assigned work, then other connections will be queued, being a FIFO queue the first queued connection is served by the first thread that finishes its job. It is necessary to use a thread pool because otherwise one cannot control how many threads the daemon will start and the system will be flooded. The number of threads in the pool will be assigned by querying the number of cores of the system [7].

When a thread starts working a `HTTPConnection` object is created. This object is in charge of receiving a request from the client and parsing it using a `HTTPParser`. The parser creates a `HTTPRequest` object that encapsulates every aspect of the current request, this request object is then passed to the `Dispatcher`. The `Dispatcher`'s job is to find the needed `HTTPRequestHandler` for the current client. Thread safety is simply achieved by getting a new instance of the `HTTPRequestHandler` every time a connection comes in, this avoids concurrent access to the plugins' internals since a new instance is used for each connection.

The response and request types mentioned earlier are part of the components package, the architecture diagram of this package can be seen in Figure 2.

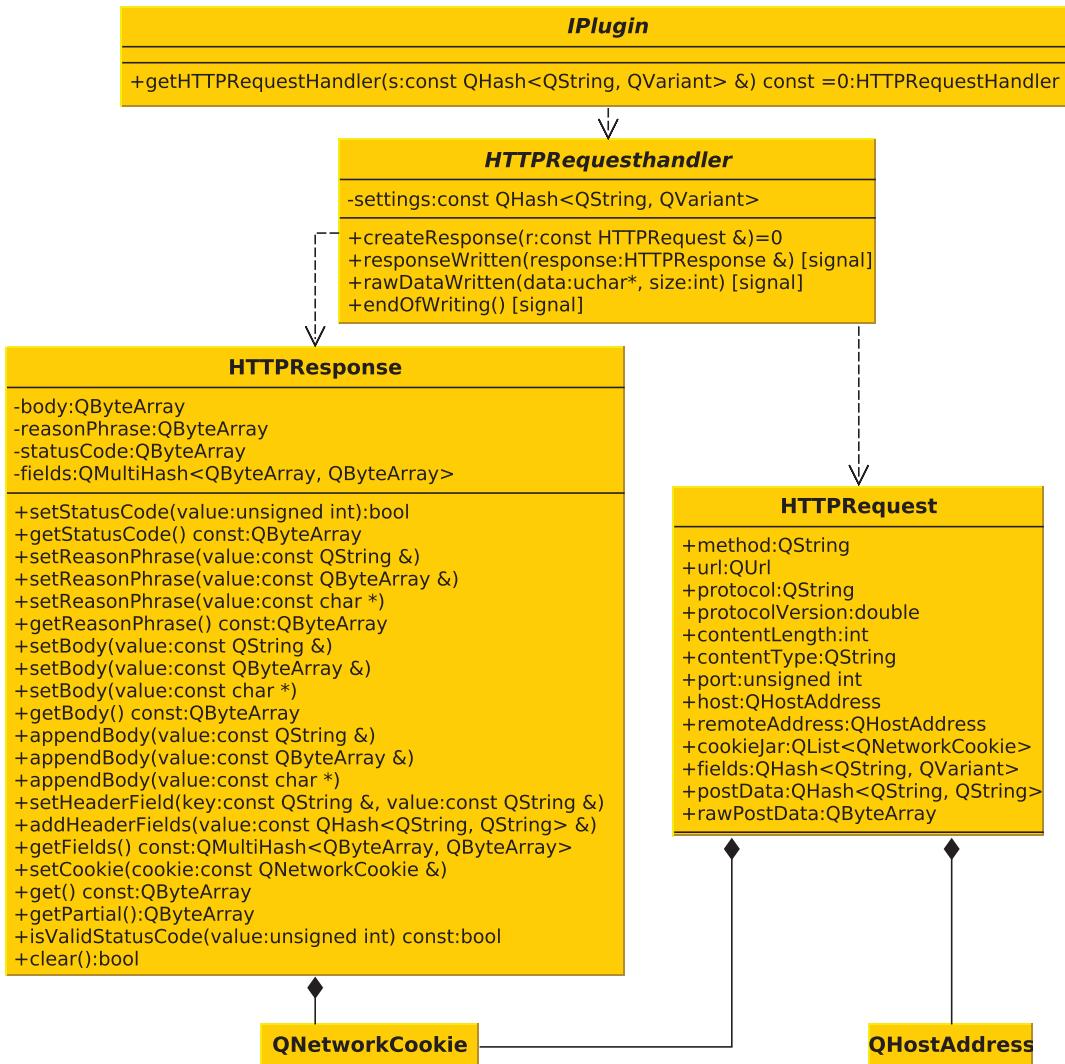


Figure 2: Components architecture

Here is declared the `IPlugin` interface. It defines the interface a plugin has to implement in order to be compatible with the core of the daemon. Every plugin has to return (through `getHTTPRequestHandler()`) a pointer to an instance of `HTTPRequestHandler` which is custom to what the plugin does.

For example the `FilePlugin` in Figure 3 implements the `IPlugin` interface and returns a `FileHTTPRequestHandler` that will be used by the core to handle the request to fetch a file. This is done by `connect()`-ing [5] to the `responseWritten()` and `rawDataWritten()` signals provided by the `HTTPRequestHandler` abstract class.

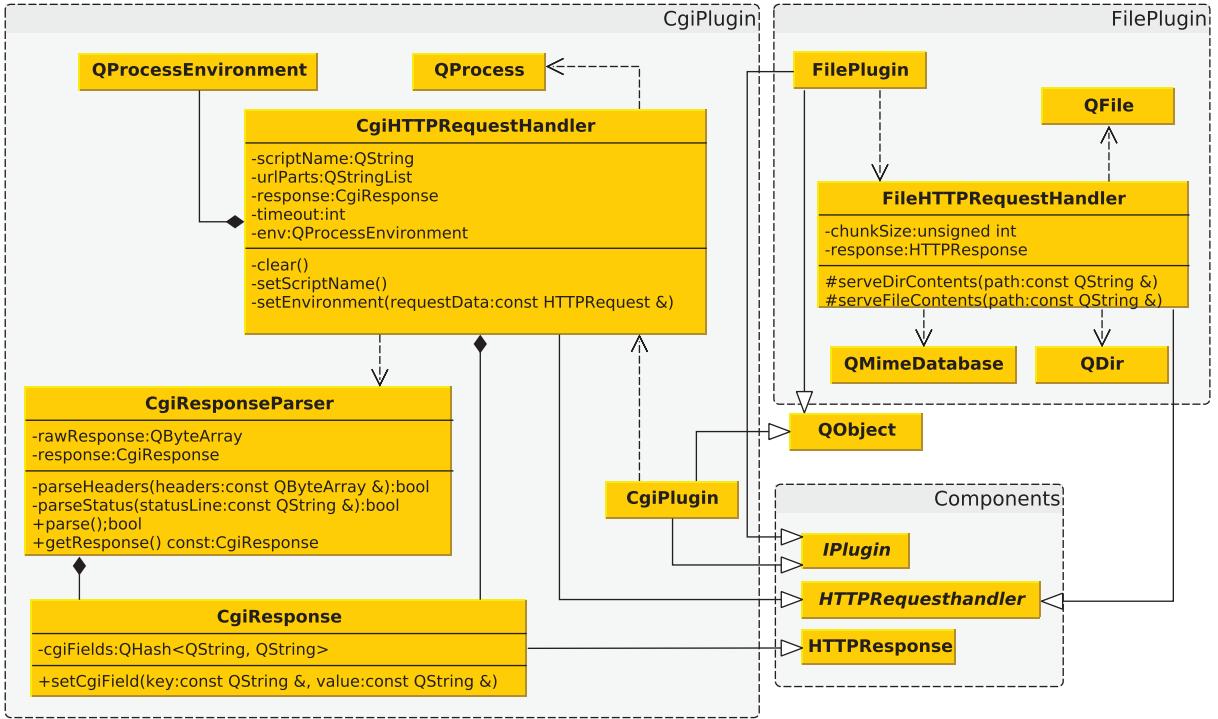


Figure 3: Plugins architecture

Another example is the `CgiPlugin` that partly implements the CGI protocol [8]. This plugin is more complex because it adds the `CgiResponseParser` and `CgiResponse` classes. These are needed since after `CgiHTTPRequestHandler` has started the CGI script, the response must be parsed and transformed in a format understood by the core of the daemon, because CGI response headers are different from standard HTTP response headers. After the CGI response has been parsed the core will be signaled to take over and send it to the client. This plugin also implements a configurable timeout system, if the CGI executable takes more than a predefined number of milliseconds to execute, it will be interrupted and an error will be sent to the client instead of the expected response, this is a preventive measure against ill-intentioned applications, but also against programmer errors that result in an infinite execution cycle.

The whole plugin system is based on the `QPluginLoader`¹² class provided by Qt, which is just a wrapper around an operating system specific method of loading shared libraries and retrieving a C++ object, without having to use a C function to do the retrieval.

4 Benchmarks

Because of the general high load of the web servers [9] it is important to be aware of the performance of my application. So I have run some benchmarks for my daemon, Apache, and Lighttpd and then compared the results. The tests ran on an Intel(R) Pentium(R) D CPU 3.00GHz CPU, 2GB RAM with an up-to-date Arch Linux system¹³.

¹²<http://qt-project.org/doc/qt-5/qpluginloader.html>

¹³output of `uname -r`: 3.12.9-1-ARCH

I conceived three tests so that each of them targets a specific part of the HTTP daemon and puts it in perspective with the other two benchmarked candidates:

- The first test requests a directory (the document root) which may be thought of as serving a small file, with the added difference that the daemons have to do some additional work in order to fetch the body of the response.
- The second test requests a CGI script that serves dynamic content in order to mimic a user-made application.
- The last test fetches a big¹⁴ file, that serves as an analogy to downloading a file from an online sharing service.

Each test was conducted three times for each daemon and the best¹⁵ was chosen as representative. No test was conceived to contain any requests to a database since benchmarking the database response times is out of scope in this document.

The tests were ran using `ab`¹⁶. This tool was chosen because it is the original tool used for the Apache daemon and it is shipped together with it. Although not as feature rich as other tools¹⁷ it is perfect to give some insights on how the daemon performs under high load.

The measurements were done by performing 1000 requests with a concurrency level of 50. The final, general, `ab` command being:

```
ab -n 1000 -c 50 http://<host>/<path>
```

Where `<host>` is the address of the host the daemon was running on and `<path>` being the path requested, which, of course, is different from test to test.

For every test `ab` produced a set of data¹⁸ that I plotted. The x-axis is the current request number and the y-axis is the time taken to respond to that request¹⁹. The general trend is that requests made at a later point in time take longer to be complete because they are added in a queue waiting for the earlier requests to finish.

4.1 Requesting the document root

First test measured how the daemon behaves under high load when clients request the document root. The results, together with the comparison to the other two daemons, can be seen in Figure 4.

¹⁴ \simeq 100 MB

¹⁵the highest average number of requests per second and the lowest average time per request

¹⁶ApacheBench, Version 2.3, <https://httpd.apache.org/docs/2.4/programs/ab.html>

¹⁷https://en.wikipedia.org/wiki/Web_server_benchmarking#Tools_for_benchmarking

¹⁸Freely available at <https://github.com/paullik/http-daemon-data>

¹⁹measured in milliseconds

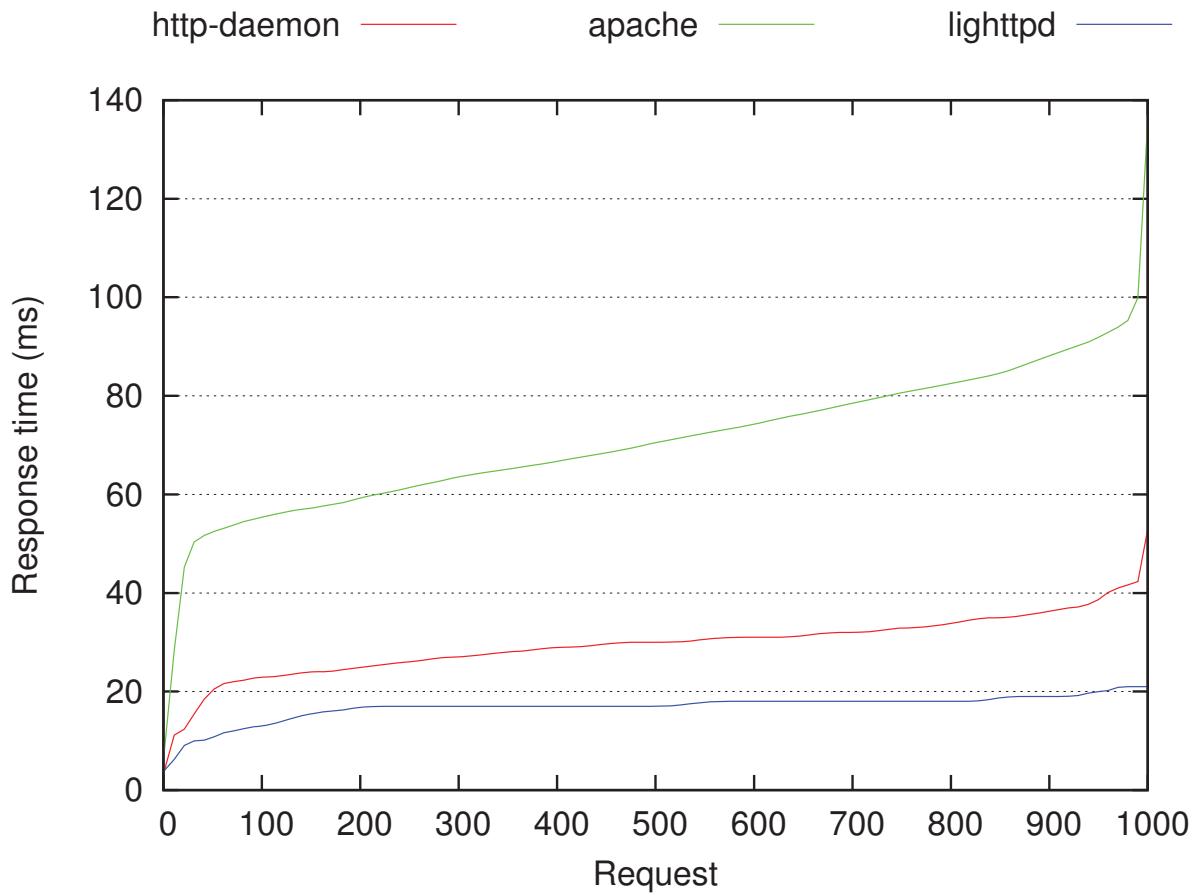


Figure 4: Requesting the document root

Due to the simple output format of my daemon, which is directly related to the small size of the HTTP response, its performance is better than Apache's, but still worse than that of Lighttpd. However this is important when creating a web server only for serving mirrored files, the users will be able to quickly navigate through the directories.

4.2 Requesting a CGI script

The second test is about how fast a CGI request is satisfied. These tests may seem inappropriate since the creation of the CGI process and the `stdin` and `stdout` communication take their toll on the performance of the actual HTTP request - response cycle, but the tests were conducted in similar conditions so the CGI overhead should, in theory, be the same for each daemon, because the same CGI executable was used, a PHP script displaying `phpinfo()`. The plotted results are represented in Figure 5.

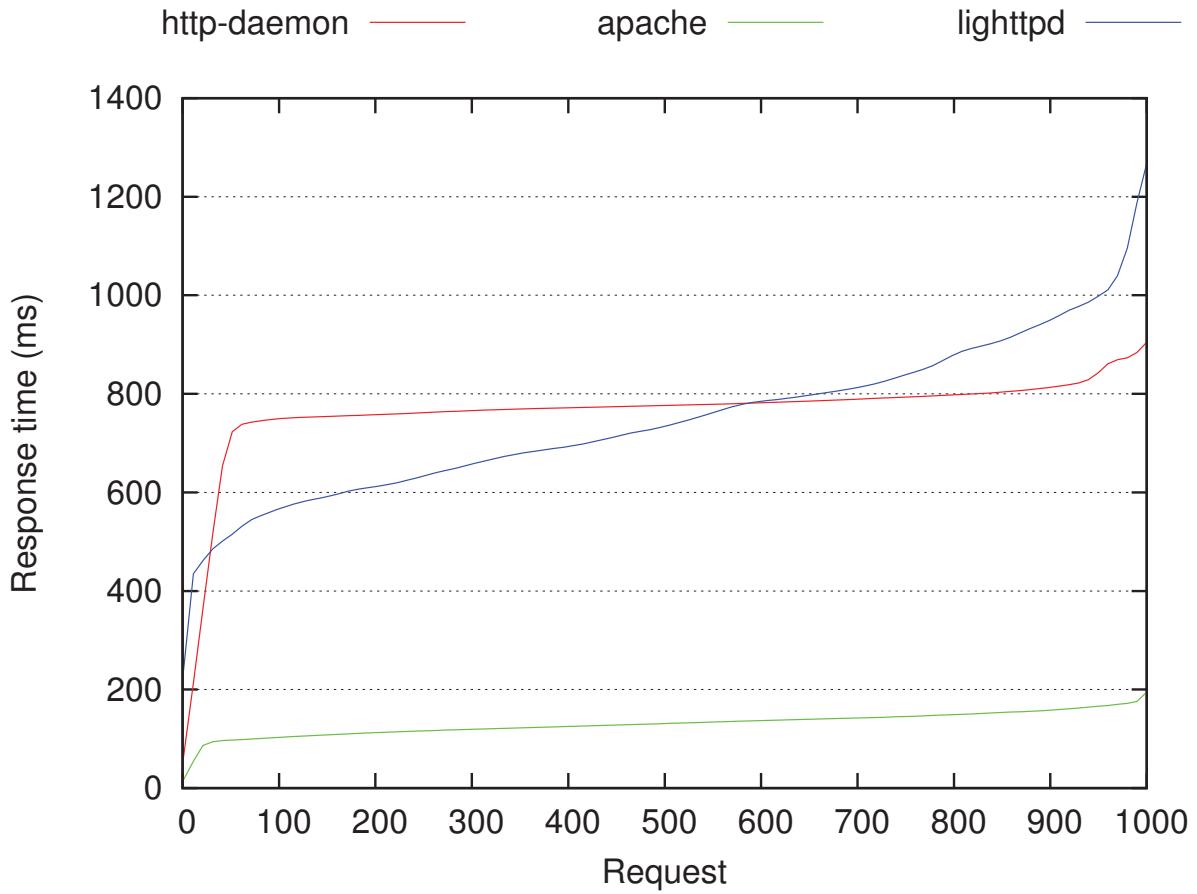


Figure 5: Requesting a CGI script

As it can be seen my daemon quickly chokes on the requests. Serving the 100th request takes 749 ms which from the user's perspective may be too much time to wait for such a simple task. Apparently after the 600th request my daemon's performance overtakes Lighttpd's. This may be due to the waiting queue gradually becoming empty so other requests are served faster. This could also explain the rapid growth of the response time at the start of the request, the worker threads become occupied and further requests are queued, hence the slope on the plot.

Since on every request a process is started in order to produce a response, the time taken by this operation may be reduced by resuming to using plain C to start the CGI executable, instead of the currently used `QProcess`²⁰, this avoids the overhead added by calling another abstraction layer (i.e. Qt).

There is also a more powerful solution, FastCGI²¹. This protocol is supported out-of-the-box in the well known daemons [10], but unfortunately my HTTP daemon doesn't implement it at the moment, although it can be easily integrated by a plugin. Other advantages offered by this approach are that the applications running on a web server may be distributed, so they are not hosted on the same machines where the HTTP daemons run. Another advantage is that they may be kept in memory, which removes altogether the overhead of process creation described

²⁰<http://qt-project.org/doc/qt-5.0/qtcore/qprocess.html>

²¹<http://www.fastcgi.com>

earlier [11].

4.3 Requesting a big file

The last test provides some insights on how the daemon performs when a big file is requested, which may be thought of as requesting a file from a file sharing website. The results of this benchmark are graphically represented in Figure 6.

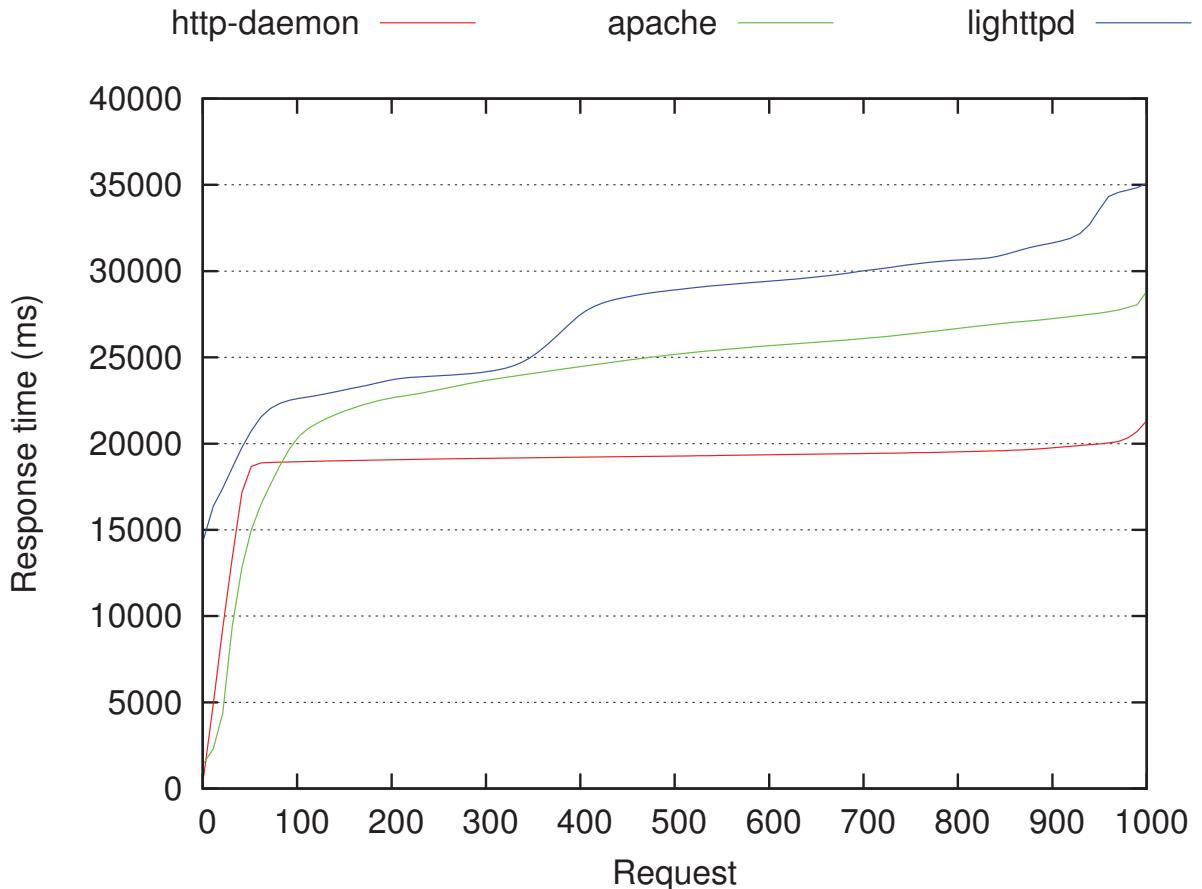


Figure 6: Requesting a big file

It's easy to see that at the beginning the performance is poorer compared to Apache, but afterwards the response time is almost constant across requests and significantly lower than that of the other two daemons, which is a very good indicator. The advantage of my daemon is that the requested file is mapped into memory which brings some speed-ups for big files, but not for small ones²². For small files the memory mapping is just an overhead, because it causes fragmentation and those files can be sent on the network as they are read.

As stated before in Section 4.2 performance may be increased if relying on plain C to do the memory mapping instead of the currently used `QFileDevice::map()`²³. Although this would

²² \simeq 5 MB

²³<https://qt-project.org/doc/qt-5.0/qtcore/qfiledevice.html#map>

stir from the goal of the daemon, ease of understanding of a simple code base, which is why the Qt framework was chosen. In this project performance was not aimed at from the beginning, either.

“Premature optimization is the root of all evil” - Folklore [12]

Without sacrificing code readability the response time can be increased by tweaking the number of kilobytes to send at once on the network, this is done via the `chunksize` configuration directive of the plugin in `httpdaemon.conf`, unfortunately this affects files of all sizes.

Without any optimization in place the response times when requesting a small file may be seen in Figure 7.

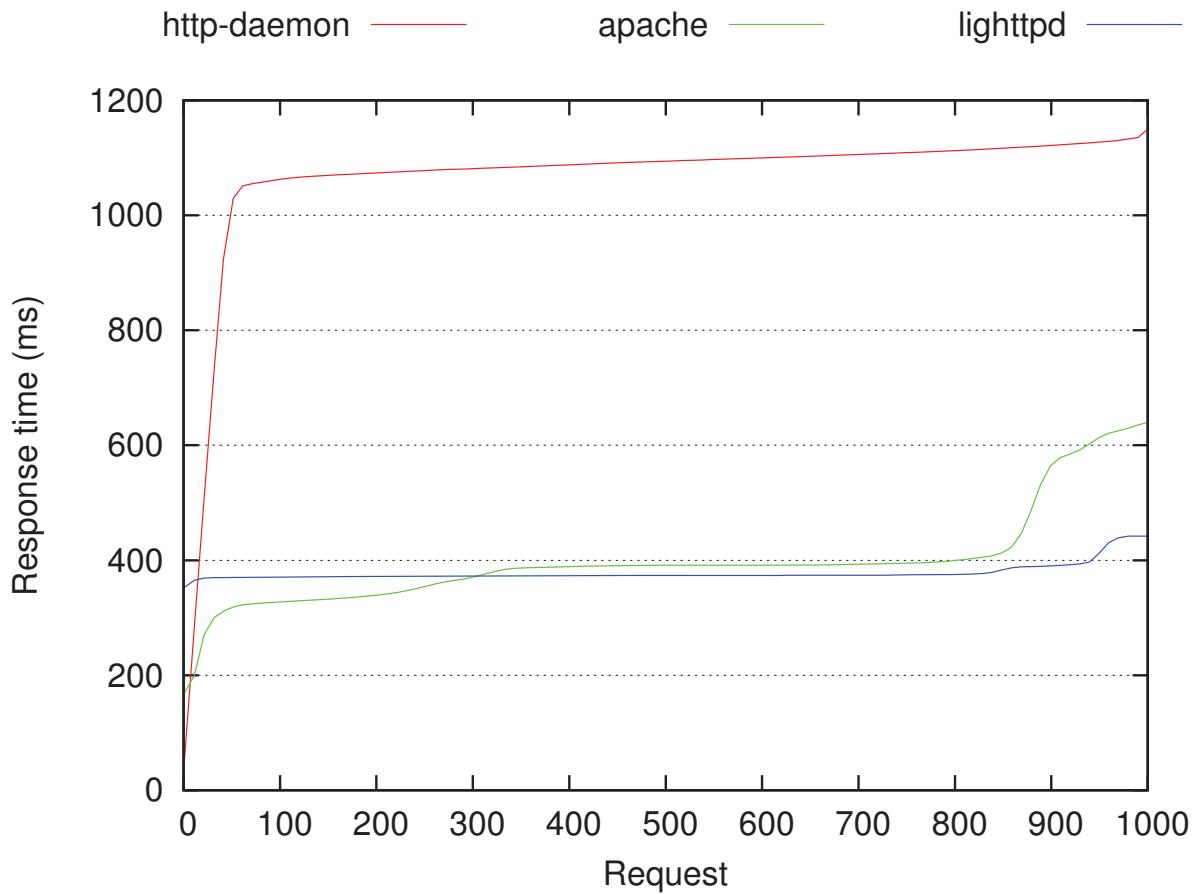


Figure 7: Requesting a small file

What's interesting in these plots as well as the one in the previous benchmark (Section 4.2) is that my daemon seems to be faster when there are only a few requests, i.e. at the start of the test, because the worker threads are not fully occupied yet, after they become occupied the performance decreases as a waiting queue builds up.

5 Conclusions and future work

The HTTP daemon presented here is just a proof of concept, with a more didactic inclination than a practical one, because it lacks several qualities needed in production environments and because it was aimed at introducing programmers to HTTP daemons from a high level perspective. It introduces programmers to the protocols involved in developing such an application: HTTP/1.0 and HTTP/1.1 as well as CGI with the added advantages of having a small and easy to read code base, being open-source, cross-platform and easy to extend by implementing plugins.

Considering the low compliance with the HTTP/1.1 standard [13] (and partial compliance with HTTP/1.0) the first step that should be taken when implementing the next version is extending the core to allow HTTP/1.1 communication and features.

Similarly with Apache's `-f` command line option my daemon should have a way of defining the configuration file path at start-up, right now it is specified by defining a C macro at compilation time.

Similarly to both Apache and Lighttpd the daemon's configuration file syntax could be extended to allow conditional configuration and configuration file inclusion, this would help making the configuration modular, every plugin could then be shipped with its own configuration file that can be included in the main `httpdaemon.conf` file.

Also the speed of the daemon can be improved as stated in Section 4.2, by implementing the newer FastCGI protocol instead of the older CGI, this wouldn't affect the readability of the code either.

Although the daemon does run on Windows systems, it will not become a daemon after being started, this should be done in a future version of the application.

References

- [1] ITU, *Key ICT indicators for developed and developing countries and the world (totals and penetration rates)*, www.itu.int/en/ITU-D/Statistics/Documents/statistics/2013/ITU_Key_2005-2013_ICT_data.xls, 2013.
- [2] The Apache Software Foundation, *The Apache HTTP Server Project*, <https://httpd.apache.org/>, 2014
- [3] Wikipedia, *Apache Portable Runtime*, https://en.wikipedia.org/wiki/Apache_Portable_Runtime#Similar_projects, 2014
- [4] Berners-Lee T., et al., *Hypertext Transfer Protocol – HTTP/1.0, RFC 1945*, <http://www.w3.org/Protocols/rfc1945/rfc1945>, 1996
- [5] Qt Project, *Signals & Slots*, <http://qt-project.org/doc/qt-5/signalsandslots.html>, 2013
- [6] Wikipedia, *Daemon (computing)*, https://en.wikipedia.org/wiki/Daemon_%28computing%29#Creation, 2014
- [7] Qt Project, *QThread Class*, <http://qt-project.org/doc/qt-5/qthread.html#idealThreadCount>, 2013
- [8] Robinson D., et al., *The Common Gateway Interface (CGI) Version 1.1, RFC 3875*, <http://www.ietf.org/rfc/rfc3875>, 2004
- [9] Facebook, *Key Facts*, <https://newsroom.fb.com/Key-Facts>, 2013

- [10] Open Market, Rob Saccoccio & others, *Servers Supporting FastCGI*, <http://www.fastcgi.com/drupal/node/3>
- [11] Open Market, Rob Saccoccio & others, *FastCGI: A High-Performance Web Server Interface*, <http://www.fastcgi.com/drupal/node/6?q=node/15>, 1996
- [12] Gerwitz Hans, *Premature optimization is the root of all evil*, <http://hans.gerwitz.com/2004/08/12/premature-optimization-is-the-root-of-all-evil.html>, 2004
- [13] Fielding R., et al., *Hypertext Transfer Protocol – HTTP/1.1, RFC 2616*, <https://www.ietf.org/rfc/rfc2616.txt>, 1999
- [14] lighty team, *Optimizing Lighty for High-concurrent, Large-file Downloads*, <http://blog.lighttpd.net/articles/2005/11/11/optimizing-lighty-for-high-concurrent-large-file-downloads/>, 2005

Barbu Paul - Gheorghe
 “Lucian Blaga” University of Sibiu
 Department of Computer Science and Electrical Engineering
 10, Victoriei Bd., Sibiu, 550024, România
 ROMÂNIA
 E-mail: barbu.paul.gheorghe@gmail.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Kinect game

Robert Sandica, Petrica Bota

Teacher Coordinator: Prof. PhD. Dana Simian

Abstract

Motion Ball is an interactive, 2D, Kinect game with the purpose of making people exercise while having fun. The game sparks your interest with its simplicity and keeps you entertained with its competitive nature.

Motion Ball is programmed in Visual Studio 2013 on the XNA 4.0 framework. It implements libraries offered by the .Net framework with motion trackers conveyed by the Kinect sensor in order to usher a new experience.

Though it is presented on a simple format as of now, we strive to implement new features and possibly even 3D characters in the near future.

1. Introduction and description

We live in a world where gaming is not only a part of our society but of us as well. The memories of playing hide and seek are being replaced with the ones of playing your favorite FPS(First Person Shooter) or RTS(Real Time Strategy) game. Actually, there isn't even a correct definition for the word „gamer”. Who is a gamer? The 14 year old teenager who plays Call of Duty for hours at an end? Or your aunt. who plays bejeweled on her free time.

Even though, at a first glance it seems that online games are bringing us closer to each other, this couldn't be further from the truth. Despite the fact that our grasp reaches hundreds of people and that there are a multitudine of online games, we are beginning to feel lonelier and lonelier.

Therefore, we concentrated our efforts on technologies that bring people together. We choose Kinect due to its built in tracker system as well as its recent popularity, spread by the release of the Xbox One.

What is more, gamers are usually labeled as being lazzy and doing little to no exercise. With Motion Ball, we combine the love for gaming together with body movements in order to create a healthy, fun as well as competitive game .

The principle of the game is to offer an interactive experience, not only with the Kinect, but with your friends as well. The game draws you in with its basic design and keeps you motivated with its interactive controls.

1.1 About the Kinect

The Kinect is currently the hardware that provides developers with the greatest opportunities for innovative programs - both games and "serious" artificial applications.

Microsoft's Kinect is a "controller-free gaming and entertainment experience" and is commonly sold bundled with the Xbox.

The depth sensing hardware is the first feature that makes the Kinect special but without software it is just so much clever optics and processing. The algorithm uses what's called a decision forest (a collection of decision trees) trained using thousands of sample datasets over even more thousands of hours of cluster compute hours. This provides advantages such as not requiring a calibration pose prior to use; not causing problems if pose varies greatly over a small amount of time; and efficient working with the type of visual data that Kinect provides.

Even if Kinect games are a relatively new genre of games there isn't a shortage of them. From simple 2D games that are meant as a recreational tool, to elaborated 3D sagas that put your mind to the test. Microsoft released Kinect software development kit for Windows 7 on June 16, 2011. This SDK was meant to allow developers to write Kinect apps in C++/CLI, C#, or Visual Basic .NET. Now, with the release of the Xbox One, the number of

Kinect games are bound to increase exponentially in the following years..



Fig 1. Kinect for the Xbox 360

Source: <http://lrla.com/wp-content/uploads/2010/11/Kinecttechnology.jpg>

1.2 Project inspiration and goals

We choose game design as a topic due to the fact that we are both gamers.

We grew up with games around us and we wanted to discover how they are made along with creating our own. We started learning Pascal, C, C++ and C# in order to achieve this goal. This project is yet another challenge, combining the entertainment of the Kinect with the puzzles of C# and XNA in one unique experience.

1.2.1 The three main goals of the project:

1. Creating a simple game that appeals to all types of people, indifferent of age.
2. Combining exercise with gaming.
3. Learning more about game development .

1.2.2 Secondary objectives of the above:

- a) Learn how to create a simple 2D environment.
- b) A better understanding of the C# language.
- c) Learn how to use the XNA libraries.
- d) Learn how to use the Kinect SDK.
- e) To prove ourselves that we can.
- f) Create good collision vectors.
- g) Promote the finished product.

2. Used technologies

2.1 XNA 4.0 framework

Computer game creation typically requires a large amount of time and coding, making the implementation of large-scale games difficult for small design teams and individual programmers. To make this easier, many software companies release programming frameworks to reduce the workload required. One such tool set is Microsoft's XNA Framework, designed for use with Microsoft's various platforms.

The Microsoft XNA Framework is a set of programming tools designed for use with Microsoft's .NET Framework. It consists of code libraries and program APIs to reduce the amount of coding required for common game-related tasks. The intent of the XNA Framework is to allow casual and independent programmers to focus more on the content of their games without having to spend a significant amount of time crafting a game engine and graphics framework beforehand.

We used Visual Studio 2013 and the XNA framework due to the fact that it has many implemented libraries, methods and it represents a friendly environment for the programmer.

Fig2 Source: http://jharbour.com/wordpress/wp-content/uploads/2010/12/xna_logo.jpg



2.2 .NET framework

.NET Framework (pronounced *dot net*) is a software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large library and provides language interoperability (each language can use code written in other languages) across several programming languages. Programs written for .NET Framework execute in a software environment (as contrasted

Fig 3. Source: http://jharbour.com/wordpress/wp-content/uploads/2010/12/xna_logo.jpg



to hardware environment). Because computer systems commonly require interaction between newer and older applications, .NET Framework provides means to access functionality implemented in newer and older programs that execute outside .NET environment.

The Framework Class Library (FCL) is a library of functionality available to all languages using .NET Framework. The FCL provides classes that encapsulate a number of common functions, including file reading and writing, graphic rendering, database interaction, XML document manipulation, and so on. It consists of classes, interfaces of reusable types that integrates with CLR (Common Language Runtime).

2.3 Kinect SDK

The Kinect for Windows SDK, created by Microsoft Research, is a programming toolkit for application developers. It enables easy access to the capabilities offered by the Microsoft Kinect device connected to computers running the Windows 7 (or higher) operating system. The Kinect for Windows SDK beta includes drivers, rich APIs for raw sensor streams and natural user interfaces, installation documents, and resource materials. It provides Kinect capabilities to developers who build applications with C++, C#, or Visual Basic by using Microsoft Visual Studio.

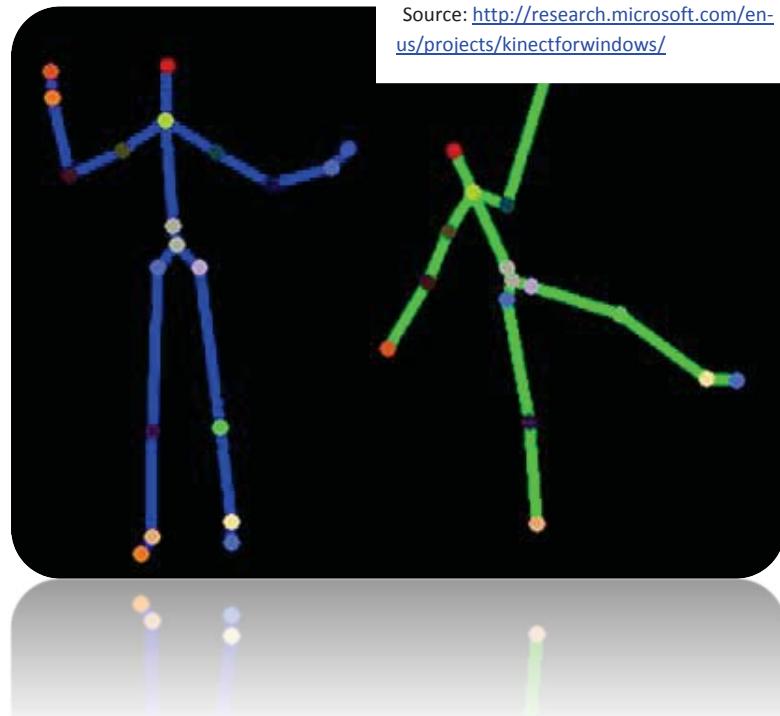
Features

The SDK includes the following features:

- **Raw sensor streams** Access to raw data streams from the depth sensor, color camera sensor, and four-element microphone array enables developers to build upon the low-level streams that are generated by the Kinect sensor.
- **Skeletal tracking** The capability to track the skeleton image of one or two people moving within the Kinect field of view make it easy to create gesture-driven applications.
- **Advanced audio capabilities** Audio processing capabilities include sophisticated acoustic noise suppression and echo cancellation, beam formation to identify the current sound source, and integration with the Windows speech recognition API.

Fig 4. Skeletal tracking

Source: <http://research.microsoft.com/en-us/projects/kinectforwindows/>



- **Sample code and documentation** The SDK includes more than 100 pages of technical documentation. In addition to built-in help files, the documentation includes detailed walkthroughs for most samples provided with the SDK.
- **Easy installation** The SDK installs quickly, requires no complex configuration, and the complete installer size is less than 100 MB. Developers can get up and running in just a few minutes with a standard standalone Kinect sensor unit (widely available at retail outlets).

3. A peek inside

Motion Ball is as user friendly as it can get; all you have to do is launch the application and enter the Kinect's area of vision. Once you complete these easy steps, the game automatically sets one hand as the attacker and the other as the goal keeper. The game is very intuitive and you quickly adapt to the controls.

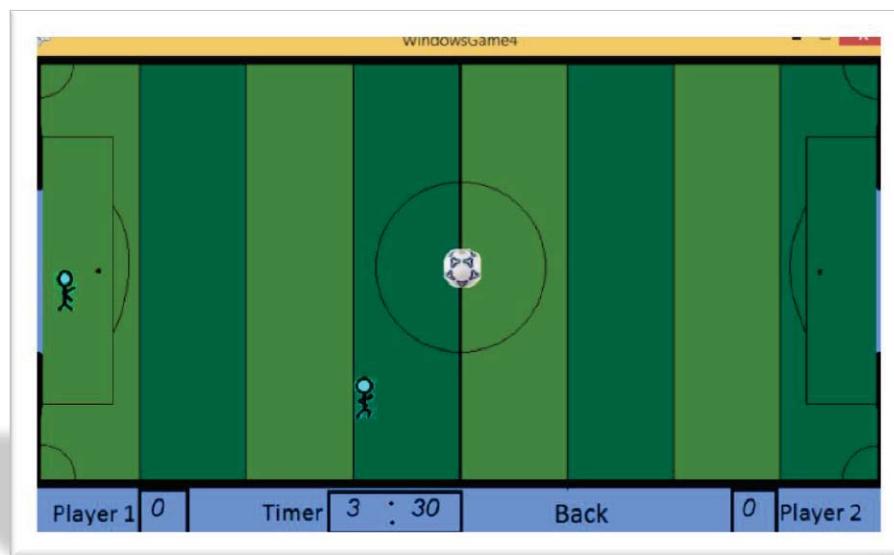


Fig 5. Simple game design

The code itself is easy to understand thanks to the XNA and .NET framework. Although we had some issues during the programming phase, we were able to overcome and learn from them

```
public void Miscare_minge()
{
    if (incheieturi.Count > 0)
    {
        Coliziune_player1(incheieturi[0], pozitie_minge);
        Coliziune_player1(pozitie_portar1, pozitie_minge);
    }
    if (incheieturi.Count > 2)
    {
        Coliziune_player2(pozitie_portar2, pozitie_minge);
        Coliziune_player2(incheieturi[3], pozitie_minge);
    }
}
```

Fig 6. Short example of code

4. Conclusions and future development

To sum up, Motion Ball is our contribution to the game industry, small as it may be in comparison. Though we are the beginning of our yellow brick road, we strive to make the best programs that we are capable of.

4.1 As future developments, we would like to implement the following:

- 3D skeletons
- More joint trackers
- Face recognition
- Implementing the game without the use of a Kinect

References:

- [1] Video tutorial - <https://www.youtube.com/user/CodingMadeEasy>
- [2] Download Kinect SDK - <http://www.microsoft.com/en-us/download/details.aspx?id=40278>
- [3] OOP - <http://www.pluralsight.com/training>
- [4] Article about Kinect - http://www.i-programmer.info/babbages-bag/2003-kinect-the-technology_.html
- [5] Paper about Kinect -<http://research.microsoft.com/pubs/145347/BodyPartRecognition.pdf>
- [6] XNA article - <http://yourbusiness.azcentral.com/microsoft-xna-framework-20700.html>
- [7] Article about SDK - <http://research.microsoft.com/en-us/projects/kinectforwindows/>
- [8] Paper structure -http://conferences.ulbsibiu.ro/icdd/2012/files/proceedings_ICDD2012.pdf

Robert Sandica
„Lucian Blaga” University
Faculty of Sciences
Str. Dr. I. Rațiu, Nr.5-7, Sibiu
Romania
Email: novaspace_robi@yahoo.com

Petrica Bota
„Lucian Blaga” University
Faculty of Sciences
Str. Dr. I. Rațiu, Nr.5-7, Sibiu
Romania
E-mail: bota.petrica2@yahoo.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Student+ - an Android Application

**Andrei Călbăjos, Bărbos Adrian, Kim Velker
Teacher Coordinator: Dr.Camelia-M. Pintea**

Abstract

The current paper illustrates a new Android application designed for student groups. Student+ is the name of the newly introduced application. The specific components and the way to use it are explained. Significant images are showing the user-friendly interface. The application could be further enlarged including specific applications for companies.

1 Introduction

Student organizations are a great place to meet new friends, stay active on campus and learn a little more about something which interests you. It is important to develop easy to use software to help students in their everyday basis. Tablets and Smartphones are frequently used nowadays by almost everyone and especially by students. Our application will help the users to keep up to date with their daily program, stay in touch with other users and be notified about future activities, and other useful information.

2 Android and Java

Android is an operating system based on Linux. It is designed for touch screen devices and allows users to create shorcuts to applications and widgets for displaying live content (e.g. emails, weather informations) on the home screen; some applications can send notifications such as new emails or RSS feedbacks. From year 2012 Android is the most popular operating system for mobile devices. There are many Android applications available on app stores such as Google Play, Amazon Appstore or other websites with APK applications files.

The Android applications are developed in general in the Java programming language. It uses the development kit, SDK, with many software libraries, including specific libraries for Android, a handset emulator and a debugger.

In this specific application we are using Google Android library, GCM (google cloud messaging), wich allows us to to send push notifications from a php based webservice. Because it is an online platform, we use several php based webservices for our interaction with a remote SQL Database where we store all our data.

3 Student+ Project

The current Student+ project was created from scratch. The Android Application part is created in the Java programming language, using google's Android Development Kit (SDK) and the Server Part that allows us to interact with Remote SQL Databases, by using PHP based webservices.

As mentioned, the Webservices we use to connect to the SQL Databases are PHP based, in which we send our data via POST. Some of the php commands we use are:

- INSERT INTO – inserts data into the selected table (example: ("INSERT INTO tablename(1column, 2column, 3column) VALUE('\$1value', '\$2value', '\$3value')"))
- CREATE TABLE – creates a table in the selected database (example: ("CREATE TABLE tablename(column1 type-of-data(value), column1 type-of-data(value), column1 type-of-data(value))"))
- SELECT * FROM – selects all the data from the selected table (example ("SELECT * FROM tablename"))

The webservice send us a JSON Object the contains an array of the requested data, which we store in the application and then list that data.

From a java point of view, the application is divided into 10 packages, as follows:

- the main package(contains the main activities, along with the classes they require and the notification controller)
- admin package(contains all of the admin modules)
- events(event activities and classes)
- forum(all the classes needed in order to add a new topic, open one, send and receive posts)
- imgload(a package used by most classes to download, store in cache and display images)
- library(contains classes needed to parse information, store data in cache, call, send email, send sms, a database, etc.)
- news(contains two activities and an adapter)
- profile(activities which allow the user to view his or her profile and change the password)
- followers(activities that allow a user to manage, contact and add followers)
- users(activities that allow a user to manage, contact and add other users)

Student+ uses only one library, namely Google Cloud Messaging(or GCM). This library provides easy access to notifications, which can be sent either from a web-interface we developed, or directly from the admin module of the application.

One of the most used classes in this application is the JSONParser class. By providing a minimal number of parameters, three to be exact, this class is able to retrieve JSONArrays and JSONObjects. In combination with several classes we created that extend AsyncTask, we provide a smooth and fast way to retrieve information, because all the work is done in the background, not affecting the main thread. This creates a stress-free environment for the user, making it easier to navigate through all the activities.

How does Student+ project work? There are many ways to manage the content of the application. We developed an administration module in the application and also a web-based administrator menu. The application is optimized for fast data parsing without slowing down the users of Smartphones or

Tablets. It is provided a user friendly interface so the users will find without much effort the information they need.

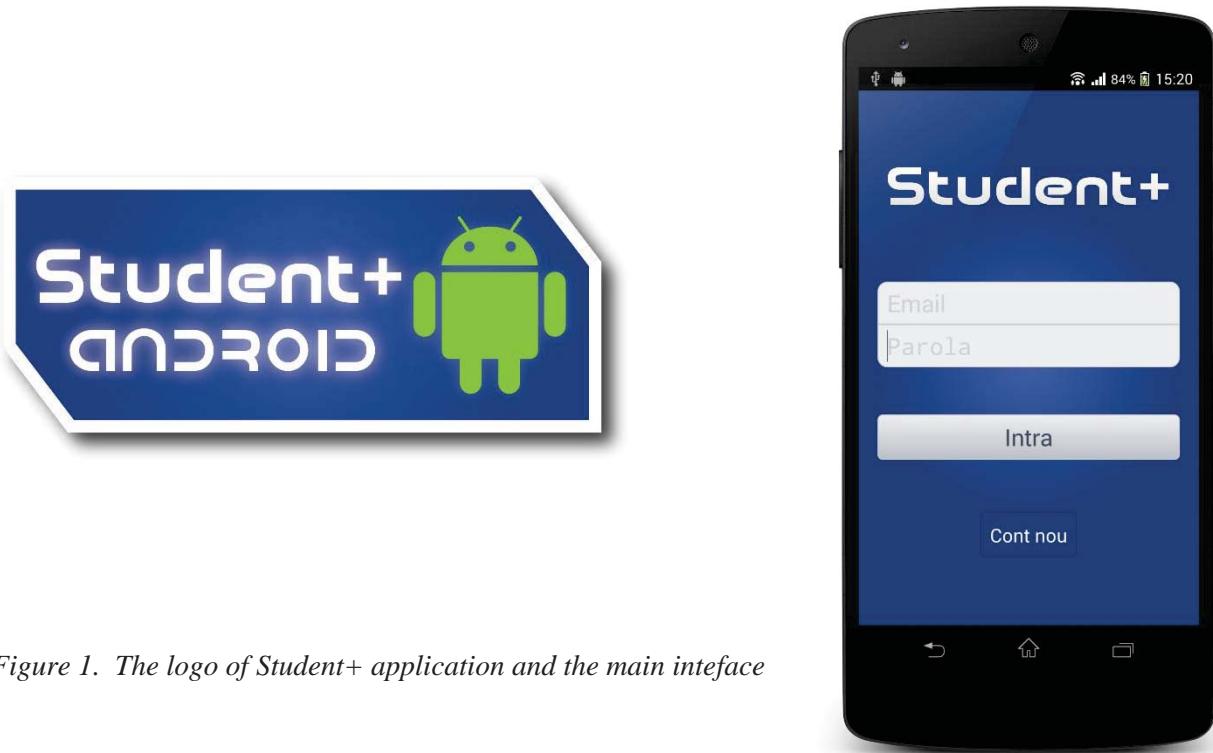


Figure 1. The logo of Student+ application and the main interface

The main functions of Student+ are following:

- Update the database, for example adding, modifying, deleting data
- View and/or modifying information about upcomming events, news, members info etc.
- Notifications, for example when the group administrator wants to notify the members or a future event based on JSON parsed data.
- Contacting each member via email, phone and sms directly from the application interface
- Sending multiple emails and/or sms` to all the followers, while the user also has the option to check or uncheck each and every one of them
- Group forum function

The following are some of the Student+ application advantages and original features:

- The application is easy to install, as any *.apk file.
- It has a build in administration panel
- The content is very easy to change
- Secured login , MD5 encrypted password

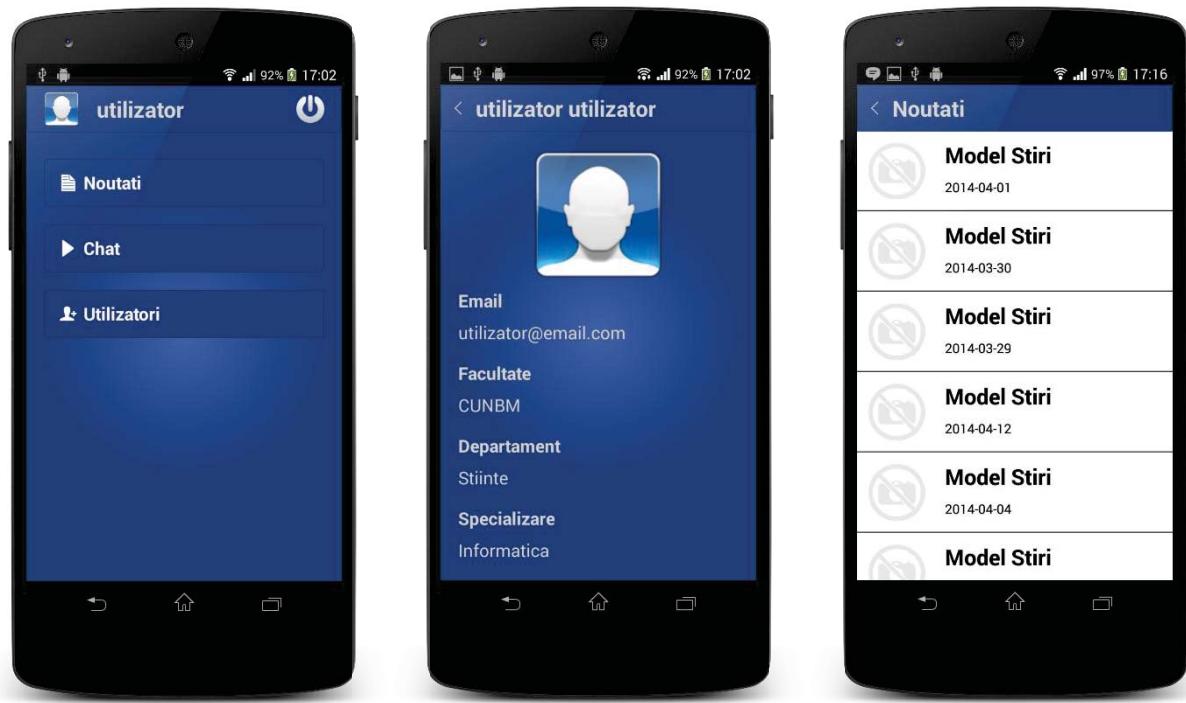


Figure 2. Illustrative sequences of Student+ Android application

4 Conclusions and Further Work

The paper describes a newly Android application: *Student+*. It is developed as a OOP-application in Java and also uses SQL/PHP. One of the many facilities of the current application is that it can be used to help group members keep in touch and be notified when new information is available.

For now the application is used by student groups. It could be improved and used by employers to communicate more efficiently with the current students, their future employees. Some updates in the nearby future will expand the application's usage to teachers, bigger companies, members of certain conferences, political organizations, and so forth.

References

- [1] S. Tanasa, C. Olaru, *Java de la 0 la expert*, Ed.Polirom 2011
- [2] R.C. Martin, *Clean Code: A Handbook of Agile Software Craftsmanship*, Prentice Hall 2008
- [3] J. Friesen, *Learn Java for Android Development*, Apress 2010
- [4] Wikipedia

Andrei CĂLBĂJOS
Technical University Cluj-Napoca
CUNBM – Faculty of Science
Department – Computer science
76 Victoriei
ROMÂNIA
andrei.calbajos@tencadis.ro

Adrian BĂRBOS
Technical University Cluj-Napoca
CUNBM – Faculty of Science
Department – Computer science
76 Victoriei
ROMÂNIA
adrian.barbos@tencadis.ro

Kim VELKER
Technical University Cluj-Napoca
CUNBM – Faculty of Science
Department – Computer science
76 Victoriei
ROMÂNIA
kim.velker@tencadis.ro

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Drive Me-A easy way to orientation

**Cosmin Constantin Calin , Diana Maria Duca
Teacher Coordinator: Dana Simian, Daniel Hunyadi**

Abstract

The aim of this paper is to presentate an original Windows8 app, designed to support the tourists to find the road to tourist attractions, restaurants, shops and exchange offices in a new city. With a simple click you can see the distance and the road map to a selected destination.

The originality of this application is that in addition to guidance implementation and exchange option, a beginner programmer can use it to learn how to create an application that contain these features.

1 Introduction

In the context of current global development, tourism is an important resource for any country. Romania is a country with many tourist attractions and a rich history.

The purpose of the application made,presented in this paper, is to help visitors by providing facilities of orientation in the cities of Romania , with the ability to determine points of interest, and in the money exchange process.

Another purpose of this work was to demonstrate that we are able to design such an application, as a starting point for other more complex applications, and to teach others how to build such an application.

The application can run on the desktop or tablet and can be used by tourists for documentation.

The rest of the paper is organized in following: The second section contains details about the technologies used to build the app and about the design of the app, the third section contains details about the future improuvments, and the last section contains the references.

2 Design and technoligies

2.1 Technologies

Drive Me is a Metro app. You can access it from any device with Windows8 operating system. This app was written in Visual Studio 2012 using C# and XAML.

“C# is an object-oriented programming language. It was designed by Microsoft. C# is based on C++ and contains features similar to those of Java but more from C++. Microsoft Silverlight is a free web-browser plug-in that enables interactive media experiences, rich business applications and immersive mobile apps. Visual Studio contains the .NetFramework, a class library which is very useful for the developers to build applications with graphic interface” [1].

2.2 Design

DriveMe is very easy to use and intuitive for the user.

The application is optimized for both orientations, Portrait and Landscape.

DriveMe consists of two parts :

1. Converter - the algorithm that return the currency from BNR.
2. Places finder –it finds all the places round me and show me the routes on a map.

2.2.1 Convertor

This currency converter takes the exchange rate from the website of the National Bank (BCR). Just choose your currency from a list of four national currencies and a window will appear with the exchange rate for the chosen currency in the last 10 days.

Observing what happened with a currency in relation with the national currency, the user would know if the current rate is a low or a high one and decide if this is the moment he wants to make the exchange.

Due to the fact that on the National Bank web site you can find a xml file with the currency, we used that file to extract data that interest us.

The code for extract data from xml file is the following:

```
private async void coinValue_Click(object sender, RoutedEventArgs e)
{
    resultCoinValue.Text = "";
    string choice = coinComboBox.SelectedItem.ToString().ToLower();

    var bnr = new BnrXml();
    var list = await bnr.ReturnCoinValue(choice);

    foreach (var item in list)
    {
        resultCoinValue.Text += item + "\r\n";
    }
}
```

2.2.2 The place finder

We designed this app to help Romania's visitors to orient in the largest cities. You can choose from 5 cities: Sibiu, Bucuresti, Cluj, Iasi and Timisoara.

The aim of the place finder is to determine the route to the selected destination.

After choosing a city you will be redirected to a page containing all the locations around you in a radius of 100 km. From that list you can choose the location you want to visit and pressing GuideMe a map will display the road map from the current position to the selected location.

The sights that are available in the application are taken from an online database: virtualearth.

For the implementation we used a bing map, and from Microsoft web site we received a bing key that allows us to implement the gps functionality.

We had also implemented a functionality that allow us to calculate the distance from the current position to the selected location.

The method to calculate the distance is:

```
private double CalculateDistance(double prevLat, double prevLong, double currLat, double currLong)
{
    const double degreesToRadians = (Math.PI/180.0);
    const double earthRadius = 6371; // kilometers

    // convert latitude and longitude values to radians
    var prevRadLat = prevLat*degreesToRadians;
    var prevRadLong = prevLong*degreesToRadians;
    var currRadLat = currLat*degreesToRadians;
    var currRadLong = currLong*degreesToRadians;

    // calculate radian delta between each position.
    var radDeltaLat = currRadLat - prevRadLat;
    var radDeltaLong = currRadLong - prevRadLong;

    // calculate distance
    var expr1 = (Math.Sin(radDeltaLat/2.0)*
        Math.Sin(radDeltaLat/2.0)) +
        (Math.Cos(prevRadLat)*
        Math.Cos(currRadLat)*
        Math.Sin(radDeltaLong/2.0)*
        Math.Sin(radDeltaLong/2.0));

    var expr2 = 2.0*Math.Atan2(Math.Sqrt(expr1),
        Math.Sqrt(1 - expr1));

    var distance = (earthRadius*expr2);
    return distance; //return distance as km
}
```

2.2.3 The user interface

Drive Me is a really easy-to-use application with a friendly interface, appropriate for users of all ages.

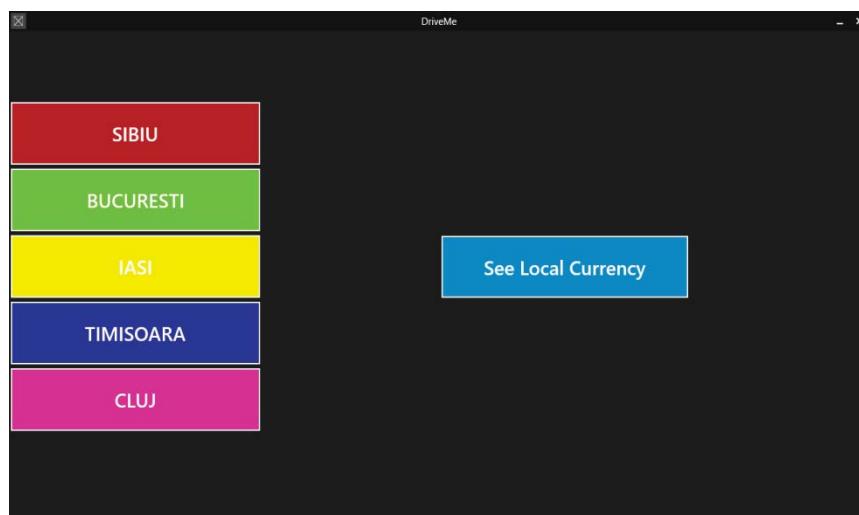


Fig. 1- Main Page

After you selected a city a new window with a map and a combo box which contains sights from the selected city will be displayed.

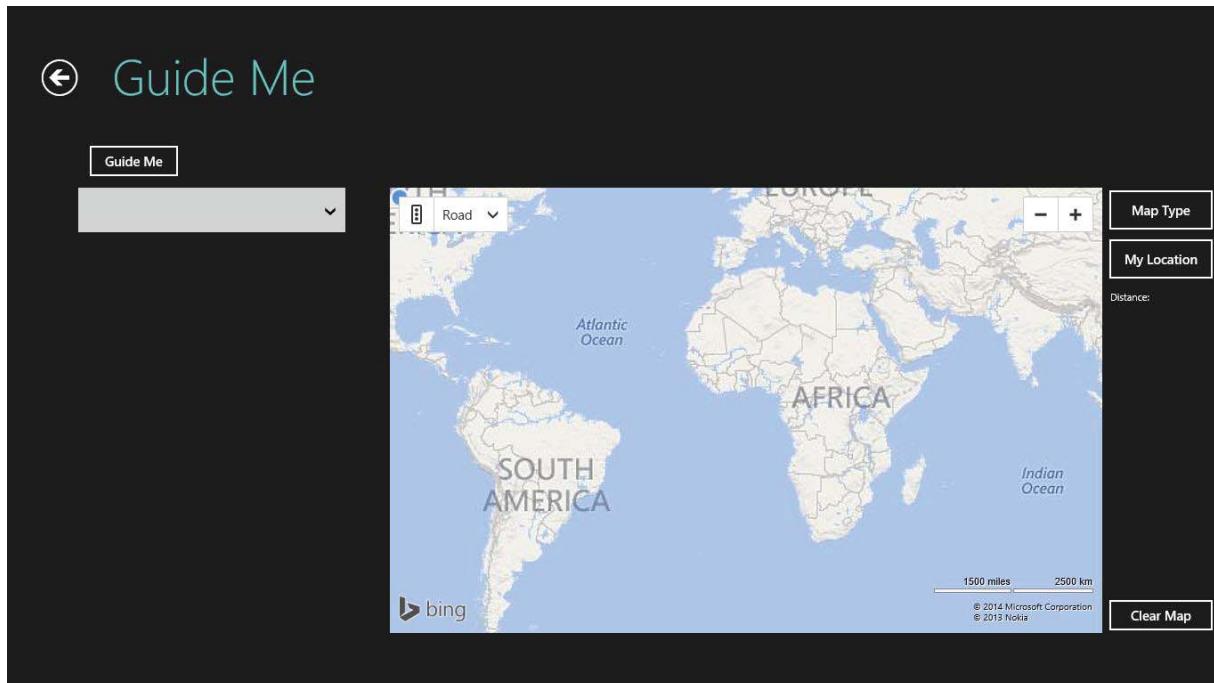


Fig. 2. BasicPage1.xml

You can change the map type. There are three options : bird's eye, road or aerial.

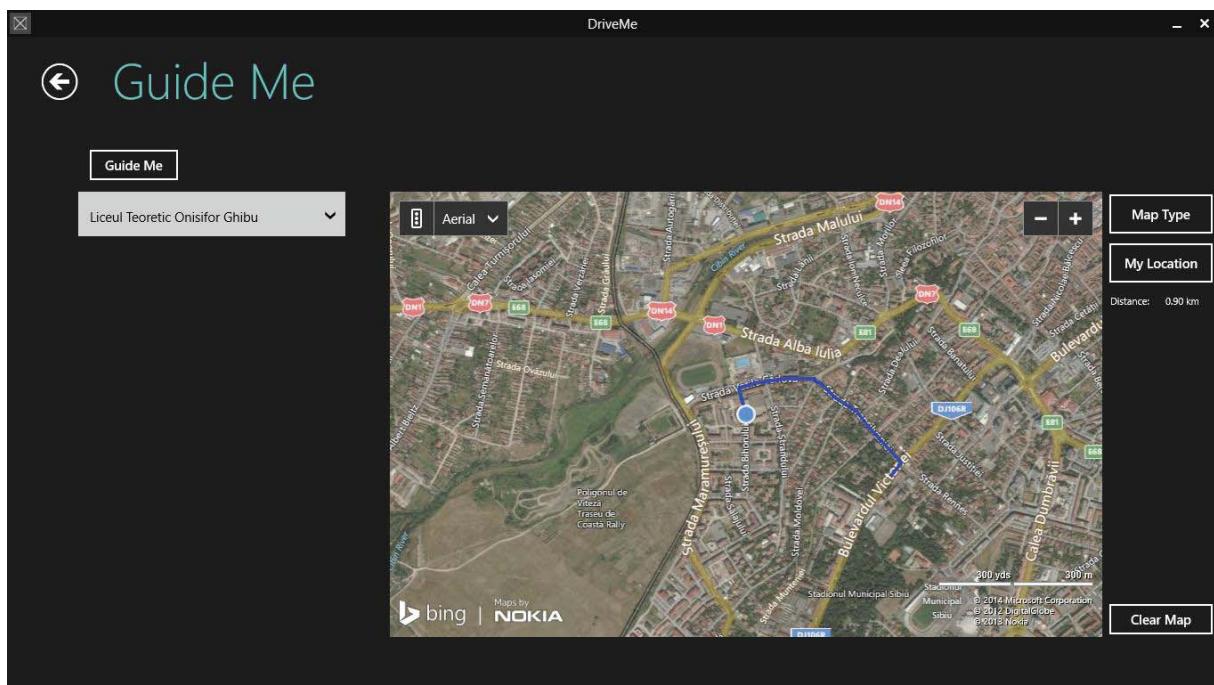


Fig 3. The road map to “Liceul teoretic Onisifor Ghibu”

The other option (to see the currency from BNR for the last 10 days) looks like the following picture.

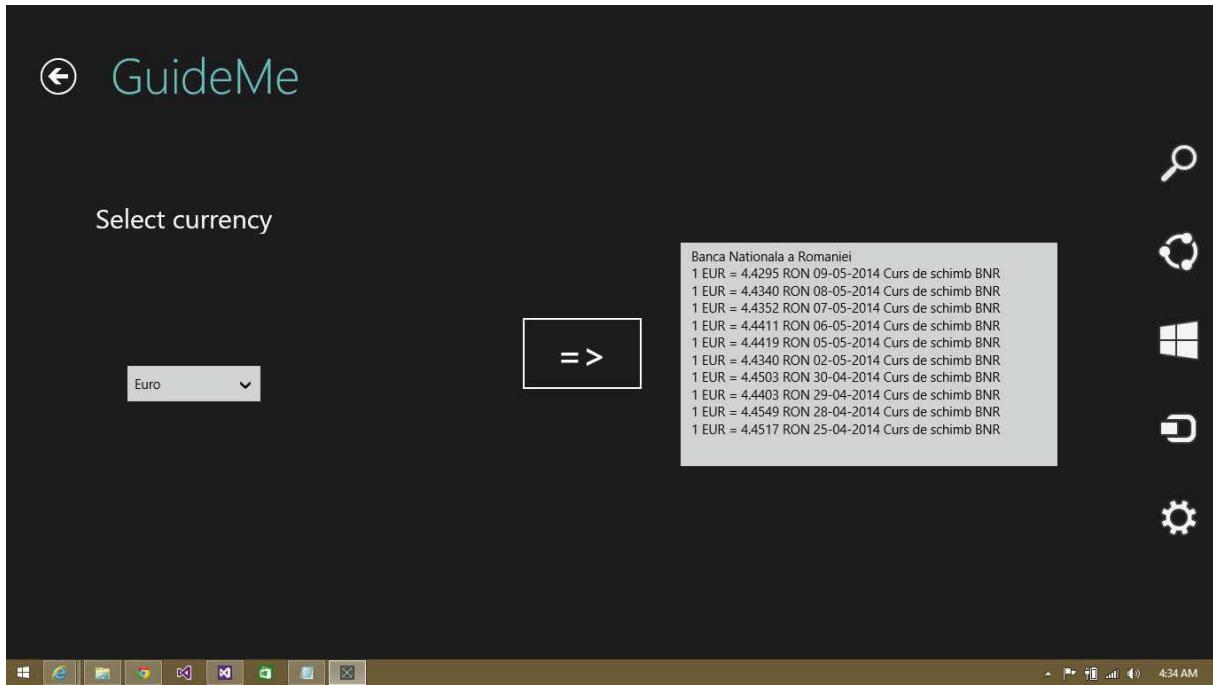


Fig 4. The currency of EURO for the last 10 days

2.3 The “How to...” option

There are a lot of applications on the market with a GPS option implemented. These applications are used only for guidance. In contrast, Drive Me is designed for developers to.

We have thought it is very useful for a beginner developer to have access to the application code sequences for seeing how different options can be implemented.

In this sense, just by clicking the "How to ..." button, from the main page, you can see screenshots with pieces of application source code that shows how to implement a map, how to customize it, how to access a key for a Bing map from Microsoft, how to access a xml page and how to retrieve the code from this page.

With this knowledge, you can personalize the application by adding new cities to search for routes in your favorite places.

3 Conclusion

Drive Me, is a very useful application because with its help, every single person could make the best choices for their money. It is useful because of the option to calculate the distance between the users position and any location he is looking for, and because of the option to see the road map to the selected item.

The originality of the application is that besides guidance and options exchange, the application offers the opportunity to learn how to configure a bing map and how to calculate distance between two points. The advantage of this utility is that the user can learn how to add a new city besides the existing five that apply the same operation.

In the near future we want to make it available for other countries and we want to implement the menu in different languages. Also we want to publish this app on Windows Store.

4 References

- [1] <http://searchwindevelopment.techtarget.com/definition/C> -accessed in 5.05.2014
- [2] Ciprian Adrian Rusen, *Programming Windows 8 Step by Step*.
- [3] MSDN Library, <http://msdn.microsoft.com/en-us/library>. – accessed in 5.05.2014
- [4] <http://channel9.msdn.com/Blogs/DevRadio/Microsoft-DevRadio-Developing-for-Windows-8-in-1-2-the-Time-60min-Challenge-Buidling-Cross-Platform-Windows 8 Courses>, <http://www.pluralsight-training.net/microsoft/Courses> -accesed in 6.05.2014
- [5] Puja Pramudya, *Silverlight for Windows (e-book)*

Cosmin Constantin CALIN
“Lucian Blaga” University
Faculty of Sciences, Informatics
Sibiu, 5-7, Ioan Ratiu str.
Romania
calin.cozmin@yahoo.com

Diana Maria DUCA
“Lucian Blaga” University
Faculty of Sciences, Informatics
Sibiu, 5-7, Ioan Ratiu str.
Romania
duca.diana92@yahoo.com

PCTVPlay – advanced media player

**Georgi Chavdarov
Teacher Coordinator: Assoc. Prof., PhD Katalina Grigorova**

Abstract

Millions of people worldwide are using the PC as a way for watching movies. Millions of people use their computers as a TV, but they have to control them like a computer. But now it's time for a revolution - media player, driven by remote control from a distance, with the ability to visualize the internet TV channels, movies and play music. The presented application PCTVPlay helps the users to control their computers without getting up from their place.

1 Introduction

Since the computer has become a huge source of files many people have started to use their computers for processing multimedia. Tens of thousands of people want to sit in a chair, or rest on the sofa and watch movies, videos, music and photos – but how? It is known that the hardware specialists have invented a wireless remote control for a computer, but the problem is that the management is very difficult, because the modern media players do not have keyboard shortcuts for fast navigation or some of them that have need a standard keyboard controls.

Some users playing movies for their children and grandparents. In the first case, he or she may want to limit children to watch only selected movies that can be paused or skipped. In the second case, older spectators may not have the necessary knowledge to switch one episode to another, or switch Internet TV channels.

Many users using software players like KMPlayer, which allows to manage the wireless. But here is the question - whether each remote control has the buttons like J, PgUp/Down, Ctrl or Shift?

With PCTVPlay, life becomes easier. On first time when user runs the application, it offers the opportunity to learn control buttons. When the setup is done, the application automatically redirects to Internet channels. By default, there is only one channel that serves as a demo. Users may introduce an unlimited number of addresses. They can be local - folder on computer, media files or network - IPTV, camera for video monitoring.

By itself, the program is not unique. On January 9, 2001 Microsoft realizes Windows Media Center [1]. It focuses on NetFlix streams - a service that is not available in each country and uses specially designed remote device, such as MCE Remote. Another similar product existing is GOM Remote. It also manages media files like TV remote, but unfortunately it is necessary availability of Android smart phone or tablet, which leads to unnecessary costs.

The application is written using Visual C# and is intended only for the operating system Microsoft Windows. The database used to store channels, settings and the learned keys is in Access 2003 format.

2 Switching channels

2.1 Main functionality

Switching TV channels has never been so easy before. Adding channels, browsing media files has never been done with such ease.

The presented PCTVPlay application allows operation of many media files with a sense of TV.

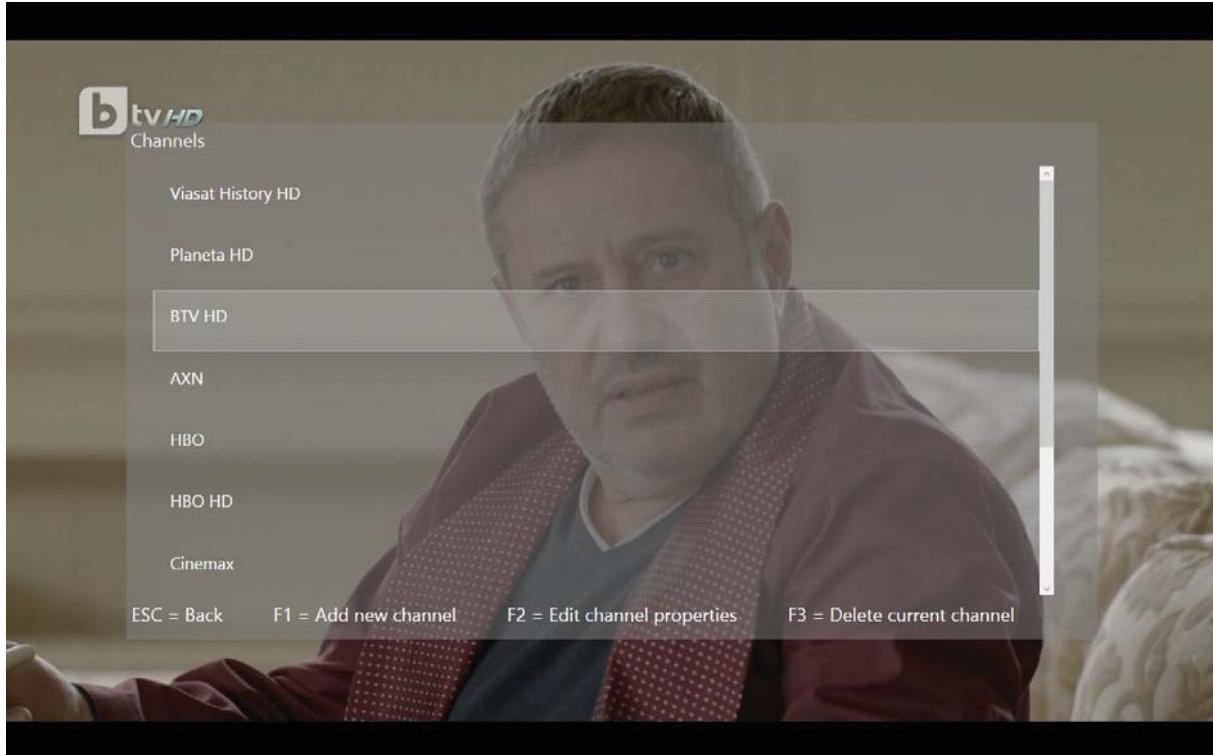


Fig. 1. IP-TV channels manager

2.1.1 Loading application

When the application loads, the first channel starts playing. If the user wants to add new channels, it can safely select Menu -> Channels -> Add New Channel (Fig. 1).

2.1.2 Reading media files and streams

For reading media files it is recommended to install latest codec packages. For reading internet streams also the fast network speed is required – 1.5 Mbps for ordinary, 5 Mbps for HD and 7 Mbps for Full HD sources [2]. PCTVPlay supports only HTTP/RTSP/UDP/MMS protocols.

Channel selection is done by selecting its number or by pressing up and down buttons. On source loading, small information box appears on the top right side of the display. For internet streams it contains the name of channel, and for media files - summary information as filename, size and date created.

When internet source is opened, the request to the server is forwarded. If the connection is lost, a new request is sent. The cycle continues until the channel is successfully loaded or a new request is started. When local file is opened, play/pause and seek buttons are enabled.

2.1.3 Application control

PCTVPlay supports all kinds of remote devices. In present example a device management PC Remote Control is illustrated (Fig. 2). Anyone who wishes to "reprogram" the device can use

keyboard manager. As remote devices work on the principle of a standard keyboard, the software is not able to know which device it manages. So if more than one device are used, it is necessary to select a profile which may be a drawback for the user. We should consider that some devices have their own KeyLearner.

2.1.4 Own explorer

For the convenience of the user the application possesses Explorer that helps to easily and quickly reach the required content. The Explorer allows scrolling through folders of different drives (Fig. 3). When the folder is presented like a channel Prev/Next track buttons are enabled instead of Prev/Next channel.



Fig. 2. PC Remote Control Example

If folder contains only media files, they can be played back sorted (usually in alphabetical order) and unsorted (Shuffle mode). If folder contains other folders play commands are the same, but if the user wants to see their names he or she must hold OK button. It is important to understand that the Folder mode is from the root to down. So if the channel explores through C:\Video, selection menu shows the entire contents of the C:\Video folder and its subfolders. But if the C:\Video\Music is present like a channel, then C:\Video can not be reached because of the restriction imposed.

2.1.5 System requirements

For a good and complete operation of the application need in addition of a good internet connection is the large display. The minimum resolution for the proper visualization of all the menus and channels is 1366x768 pixels. It possible to narrow the window, but during the resizing all the displayed menus hide and the entered information is deleted.

Although these minimum requirements, the user may need to install additional drivers for correct system work!

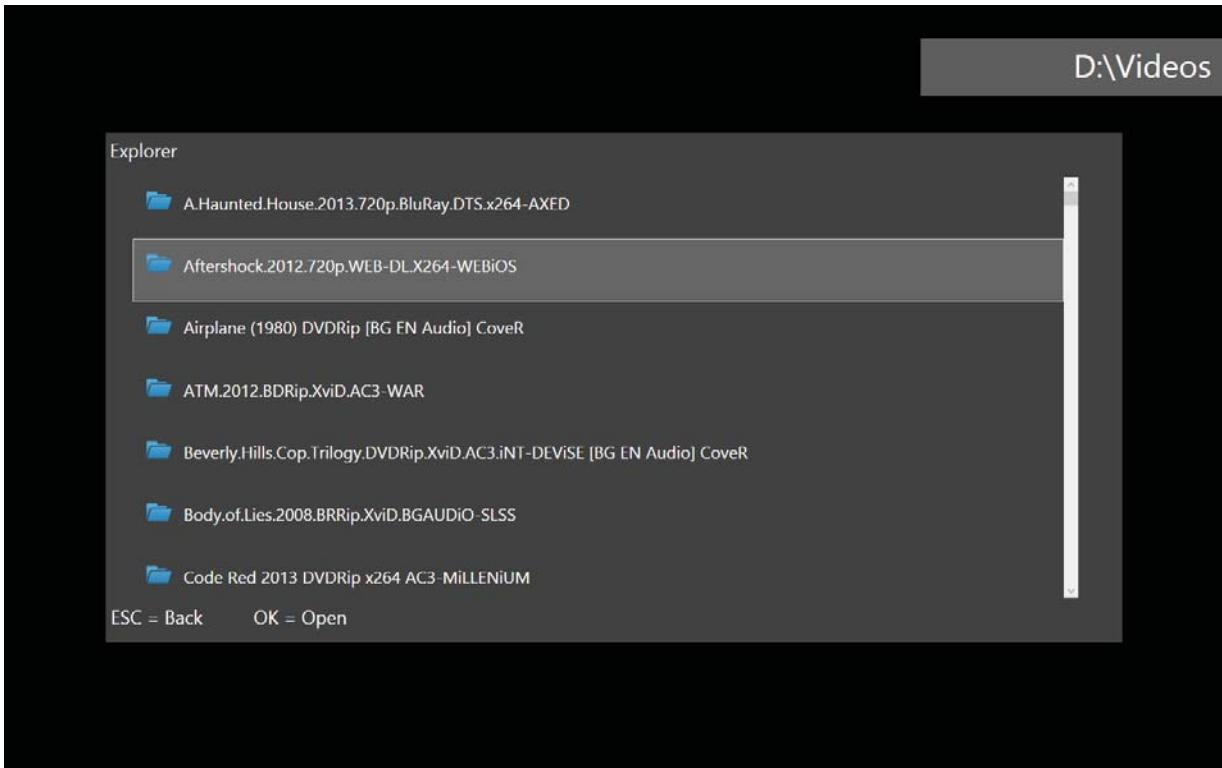


Fig. 3. PCTVPlay Explorer

3 Conclusion

The designed application proposes convenient and easy to use features. But as expected the application can not be perfect and therefore the author plans expanding the capabilities in the following areas:

- Synchronizing with external display (like ATmega32 DOGM 081) that shows current channel;
- Preliminary mapping of known remote devices by default;
- Supporting picture and presentation viewing;
- Integration with Windows 8 Metro UI;
- Sound equalizer;
- Picture correction tools;
- Subtitle fast and easy synchronization;
- M3U playlists support;
- Remote control with self-developed Android application.
- Support of game controllers;
- Recording functions;
- Ability of adding YouTube channels;
- Splitting screen and watching up to 4 sources at the same time (useful for CCTV).

References

- [1] Wikipedia. *Windows Media Center*. 2009, http://en.wikipedia.org/wiki/Windows_Media_Center.
- [2] NetFlix. *Internet Connection Speed Recommendations*. 2013, <https://help.netflix.com/en/node/306>.

Georgi CHAVDAROV
University of Ruse
8 Studentska str., POB 7017, Ruse,
BULGARIA
E-mail: powerbsd@abv.bg

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Concepts and technical solutions for building scalable cloud applications

**Bogdan CIMPOESU, Radu CIGMAIAN
Teacher Coordinator: prof. univ. dr. ing. Mircea VLADUTIU**

Abstract

The web is becoming a predominant platform for developing rapid software solutions. Currently there are multiple solutions available for building network applications. When starting from scratch questions like “what should I use?”, “how can I scale my application?”, “how can I get the performance avoiding high costs?” and others pop up. Finding answers to all this questions is time consuming and hard to get.

1 First section

This research intends to offer a comparison between some of the existing solutions for building scalable cloud applications and show advantages/disadvantages between two possible implementations. It also suggests possible solutions for developing reusable modules for solving problems like security, event logging and flood detection when building cloud applications.

Security wise, every cloud application is prone to risks and attacks, including REST architectures. The most common of these attacks are Distributed Denial of Service (DDoS) and Brute-force attacks. Since these types of attacks make use of weaknesses from both the system and user side it is important to provide a protection layer to filter and prevent these types of attempts.

Two approaches will be implemented for being able to compile a comparison report which will include analysis for:

- Performance
- Learning curve
- Development time

Performance analysis will focus on speed and throughput of the PhP implementation versus the new nodeJs approach. The aim is to achieve better results with using nodeJS.

With learning curve analysis the time required to learn, get familiar and become productive using the node architecture is evaluated. Both short and long term advantages and disadvantages are taken into account. Development time, ease of coding and maintainability represent the key factors in this evaluation.

2 State of the art

2.1 Background

A cloud application is a program that functions in the cloud. The reasons for choosing this kind of architecture for an application are both technical and business related. The main characteristic of a cloud application is that it requires no infrastructure, all the concerns regarding real estate, physical security and hardware management being taken over by the cloud. The cloud provides a just-in-time infrastructure making the application scalable as it grows. Moreover, being a "scriptable infrastructure" automation eases development lifecycle and improves system's testability. Another characteristic is the cloud's ability to handle disaster recovery, business continuity and traffic excess by routing the excess to the cloud [1].

One of the risks that such an architecture brings is the vulnerability to abusive behaviours: flood, denial of service and brute force attacks. Currently there are a number of methods that can be used to prevent such abuse. A solution would be to monitor actions of the user. Form field validation and data validation can be set in place. It is possible to track the time between submits/requests from a user and implement a behaviour for certain patterns [2].

Other solutions involved encrypting all forms using javascript with tools that allow you to enter the HTML code and receive a working, if not optimal, piece of javascript code that has the exact same functionality. A safer but unappealing method is using a captcha. This while ensuring the authenticity of the user can easily alienate them.

All these solutions offer a certain level of security bound with a certain compromise.

2.1 Related Work

NodeJS is considered "the latest" technology for building fast, scalable network applications and there is intensive work done on stability and feature coverage. The goal is to contribute to this cause by improving existing modules, developing new ones and bringing new concepts to the table.

Given their nature and scope, cloud applications are highly dependent on security and uptime. Protection layers are being used in a multitude of scenarios with the goal of giving the user safe access to the desired content.

One of the most common forms of aggression is a denial of service attack. This aims to push the limits of the host system to a failure or blocking state. To prevent and dissipate the damaging effect of such attacks an approach called "Web application level approach against HTTP flood attacks" has proven itself very useful [3].

An outcome of this thesis is a functional protection module for a node based rest API implemented in a running web based application. While the ideas behind the protection mechanisms are known, implementing such a module for a node based application would allow any software using this API ease of access and plug-and-play utility of it.

3 Implementation

The goal is to offer solutions for real time cloud applications. For fulfilling these requirements a client-server architecture is required.

The most used protocol for communicating between server and client is HTTP. One of the advantages is the possibility to implement a browser based client side application. We've also taken in consideration that most application development environments offer APIs to interface with HTTP protocols, for example building an android application will be possible.

3.1 Architecture

As mentioned there will be two approaches implemented (Figure 3.1). These will broadly follow same concepts using different implementation methods. The first approach will have a scalable nginx web server for load balancing and for serving files and nodeJs as data provider while

second method will use PHP with MySQLi connector and a web application with a internal developed pure Javascript framework.

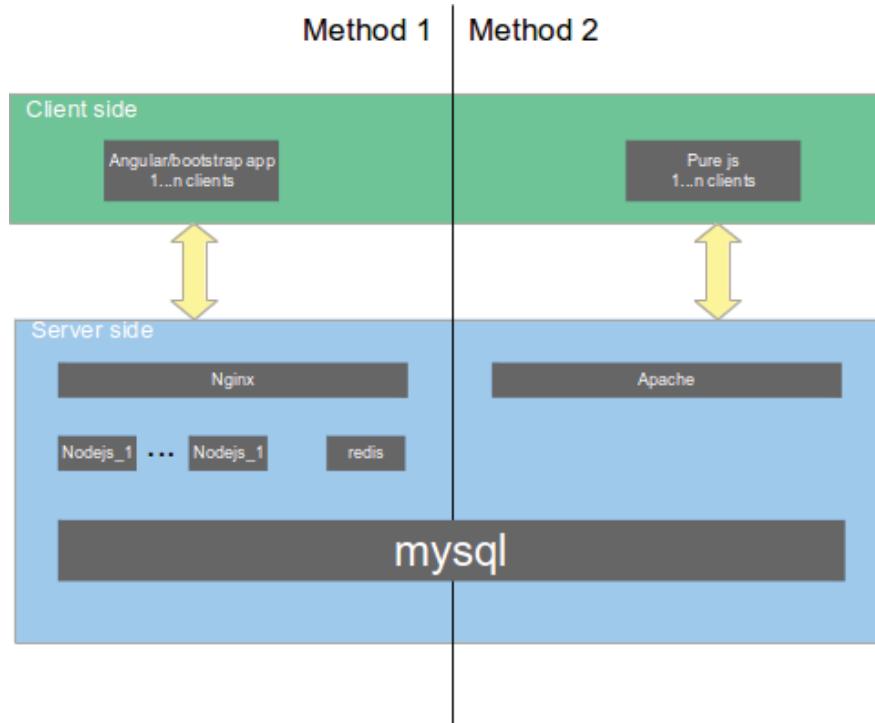


Fig 3.1: Proposed solutions

Both methods are publishing a REST API accessible to World Wide Web through HTML protocol [4].

The database will implement a multi tenant architecture facilitating user rights management and synchronous multiple access. The database can also store most of the business logic implemented by triggers, stored procedures, etc [5].

For being able to compare the two solutions we've implemented a simple html REST compliant API using:

- nginx with node, more exactly express js for route handling
- apache, having a php implementation

The API is designed to respond on simple routes like localhost/auth/user, providing the user information. Both implementations are triggering a mysql request and returning the response. In order to provide a solid application, features like anti-flood, requests monitor, user data security module and data sanitization module are a must. We propose a layered architecture for implementing the upper mentioned features. Each module should have the capability to cancel the request or forward to the next module. NodeJS middleware functionality fits perfectly to these requirements; in consequence our chosen solution is node-based (Figure 3.2). Scalability is another high importance factor when developing cloud application. Nginx server is used for load balancing and file serving. Nginx is a lightweight server designed for such features.

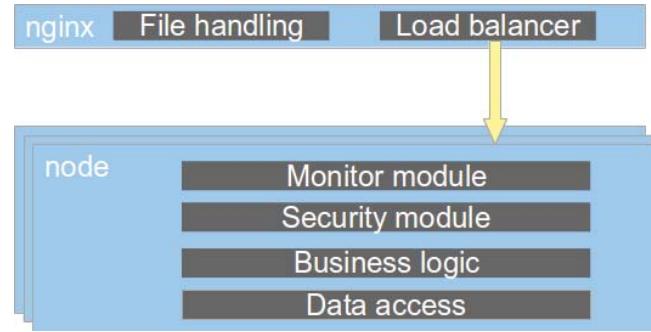


Fig. 3.2: Nginx, node based architecture

3.2 Monitor module

Denial of service attacks are usually attempted through flood attacks that aim to choke the system with more requests than it can handle under normal circumstances with the scope of crippling the system. A distributed denial of service (DDoS) attack is one where the requests come from multiple compromised systems towards a single target [6].

As nodeJS is still in a development stage there are multiple vulnerabilities that keep emerging, amongst which DoS vulnerabilities. They are actively patched but new security issues are often found and the latest version of nodeJS is not always used. With this in mind a pre-emptive DoS/flood protection layer would guarantee the stability and security of a HTTP server.

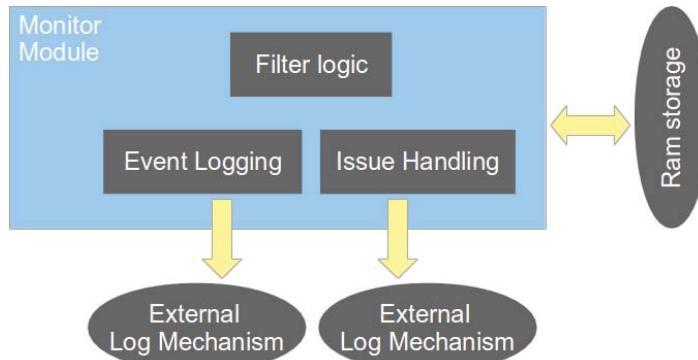


Fig 3.1: Node monitor module architecture

The protection layer consists of a monitor module that receives the HTTP requests. It's role is to filter out unwanted requests through a configurable set of rules to mitigate and prevent flood and DoS/DDoS scenarios (Figure 3.1).

NodeJs was chosen as the implementation language for this module as it allows fluid integration with the rest of the application. It allows easy integration with NoSQL databases such as MongoDB through its JSON API producing a lightweight and fast layer. Each individual request is parsed and analyzed. A request becomes an JSON containing the IP address and the timestamp of the last request together with a blocked status.

```
{
  ip : {
    blocked : false,
    route : {
      requests : [sec , hour , day],
      date : last_request
    }
  }
}
```

All database queries are handled in-memory using MongoDB to increase speed together with ease of use as it is also a JavaScript based NoSQL database. Different routes can have differently configured filters, the mapping of which can be handled separately. For example the route “/api/login1route” can be allowed to receive up to 400 requests/second, but a more database intensive route such as “/spreadsheets/acct1lookup” can be limited to 100 requests/second. Besides being a DoS prevention mechanism this also functions as a brute-force defence, as you can filter and restrict login attempts.

Any found issue is reported back to the calling layer which logs the event and decides on the action taken. Thus a user can be warned, delayed for a fixed amount of time or blocked completely. The event logging mechanism also sends an in-app notification when a violation of the rules occurs. This can be in the shape of an email to the web-app admin to validate whether the blocked IP actually executed an attack attempt or if it was just a false positive (i.e. multiple users with the same IP).

Every valid request is then passed on to the following layer.

3.3 Security module

The security module intends to be generic and provide an API that allows each application to implement its own role and validation scheme. The session and cookie management implementations are also application specific (Figure 3.3).

The module exports a middleware function with following parameters:

- role - the role needed in order to be able to access the required route
- data - [['key1'...], ['field1'...], ['notMandatoryField1'...]]

keys - data that can be found under req.params(in the route), also used for role validation

ex: ['idShop'] used for checking if user is administrator on shop with id field - data that can be found on req.body and needs to be validated

notMandatoryField - same as filed excepting that validator is omitting the unset data

Usage example:

```
app.post("api/shop/:idShop", s.validate('admin', ['idShop'], null, ['address'])) , shop.update);
```

The above route will be executed when the user makes a post request on www.app.ro/api/shop/1.

The user needs to be an administrator of the shop defined by id 1. There are no mandatory parameters for this route. If the address is available it will be verified by dataValidator. The “administrator of shop” validation logic is specific and needs to be implemented by each application.

The shop.update(business logic function for this route) is called only if the validation and authentication is successful. An error code is returned by server for illegal requests.

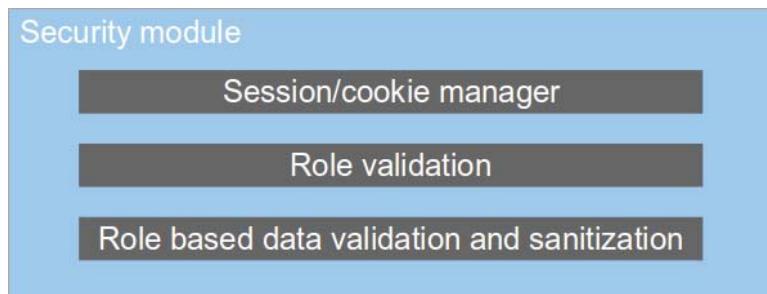


Fig 3.3: Security module

The data validator will check each parameter, sanitize the data and move it to a secure object where it will be accessed by the following modules.

Using this module will assure that information with incorrect format will not be used later; also it will prevent SQL injection and other methods of data injection.

4 Solution analysis and experimental results

One of the big advantages of using a nodeJS based solution is having the same language on both client and server side. This improves the learning curve as there is only one language that needs to be learnt and used. Switching from javascript(frontend development) to other languages(Ex: PhP for backend) results in time loss. There are multiple articles and studies that prove multitasking worsens development speed. Using javascript as a unique language will break the wall between frontend and backend developers and will reduce the delays added by situations where frontend is waiting for backend. Furthermore, for architectures where a relational database is not a must, NoSQL databases can boost the development process offering javascript APIs and JSON data structures [7].

An important factor is the performance. In order to test our concepts we've used one of our applications where we had both a PhP and nodeJS api implementations. The tests were run on the same machine using the localhost interface to avoid network delays.

The PhP application was ran by apache 2.4.6 and PhP 5.5.3 using mysqli as a mysql connector. For the nodeJS api, node v0.10.23 with the cluster interface was used. Node-mysql facilitated the access to the mysql server [8].

The test implied:

- running a GET request on a specific route
- processing the request on the server(the request implied an mysql query)
- returning the result in JSON format

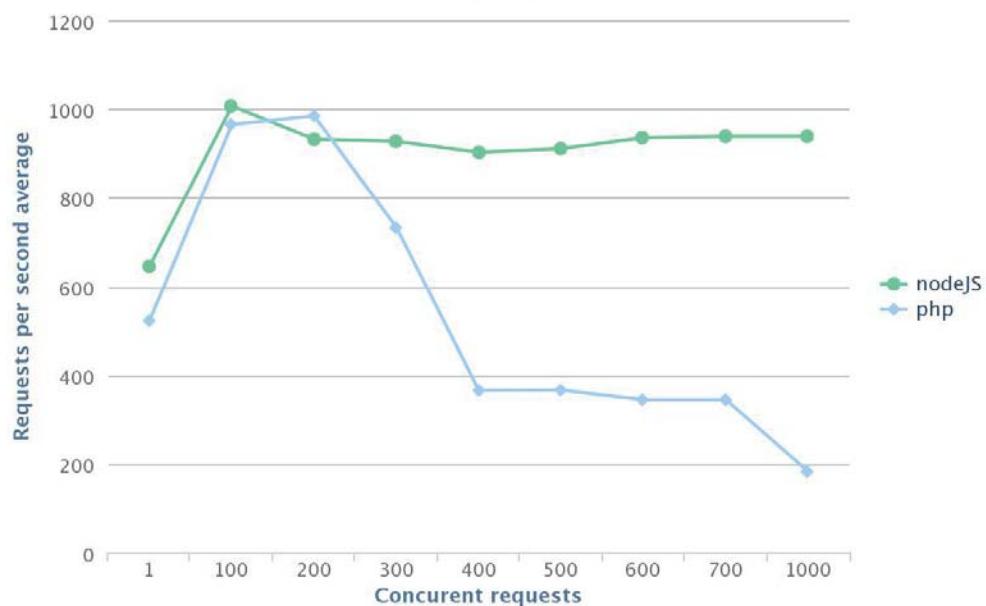


Fig 4.1 : PhP and nodeJS performance graph

The results show that nodeJS and PhP go head to head until 300 simultaneous requests. From there nodeJS's single thread event based architecture starts to overcome PhP multiple thread implementation. Results show that nodeJS is capable to server 1000 requests/second in 1000

concurrent requests situations, making it more suitable for applications where non intensive server computation is required.

Even though nodeJS fits for a broad spectrum of use cases and maps really well to today's continuously changing features and performance requests, there are situations where choosing compiled languages is a wise decision. A good example for this is applications where intensive server side computation is needed. Node was designed for simple, quick tasks and will suffer performance penalties for complex tasks. Compiled languages will by far overcome nodeJS for this kind of applications [9].

4 Conclusions and future work

The state of the art relies in the design concepts implemented in order to meet high performance and scalability requirements. The proposed architecture represents a solution for switching from desktop application to cloud base application without losing the real time aspects. We are facing impressive growth in cloud based application and we expect this growth to continue.

As with any growing trend, this brings with itself risks and vulnerabilities. Given this, security and threat prevention represent an important part in the development of such an application.

Future work consists of two main directions. One implies continuing developing and improving the security and monitor modules. The intention is to publish these modules and contribute to node community. Some refactoring is still needed in order to boost performance. Currently the modules have parts that contain "hardcoded" implementations for specific applications, which results in the need of making the modules more generic. When creating the design, concepts for expandability were taken in consideration leaving place for growth and customization.

The second direction implies accumulating more experience with these technologies in order to help provide a better overview in upcoming work. Currently we are developing multiple applications and prepare for production phase. This step represents the proof of concept for our work.

References

- [1] Varia, Jinesh. “Architecting for the cloud: Best practices.” Amazon Web Services 2010
- [2] Thomas Landauer “Preventing Spam in Form Submissions without Using a CAPTCHA” 2011
- [3] Gökhan Muharremoğlu, “Web Application Level Approach against the HTTP Flood Attacks IOSEC HTTP Anti Flood/DoS Security Gateway Module”, Hakin9 IT Security Magazine November 2012
- [4] Gregor Roth, “RESTful HTTP in practice”, August 2009 [Online: <http://www.infoq.com/articles/designing-restful-http-apps-roth>]
- [5] Frederick Chong, Gianpaolo Carraro, Roger Wolter “Multi-Tenant Data Architecture”, June 2006 [Online: <http://msdn.microsoft.com/en-us/library/aa479086.aspx>]
- [6] Hrishikesh Khatavkar “Stop DDoS Attacks”, November 2012 [Online: <http://www.e-zest.net/blog/stop-ddos-attacks/>]
- [7] Tomislav Capan, “Why The Hell Would I Use Node.js? ”, August 2013 [Online: <http://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>]
- [8] Rauch, Guillermo. “Smashing Node.js: JavaScript Everywhere”. John Wiley & Sons, 2012.
- [9] Gackenheimer, Cory. “Node.js Recipes: A Problem-Solution Approach.” Apress, 2013.

Radu CIGMAIAN
Universitatea “Politehnica” Timisoara
Bulevardul Vasile Parvan 2, Timișoara 300223
Romania
radu.cigmaian@gmail.com

Bogdan CIMPOESU
Universitatea “Politehnica” Timisoara
Bulevardul Vasile Parvan 2, Timișoara 300223
Romania
cimpoesu.bogdan@gmail.com

The Future of Augument Reality

**Octavian-George Cimpău
Teacher / Coordinator: Anca Vasilescu**

Abstract

Purpose – This article aims to introduce the reader in the astonishing world of augument reality. It presents general information on augument reality, focusing on the possibilities of development in that direction by using the technology available at this point on the market. The goal is to present a new concept about ‘reality’, the result consisting of an element that the eye can see but the hand cannot touch.

Research limitations/implications – The reliability of the research presented below is however limited as the augument reality itself is an evolving technology in the phase of market introduction. The outstanding available technology is only available within the top of IT companies, the concept being however achievable, but at a smaller scale, with medium results, using limited resources (such as a laptop and a smartphone). Documentation is rare and vague, so is the community, and the good hardware is hard to get.

The definition and characteristics of augument reality are still uncertain. Still, if you think out of the box, you will see beyond the concept of augument reality and get an image on its potential. If we take a look at the big companies on the market, we will notice a huge growth in this area. Combining the needs of our society with new innovative ideas, the possibilities are unimaginable.

1 Introduction on Augument Reality

Video games have been with us for nearly 30 years, entertaining us and filling our hearts with joy ever since the early 70's when the very first arcade game was introduced. Computer graphics grow year by year, becoming much more sophisticated than they used to be, by this the games are pushing the boundaries of photorealism. Nowadays, researchers and engineers or passionate technology users are able to bring to life that graphics by pulling them out of our screens (television, computer or mobile displays) and integrating them into real-world environments. This strange and interesting new technology at the same time, called augmented reality, blurs the line between what's real and what's computer-generated by enhancing what we see, hear, feel and smell. Henri Bergson said that “The eye sees only what the mind is prepared to comprehend”, augument reality is trying to break that boundary.

On the spectrum between virtual reality, which creates immersive computer-generated environments, and the real world, augmented reality is closer to the real world because it doesn't distortion the natural environment around it, but leaves it as it is and adds just some extra elements such as graphics, sounds, haptic feedback and smell to it. Everyone, starting with children, who can animate their story

books, to tourists who can animate the places they visit, for example by looking for the closest subway stop, can now benefit from the ability to place computer-generated graphics in their field of vision. In other words augmented reality is not changing the world around us, but just the way we view it - or at least the way we users see the world. Picture yourself walking or driving down the street, with an augmented-reality device such as a smart pair of glasses, informative graphics will appear in your field of view, and audio will coincide with whatever you see. These enhancements will be refreshed continually to reflect the movements of your head. Similar devices and applications already exist today, particularly on smartphones.

In this paper, we'll take a look at where augmented reality is now and where it may head soon, by considering all aspects related to the available technology, the market trends and the costs.

1.1 Definition

According to a research conducted by Graham, Zook and Bolton, **Augmented reality** (AR) is a live direct or indirect view of a physical, real-world environment whose elements are augmented (or supplemented) by computer-generated sensory input such as sound, video, graphics or GPS data. It is related to a more general concept called mediated reality, in which a view of reality is modified (possibly even diminished rather than augmented) by a computer. As a result, the technology functions by enhancing one's current perception of reality.[6] By contrast, virtual reality replaces the real world with a simulated one.[3] Augmentation is conventionally in real-time and in semantic context with environmental elements, such as sports scores on TV during a match. With the help of advanced AR technology (e.g. adding computer vision and object recognition) the information about the surrounding real world of the user becomes interactive and digitally changeable. Artificial information about the environment and its objects can be overlaid on the real world.[2][4]

1.2 Technology

1.2.1 Hardware

Augmented reality requires a series of hardware components, the most relevant being the processor and display which are compulsory and which model and display the data provided by the sensors (i.e. camera, MEMS sensors, GPS and solid state compass), all of the hardware elements described being available in smartphones and tablets.

1.2.3 Technology used

Augmented reality can be achieved by means of optical projection/display systems, monitors and hand held devices. A key measure of augmented reality systems is how realistically they integrate augmentations in the real world. The software must derive real world coordinates, independent from the camera, from camera images. This process is called image registration and it uses different methods of computer vision, mostly related to video tracking.

Augmented Reality Markup Language (ARML) is a data standard developed within the Open Geospatial Consortium (OGC)[5], which consists of an XML grammar to describe the location and appearance of virtual objects in the scene, as well as ECMAScript bindings to allow dynamic access to properties of virtual objects.

Some of the most well-known software development kits in AR SDKs are offered by Vuforia, Metaio, Wikitude and Layar.

2 Research

2.1 Market research

According to a report made by Business Wire – Research and Markets, the analysts forecast the Global Augmented Reality market to grow at a CAGR of 132.2 percent over the period 2013-2018. The number of applications is increasing, augmented reality being used in many sectors such as Defence, Education, Consumer Electronics, Industrial, and Healthcare. A market increase in this direction might however take a while due to the high cost of devices used.

Commenting on the report, an analyst from the team said: “The Global Telecommunication market is continuously upgrading the technology and protocols used for communication. While developed countries are switching from 3G to 4G networks, developing countries are switching from 2G to 3G networks. Such instances indicate the upgrading of the global telecommunication network, enabling vendors in the online software and application space to connect strongly with their target customers. Moreover, the introduction of customer-friendly mobile internet packages is further driving the demand for mobile internet applications such as augmented reality applications. Therefore, the rapid development of network infrastructure is a major trend that is expected to propel the growth of the Global Augmented Reality market.”

2.2 Vuforia Augmented Reality SDK

2.2.1 What is vuforia?

Vuforia is a software platform with support for mobile devices that operate on iOS or Android.

2.2.1 How does vuforia works?

This software development kit (SDK) uses Computer Vision technology to recognize and track planar images and simple 3D objects, such as boxes, in real-time. This image recognition capability allows the developers to position and orient virtual objects (i.e. 3D models and other media), in relation to real world images when these are viewed through the camera of a mobile device. The virtual object then tracks the position and orientation of the image in real-time so that the viewer's perspective on the object corresponds with their perspective on the Image Target, so that it appears that the virtual object is a part of the real world scene.

2.2.1 Why should we use vuforia?

A first benefit of using vuforia is the wide range of Application Programming Interfaces (API) supported such as C++, Java, Objective-C, and the .Net languages through an extension to the Unity game engine. In this way, the SDK supports both native development for iOS and Android while also enabling the development of AR applications in Unity that are easily portable to both platforms. Another great advantage is that Vuforia SDK supports a variety of 2D and 3D target types including ‘markerless’ Image Targets, 3D Multi-Target configurations, and a form of addressable Fiduciary Marker known as a Frame Marker. Additional features of the SDK include localized Occlusion Detection using ‘Virtual Buttons’, runtime image target selection, and the ability to create and reconfigure target sets programmatically at runtime.[8]

2.2.1 When should we use vuforia?

Whenever you want to integrate AR in your mobile application with a fast and easy to use SDK.

2.3 Unity (game engine)

2.3.1 What is unity?

Unity is a cross-platform development engine for the creation of 2D and 3D games and interactive content, with a built-in IDE developed by Unity Technologies.

2.3.2 Why should we use unity?

Because you can do it all with Unity: create, deploy, market and monetize cross-platform games and interactive content. It also supports a wide range of programming languages such as: Javascript, C# or Boo (which is a Python-inspired syntax).

“JavaScript (JS) is an interpreted computer programming language. It was originally implemented as part of web browsers so that client-side scripts could interact with the user, control the browser, communicate asynchronously, and alter the document content that was displayed.”

“C# (pronounced as see sharp) is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.”

“Boo is an object-oriented, statically typed, general-purpose programming language that seeks to make use of the Common Language Infrastructure's support for Unicode, internationalization, and web applications, while using a Python-inspired syntax.”

2.3.3 When should we use unity?

Whenever you want to develop a fast cross-platform application.

3 Program Code

3.1 General info

With the help of Vuforia Framework we will be able to recognize a certain image pattern and animate a 2D or 3D object on top of it. Here is how it works:

3.2 Unity 3d application

A unity application consists of 3 big components, the forth being the augment reality SDK:

1. Materials
2. Scripts
3. Plugins
4. Augment Reality SDK

1. In the Materials folder we can find all our 2D or 3D objects ready for use. From here we can import and display them to into our application.
2. In the scripts folder we have the logic of our application, from general scripts that animates our objects to the business logic.
3. The Plugins folder contains dependencies of special platforms specific on which we want to deploy the app.

All of those can be found under the Assets folder.

3.2.1 TriggerAnimation.js (simple javascript controller)

```
#pragma strict

function Start () {

}

// when the object is rendered start its animation
function Update () {
    if (!animation.isPlaying("Space_Man@gangnam_style")){
        animation.Play("Space_Man @gangnam_style");
    }
}
```

3.2.2 AndroidUnityPlayer.cs (more complex cs controller)

```
using UnityEngine;
using System.Collections;

public class MinionController : MonoBehaviour {

    private Animator anim;
    Quaternion originalRotation;
    public string ID;
    bool dance = false;
    Transform lookTransform = null;
    float lookWeight = 0.0f;
    float lookSmoothen = 3.0f;
    Transform Trans = null;

    public Transform cachedTransform
    {
        get
        {
            if (Trans == null) { Trans = transform; }
            return Trans;
        }
    }

    // Called before the script starts
    void Awake ()
    {
        anim = GetComponent<Animator>();
        originalRotation = cachedTransform.rotation;
    }

    // Update is called once per frame
    void Update ()
    {
        // Viewing
        anim.SetLookAtWeight(lookWeight);
        if (IsLook())
        {
            anim.SetLookAtPosition(lookTransform.position);
            lookWeight = Mathf.Lerp(lookWeight, 1f, Time.deltaTime * lookSmoothen);
        }
        else
        {
            lookWeight = Mathf.Lerp(lookWeight, 0f, Time.deltaTime * lookSmoothen);
        }
    }

    // Viewing settings
```

```

public void SetLook(Transform lookTransform)
{
    this.lookTransform = lookTransform;
}

public bool IsLook()
{
    return (lookTransform != null);
}

// Set dancing
public void SetDance(bool dance)
{
    this.dance = dance;
    anim.SetBool("Dance", dance);
}

public bool IsDance()
{
    return dance;
}

// Transform initialization
public void ResetTransform()
{
    cachedTransform.localPosition = Vector3.zero;
    cachedTransform.rotation = originalRotation;
}

}

```

3.2.3 AndroidUnityPlayer.cs (Qualcomm Augmented Reality Scripts)

```

using System;
using UnityEngine;

/// <summary>
/// This class encapsulates functionality to detect various surface events
/// (size, orientation changed) and delegate this to native.
/// </summary>
class AndroidUnityPlayer : IAndroidUnityPlayer
{
    .....
    /// <summary>
    /// check for screen orientation changes
    /// </summary>
    public void Update()
    {
        if (Application.platform == RuntimePlatform.Android)
        {
            if (QCARWrapper.Instance.HasSurfaceBeenRecreated())
            {
                InitializeSurface();
            }
            else
            {
                // if Unity reports that the orientation has changed, reset the member variable
                // - this will trigger a check in Java for a few frames...
                if (Screen.orientation != mScreenOrientation)
                    ResetUnityScreenOrientation();

                CheckOrientation();
            }
        }
    }

    if (mScreenWidth != Screen.width || mScreenHeight != Screen.height)

```

```

        {
            mScreenWidth = Screen.width;
            mScreenHeight = Screen.height;
            QCARWrapper.Instance.OnSurfaceChanged(mScreenWidth, mScreenHeight);
        }
    }

    mFramesSinceLastOrientationReset++;
}

// Java resources need to be explicitly disposed.
public void Dispose()
{
    if (Application.platform == RuntimePlatform.Android)
    {
        mCurrentActivity.Dispose();
        mCurrentActivity = null;

        mJavaOrientationUtility.Dispose();
        mJavaOrientationUtility = null;
    }
}
.....
}

```

4. Conclusions

The Augument reality is by itself a challenge: it is a brand new concept, a new technology which promises to render a series of benefits to users, by facilitating access to information, personalising it according to the user's needs, making it interactive, logical, alive and maybe even fun. It can radically impact the way we perceive education, tourism, social defence and other major sectors that we depend on. The first few steps in this direction have already started and although the affordable existing technology is limited, the benefits that a development of Augument reality would bring surpass all present challenges like technology and costs. All we need is just one small step in this direction-hopefully, this presentation has provided just the impulse that was needed.

References

- [1]Augument World Expo. “*Top 10 AR SDKs*”
- [2]Brian X. Chen “*If You’re Not Seeing Data, You’re Not Seeing*”
- [3]Jonathan Steuer “*Defining Virtual Reality: Dimensions Determining Telepresence*”
- [4]Kerry Maxwell “*Augument Reality*”
- [5]M. Bajura and Ulrich Neumann „*Dynamic Registration Correction in Augmented-Reality Systems*”
- [6]M. Graham, M. Zook and A. Boulton “*Augument reality in urban places: contested content and the duplicity of code.*” – July 2013
- [7]Qualcomm Developer Network “*Augument Reality (Vuforia)*”
- [8]Vuforia Developer “*Vuforia SDK Native – Android & iOS API Reference*”

CIMPU Octavian-George
 Universitatea Transilvania
 Facultatea de Matematica si Informatica, Informatica Aplicata
 Str. Iuliu Maniu nr. 50, 500091 Brasov
 Romania
 E-mail: octavian.cimpu@gmail.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Improving word learning experience using spaced repetition technique implemented in Windows Phone 8 application

**Mykhailo Dorokhov, Mariia Honcharova
Teachers Coordinators: Iryna Ushakova, Snezhana Lembik**

Abstract

In this paper significant lack of online and mobile educational services for improving vocabulary skills of French languages are revealed, especially on mobile platforms. The spaced repetition technique are considered as a pivot for improving vocabulary skills. Use cases for future application which will help to solve this problem are defined, all features of French language are considered. Actual mobile application for Windows Phone 8 are developed, database design, user interface and some code samples, which show the details of implementation are presented.

1 Introduction

Learning a foreign language is a big part of educational process of a person. Many of students starts learn foreign language in school, then continue in university, and still it is a process of a whole life, because the higher stage are completed, the harder reaching the next stage becomes. There are many private language courses around the world, and many online services appears for foreign language learners. However, usually, if we speak about online services, all of them are aimed to teach you English, which is evident, as English has the highest popularity among the students. Still, those who want to study other languages cannot find appropriate tools. The reason is that English is so wide spread because of more or less clear grammar, absence of most of verb grammar form, a little difference for adjectives of different genders. According to The Economist journal, “English is a basic skill of modern life comparable with the ability to drive a car or use a personal computer” [1].

Still, French language remains as the world’s second language. It is official language in 29 countries. 338 million people speak French in the world [2]. Speaking French offers various opportunities: studying in French universities, which are open for students from other countries, many scholarships are available for those who want to study in France; many jobs where French is required; international business with French-speaking countries.

So why we do not see a lot of French language learning services or mobile applications? The reason is again in complexity. It is hard to consider all the features of French so that it would be effective. The cost of developing such online service is high, while the audience is smaller than English language services have. Where, by the way, there are much more experience of developing such things.

So what do we have here? A free place in the market of mobile application, which could be filled with the appropriate application, which should consider all the distinctive features of French language, and definitely would have an audience as well.

2 French distinctive features and language learning

As we see on the Fig. 1, many people in EU more or less acquainted with French, which means that it is learnt in schools and universities. The next question is about the process of studying.

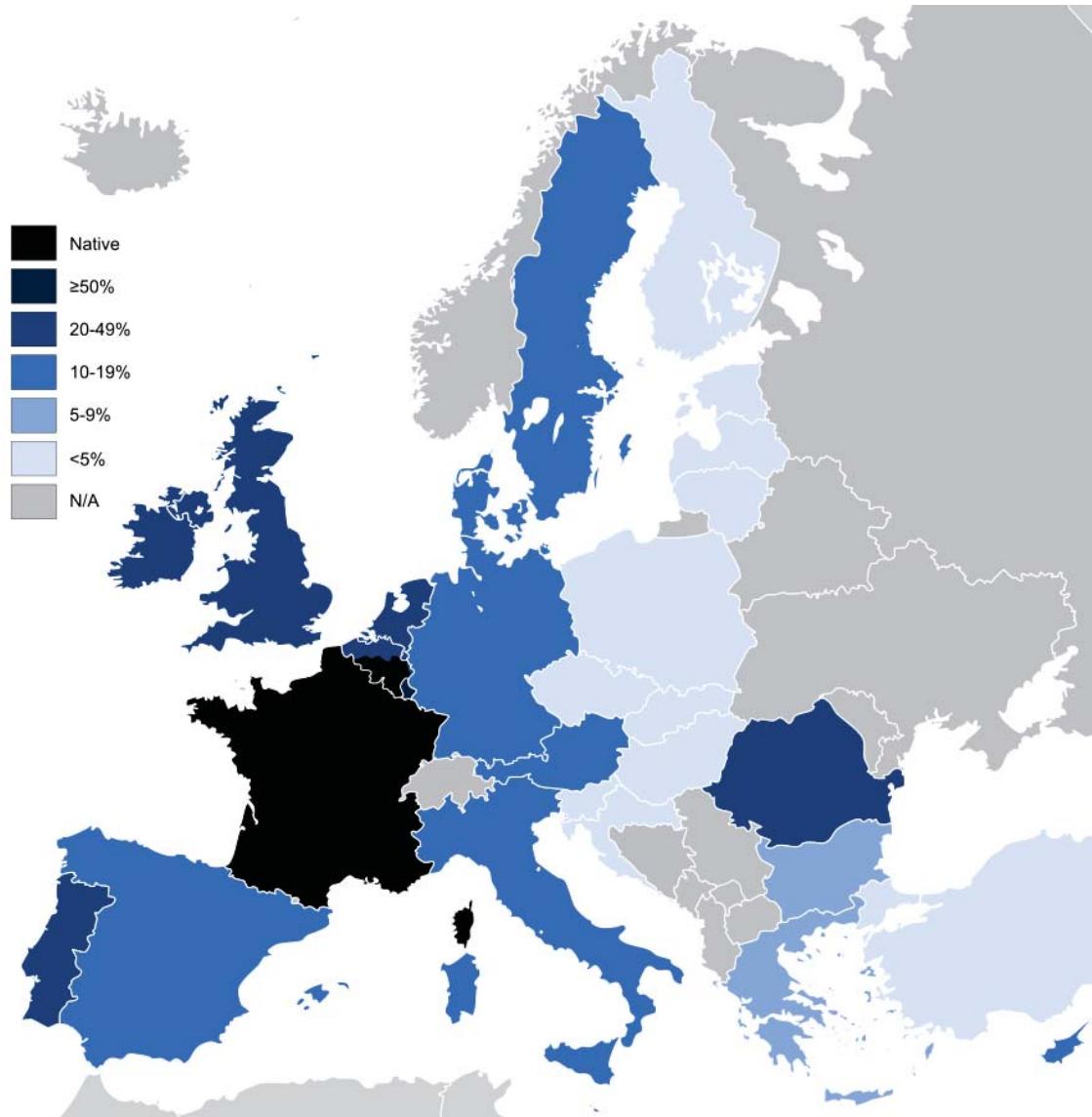


Fig. 1. French language popularity in EU

So how do we learn a foreign language? There are usually two parts – grammar and vocabulary. Many grammar manuals provide well-structured information and useful grammar references. Teachers could explain grammar rules, they could be remembered or learnt by heart. Although, training vocabulary skills is still a time-consuming process, which also depends on how good memory one's have and what memorizing techniques one's uses.

How typical lesson in school or university passes? Especially if it is a third language lesson, which is in some countries typical situation for French. There is often a home task with text which contain new words, that should be learnt by heart by the next lesson. When there is a need to learn a new set of words by heart, it is usually a challenge for a student who starts learning a new language.

That is why those who learn a foreign language often seek for a new and efficient way to do it.

2.1 Tools for improving vocabulary skills

Nowadays, several services exist as a web application or as a mobile application to help learners to memorize words. Let's briefly talk about what do we have now in order to solve this problem. Wordsteps.com (Fig.2) provides web-service and mobile application, in which you can choose a package of words of certain theme. However, if you have a need to learn your own list of words, the application has no support for this scenario.



Fig. 2. Wordsteps.com

Memrise (Fig. 3) is a web-service with the android application with similar functionality has defined word dictionaries but has a lack of flexibility.

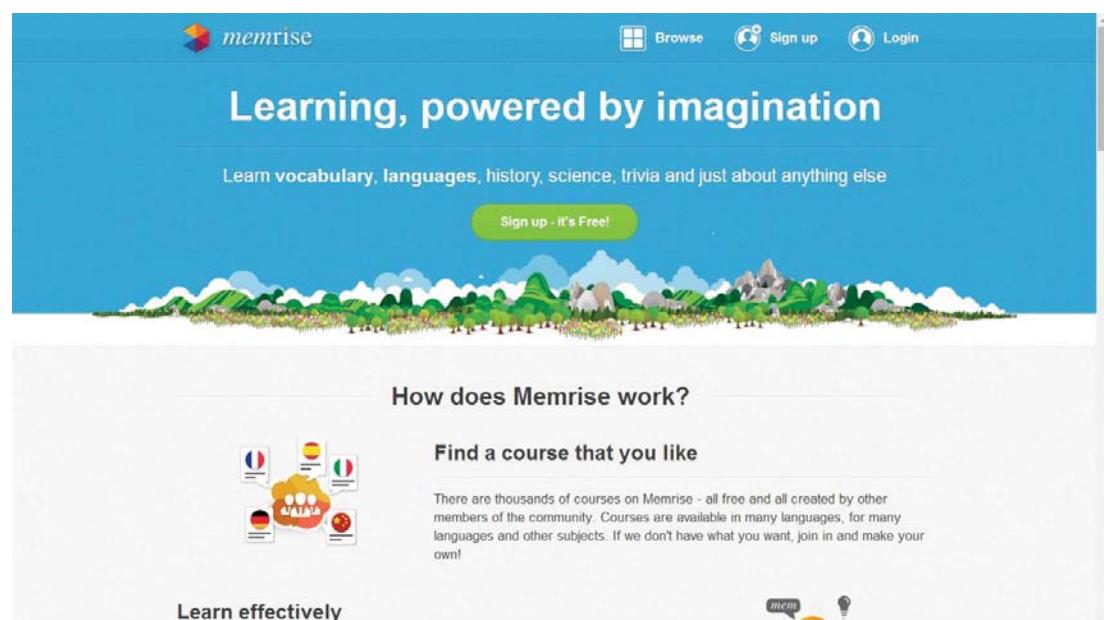


Fig. 3. Memrise.com

Unlike previous applications, Wordile (Fig.4) is Windows 8 modern-style application, which supports possibility to create your own word packages and train them. This is great for those students, who learn a language with teachers giving them texts to learn thus, there is a need to learn by heart with the vocabulary in the text. Nevertheless, the disadvantage is that Wordile has only one type of training that is just typing the word without various types of tests.

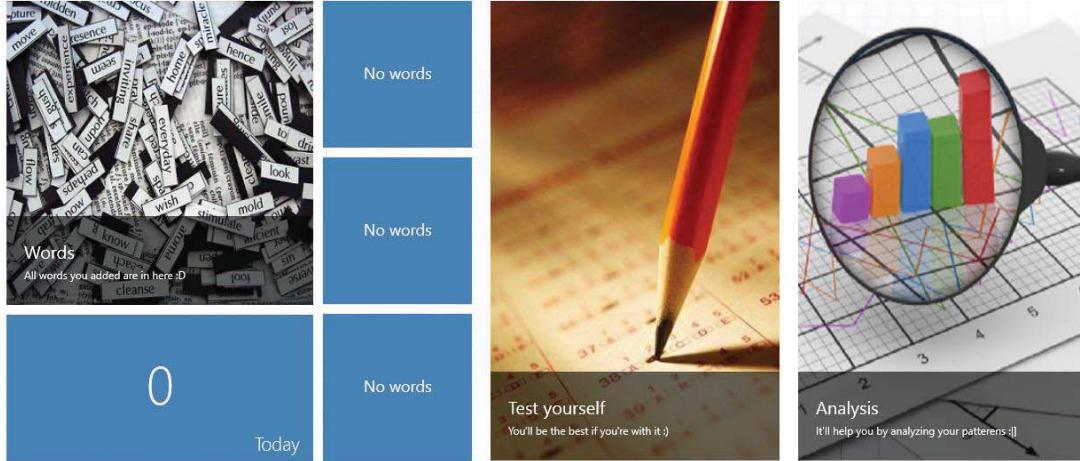


Fig. 4. Wordile

So, as we can see, a lot of application and services exists, but mostly all of them are developed for those who learn English. And even if there are support for French words, this is not enough and do not cover all the French language features.

2.2 Use cases for concept application for improving vocabulary skills

So, considering all that was mentioned above, the application should have the features as follows:

- Creating and managing your own wordbook, where user shall have the possibility to add words or delete them if necessary, and always have beautiful, well-organized list of the words right in the application;

- Various tests based on the words from the wordbook;

In UML it looks like this (Fig. 5):

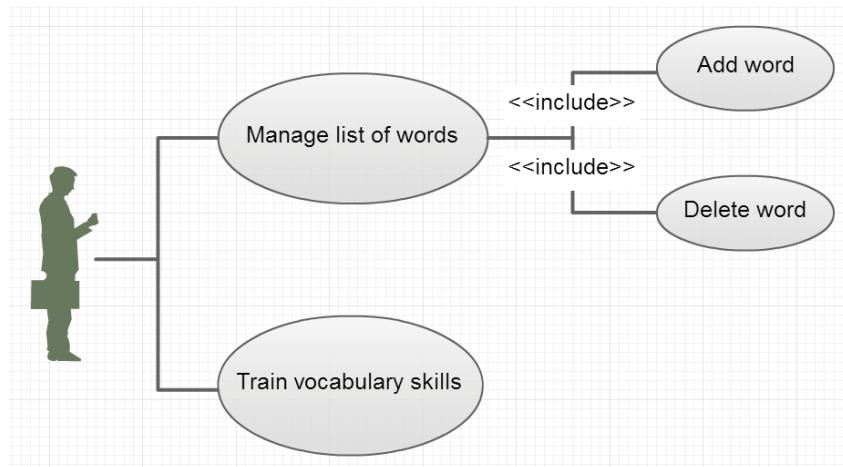


Fig. 5 Use Cases diagram

2.3 Spaced repetition technique

In 1885, Hermann Ebbinghaus extrapolated the hypothesis of the exponential nature of forgetting [3]. The following formula (1) can roughly describe it:

$$R = e^{-\frac{t}{S}} \quad (1)$$

Where R is memory retention, S the relative strength of memory, and t is time.

Ebbinghaus studied the memorisation of nonsense syllables, such as "WID" and "ZOF" by repeatedly testing himself after various time periods and recording the results. He plotted these results on a graph creating what is now known as the "forgetting curve". From his discovery regarding the "forgetting curve", Ebbinghaus came up with the effects of "overlearning". Essentially, if something is practiced more than what is usually necessary to memorize it, overlearning would have been effectively achieved. Overlearning ensures that information is more impervious to being lost or forgotten, and the forgetting curve for this overlearned material is shallower [4], where forgetting curve hypothesises the decline of memory retention in time (Fig. 6). This curve shows how information is lost over time when there is no attempt to retain it.

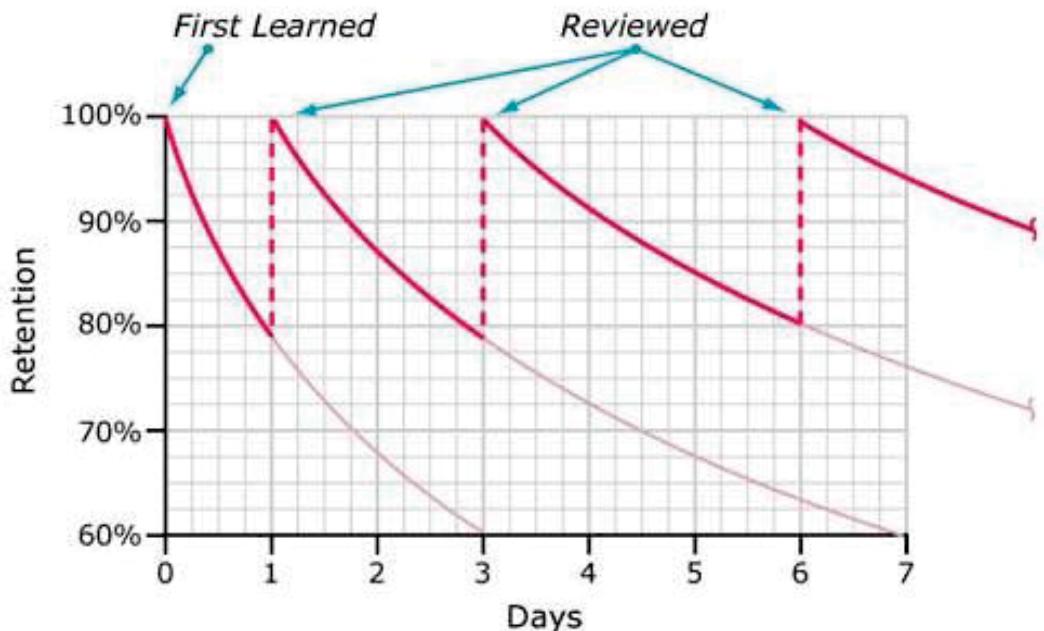


Fig. 6 The forgetting curves

This shows, that as number of reviewing increases, the memory retention strengthen as well. Thus, another theory appeared, called "Spaced repetition technique".

Spaced repetition is a learning technique that incorporates increasing intervals of time between subsequent review of previously learned material in order to exploit the psychological spacing effect [5].

Although the principle is useful in many contexts, spaced repetition is commonly applied in contexts in which a learner must acquire a large number of items and retain them indefinitely in memory. It is therefore well suited for the problem of vocabulary acquisition in the course of second language learning [6].

So in order to build effective mobile trainer, it is obligatory to use this technique to build the order of how the words would appear in the tests.

3 “French Wordbook” Application

In the developed application, all the requirements were implemented, starting with the basic use cases, finishing with the language features. In this part, all the components of application will be reviewed.

3.1 Architectural pattern

The MVVM (Model-View-ViewModel) pattern was implemented as the most convenient for creating Windows Phone 8 application (Fig. 7). Its intent is to provide a clean separation of concerns between the user interface controls and their logic.

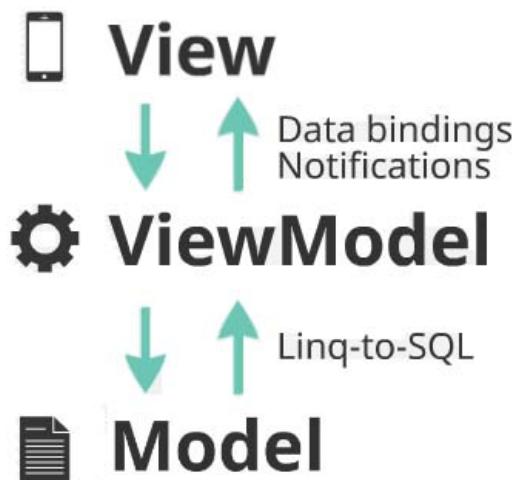


Fig. 7 The illustration of MVVM pattern

The view is responsible for defining the structure, layout, and appearance of what the user sees on the screen. Ideally, the view is defined purely with XAML, with a limited code-behind that does not contain business logic.

The view model acts as an intermediary between the view and the model, and is responsible for handling the view logic. Typically, the view model interacts with the model by invoking methods in the model classes. The view model then provides data from the model in a form that the view can easily use. The view model retrieves data from the model and then makes the data available to the view, and may reformat the data in some way that makes it simpler for the view to handle.

The model in MVVM is an implementation of the application's domain model that includes a data model along with business and validation logic. Examples of model objects include repositories, business objects, data transfer objects (DTOs), Plain Old CLR Objects (POCOs), and generated entity and proxy objects [7].

3.2 Database design and implementation

The database engine on Windows Phone 8 is Microsoft SQL Server Compact, and the very convenient way to interact and operate the database is using LINQ to SQL technology in MVVM pattern. LINQ to SQL is a component of .NET Framework that provides a run-time infrastructure for managing relational data as object

In LINQ to SQL, the data model of a relational database is mapped to an object model expressed in the programming language of the developer.

When the application runs, LINQ to SQL translates into SQL the language-integrated queries in the object model and sends them to the database for execution. When the database returns the results, LINQ to SQL translates them back to objects that you can work with in your own programming language.

The database description is situated in the Model part. Here you can see how the code for the database looks like:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Data.Linq;
using System.Data.Linq.Mapping;
using System.ComponentModel;

namespace myfrenchdictionary.Model
{
    class WordsDataContext : DataContext
    {
        // Pass the connection string to the base class.
        public WordsDataContext(string connectionString) : base(connectionString) {}

        // Tables here
        public Table<WordItem> Words;
    }

    [Table]
    public class WordItem : INotifyPropertyChanged, INotifyPropertyChanging
    {

        private int _ID;

        [Column(IsPrimaryKey = true, IsDbGenerated = true, DbType = "INT NOT NULL Identity",
        CanBeNull = false, AutoSync = AutoSync.OnInsert)]
        public int ID
        {
            get { return _ID; }
            set
            {
                if (_ID != value)
                {
                    NotifyPropertyChanging("ID");
                    _ID = value;
                    NotifyPropertyChanged("ID");
                }
            }
        }
    }
}

```

And then goes all the descriptions of the columns. Annotations are used to mark the attributes of the columns.

Then, Data Context object is stored statically in App.xaml.cs file

```

// The static ViewModel, to be used across the application.
private static WordsViewModel _viewModel;
public static WordsViewModel ViewModel
{
    get { return _viewModel; }
}

```

It is initialized in the same file, in the entry-point method, so that the connection is created when application start, and is closed when the application closes:

```

string DBConnectionString = "Data Source=isostore:/WordsDB.sdf";
WordsDataContext db = new WordsDataContext(DBConnectionString)

```

3.3 Wordbook module

For main screen “pivot” UI pattern was chosen, and it looks like this (Fig. 8):

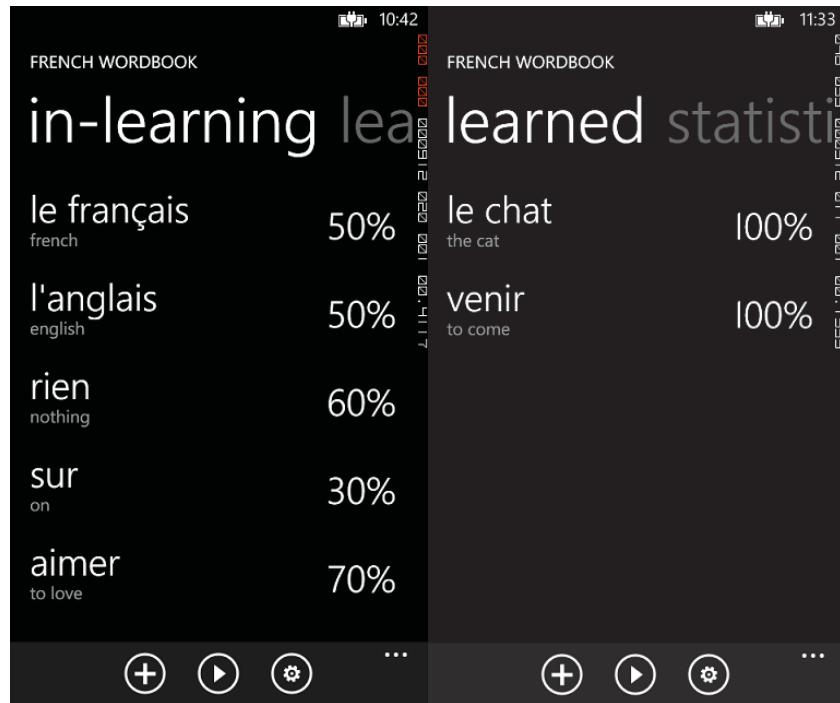


Fig. 8 Main screen

Button “+” stands for adding new word, button “play” starts the training, and the last button opens the setting menu. It is possible to delete the word by using long tap on it (Fig. 9), which is common practice for Windows Phone applications.

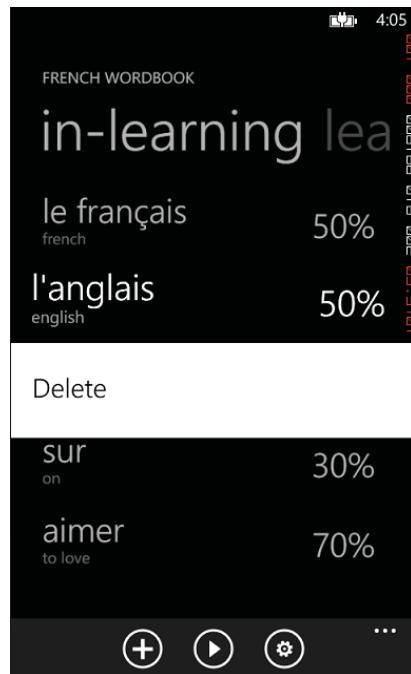


Fig. 9 Deleting the word

Deleting the word is implemented using the context, as it is shown below:

```
string DBConnectionString = "Data Source=isostore:/WordsDB.sdf";  
  
var button = sender as MenuItem;  
  
if (button != null)  
{  
    // Get a handle for the to-do item bound to the button.  
    WordItem WordToDelete = button.DataContext as WordItem;  
  
    App.ViewModel.DeleteWord(WordToDelete);  
}  
  
// Put the focus back to the main page.  
this.Focus();
```

Implementation of DeleteWord method is obvious, so it is no need to present it here.

Adding new word is a two-step process, firstly user have to type the word. Then, if it is possible, algorithm identifies what part of speech is the word, and regarding to this information, proceeds to the next screen. In fact, the only thing that is important to know is whether the word is a noun or an adjective. For a noun, there is the special test for article and another form and for an adjective there is the test for another form. In case if part of speech could not be detected (do not contain an article), program have to know could it be an adjective. The dialog is shown below (Fig. 10):

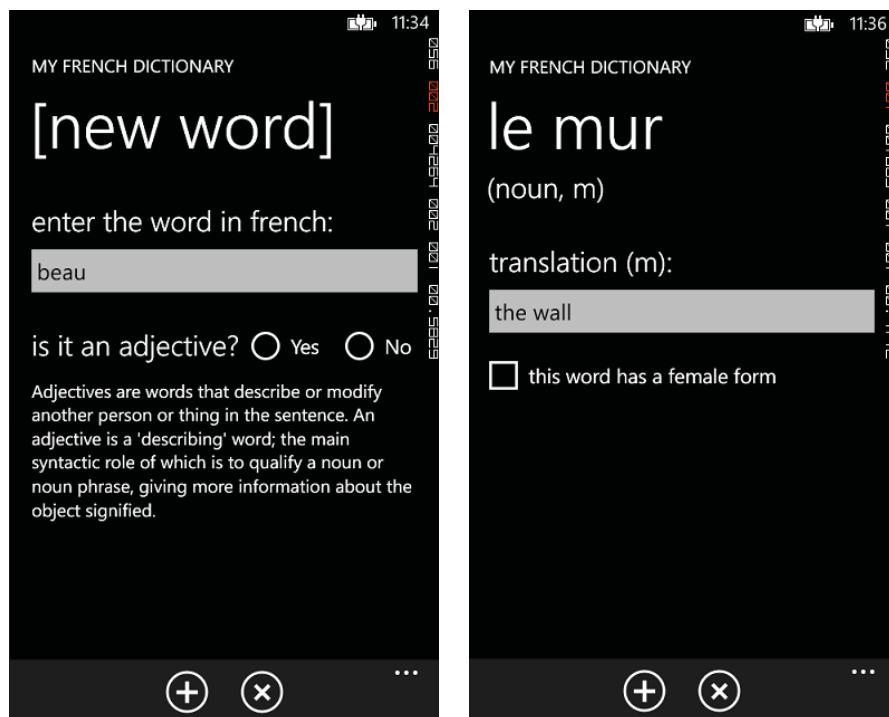


Fig. 10. Dialog in case the word could be an adjective

When all the necessary information is detected, the next screen ask to provide the translation. The future steps in the application development is to provide automatic translation using bing or google API for translating.

In this case, the word “the wall”, which is “le mur” in French has the only one translation and form. But in French, there are many gender-sensitive endings for nouns and adjectives, and that is why there is checkbox for alternative variant.

3.4 Trainings module

If CRUD functions are mostly common for all the typical projects, in the testing module of this application there are lot of algorithmic details which should be viewed.

When user tap on the “play” button, which means the test to be started, program performs some steps before the test screen appears.

Firstly, it checks is there enough words to start the test (5 at minimum). Then, five words are selected, by the following criteria:

- the word can appear in this test session, attribute “AppearAfter” is 0. (this attribute helps with implementing spaced repetition idea, so that the word firstly appears every session, then if the answers are correct, it starts appear less frequently)
- the word has retention level less than 100%

When the words are picked up the words with lower retention level have higher priority. In the another query specific words, like nouns and adjectives with different form are selected. Look at how it is implemented in the InitializeTraining method:

```
public void InitializeTraining()  
{  
    _question = -1;  
  
    // Common Section  
    // -----  
    wordsToTrain = (from WordItem word in WordsDB.Words  
                    orderby word.Percentage ascending  
                    select word).Take(5).ToArray<WordItem>();  
  
    WordProcessing.ShuffleFisherYates<WordItem>(wordsToTrain);  
  
    //Gender Section  
    //-----  
    gWords = (from WordItem word in wordsToTrain  
              where (word.Gender == null && word.WordF != null) ||  
              (word.Gender != null)  
              select word).Take(5).ToArray<WordItem>();  
  
    _genderSection = gWords.Count();  
}
```

As you can see, the Fisher-Yates shuffle algorithm is mixing the order of word then:

```
// Here is Fisher-Yates Shuffle  
private static Random _random = new Random();  
public static void ShuffleFisherYates<T>(T[] array, int Length = -1)  
{  
    if (Length == -1) Length = array.Length;  
  
    var random = _random;  
    for (int i = Length; i > 1; i--)  
    {  
        // Pick random element to swap.  
        int j = random.Next(i); // 0 <= j <= i-1  
        // Swap.  
        T tmp = array[j];  
        array[j] = array[i - 1];  
        array[i - 1] = tmp;  
    }  
}
```

Then, using the special class QuestionItem, one by one – test questions is being transferring to the UI part.

In the following code you can see how test questions with options are created:

```
/  
TestWord = wordsToTrain[_question].Word;  
  
OptionsRAW = (from WordItem word in WordsDB.Words  
    where word.ID != wordsToTrain[_question].ID &&  
    word.Percentage <= wordsToTrain[_question].Percentage  
    orderby word.Percentage descending  
    select word).Take(3).ToArray<WordItem>();  
  
Num = OptionsRAW.Count();  
Options = OptionsRAW.Take(Num).ToArray<WordItem>();  
if(Num < 3)  
{  
    OptionsAddRAW = (from WordItem word in WordsDB.Words  
        where word.ID != wordsToTrain[_question].ID &&  
        word.Percentage > wordsToTrain[_question].Percentage  
        orderby word.Percentage ascending  
        select word).Take(3-Num).ToArray<WordItem>();  
  
    Options = OptionsRAW.Concat(OptionsAddRAW).ToArray();  
}  
else  
{  
    Options = OptionsRAW;  
}  
  
TestRightAnswer=wordsToTrain[_question].Translation;  
  
TestOptions[0] = wordsToTrain[_question].Translation;  
if (Options.Length >= 1) TestOptions[1] = Options[0].Translation;  
if (Options.Length >= 2) TestOptions[2] = Options[1].Translation;  
if (Options.Length >= 3) TestOptions[3] = Options[2].Translation;  
  
WordProcessing.ShuffleFisherYates<string>(TestOptions);
```

Linq queries, that are shown above, forms options in the way, that all the translations which are false are picked up from the words with the lower percentage of retention, so that user will not meet translation which is more familiar than the right one.

Then, there are right and backward tests, with original-translation tasks, and translation-original tasks as well. This are the two first sections.

The third section is a little bit more complicated and consist of questions about the right article (which is actually the questions about the gender form) and questions about another form of word. Here are how these questions are created:

```
if (WGCurrentWord.Gender == null)  
{  
    // TYPE 1  
    // Another form of word  
    TypeOfTest = typeOfTest.AnotherForm;  
    WGOptions = null;  
  
    if (WordProcessing.GetRandomNumber(5) < 4) // 60% of cases  
    {  
        WGender = "m";  
        WGWord = WGCurrentWord.Word;  
        WGRightAnswer = WGCurrentWord.WordF;  
    }  
    else //40% of cases  
    {  
        WGender = "f";  
        WGWord = WGCurrentWord.WordF;
```

```

    WGRightAnswer = WGCurrentWord.Word;
}
}
else
{
    // TYPE 2
    // What is the gender?
    TypeOfTest = typeOfTest.WhatGender;
    WGOptions=new string[2];

    WGWord = WordProcessing.RemoveArticles(WGCurrentWord.Word); // Removing articles
    WGender = WGCurrentWord.Gender.ToString();

    if (WGCurrentWord.Word.StartsWith("les ") || WGCurrentWord.Word.StartsWith("des "))
    {
        // type 3 (les, des)
        // What is the gender pluriel?
        TypeOfTest = typeOfTest.WhatGenderPluriels;
        WGOptions = WhatGenderVariants(WGOptionsTypes.PlurielWord);
    }
    else if (WGCurrentWord.Word.StartsWith("l"))
    {
        // l'
        WGOptions = WhatGenderVariants(WGOptionsTypes.PutArticlesElision);
    }
    else
    {
        // le, la
        WGOptions = WhatGenderVariants(WGOptionsTypes.PutArticles);
    }

    WordProcessing.ShuffleFisherYates<string>(WGOptions);
}

```

The fourth section is about direct translation of the words, without variants, by just typing the correct words. Screens with all four types of tests is on the Fig. 11



Fig. 11 Four stages of training

So one single training consist of four sections with five questions in each. The number of 20 questions in the training was selected as optimal for this unit of activity, to not to bore the user and to give him an opportunity to tune his/her rhythm of learning

4 Conclusions

The lack of applications with described features was revealed, and the necessary functions was implemented, and the future work needs more marketing than programming, which hopefully possible with quite friendly situation in Windows Phone 8 store, because both application and smartphones needs significant amount of PR.

Creating the first version of application was the very first step, and still there are a lot of work to do and a lot of improvement to be completed.

Bing and Google API should be used in order to automate translation of the entered words. Web-service should be created. Therefore, several features are planned, such as:

- Chrome extension for convenient putting words into personal wordbook
- Web-server with back-ups for wordbooks
- Personal page with several wordbook files, including the possibility to name them according to, for example, texts or themes

Of course, other platforms should not be ignored, that is why using tools like Xamarin for multi-platform programming based on C #code should be considered.

Some design improvements and personal style for application in production step may be needed, still this is something to talk about after all other necessary features will be implemented.

References

- [1] Pang Jixian, Xing Zhou, Zheng Fu, English for International Trade: China Enters the WTO. *World Englishes* 21, no.2, 2002.
- [2] http://en.wikipedia.org/wiki/French_language.
- [3] H. Ebbinghaus *Memory: A Contribution to Experimental Psychology*, 1885
- [4] <http://users.ipfw.edu/abbott/120/Ebbinghaus.html>
- [5] Alan D. Baddeley *Human Memory: Theory and Practice*, 1997
- [6] http://en.wikipedia.org/wiki/Spaced_repetition
- [7] <http://msdn.microsoft.com/en-us/library/hh848246.aspx>

MYKHAILO DOROKHOV

Simon Kuznets Kharkiv National University of Economics
Department of Information Systems
9A, Prospect Lenina, Kharkiv
UKRAINE
E-mail: michaeldorokhov@gmail.com

MARIIA HONCHAROVA

H.S. Skovoroda Kharkiv National Pedagogical University
Department of Roman Languages
29, str. Artem, Kharkiv
UKRAINE
E-mail: mariehoncharova@gmail.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

My Virtual Trainer An application to get you fit in no time

**Bogdan – Mihai Fieraru, Vasile – Sebastian Vidrea, Sorina – Gabriela Irimie
Teacher Coordinator: Anca Vasilescu**

Abstract

Abstract: This is an application developed for Android smartphones using Java programming language. It is designed for people who want to start to train in a gym and its purpose is to generate a training considering the user's profile or allow him to customize his own. This application also provides the option to track user's progress and an overall view of all the exercises organised by muscle groups.

1 Why we made our choice?

The obesity is a common problem in our days. More and more teenagers have weight problems and tend to choose fast-food over healthy food. When they try to make a change in their life by going to the gym, they face two main problems: the first one is that they don't have any knowledge about the exercises and they either have to pay a personal trainer or work the exercises incorrect. The second problem is that even if they work hard, they probably don't know how to eat properly and their work is in vain. Regarding these problems and the fact that almost everyone has a smartphone, we decided to make an application to help those people get fit without paying money on trainers or nutritionists.

2 About the application

2.1 Technology that we used

The application was developed using Java programming language for Android and SQLite for storing exercises in a database. For communication with the server we used JSON(JavaScript Object Notation) and PHP Web Service. For icons and layout we used Photoshop, and also we used Tortoise SVN to work on the same project simultaneously.

2.2 System requirements

The system requirements for this software are low cost:

- smartphone with Android 4.1.2 JellyBean
- internet connection (for first run and for update)

2.3 How the application works

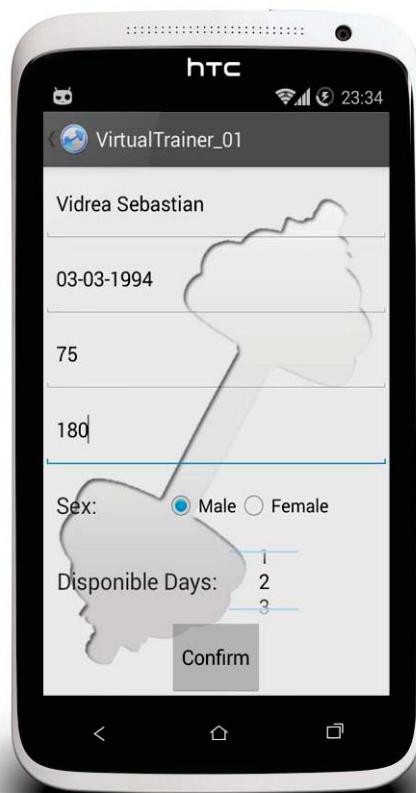
The application is easy to use. When the user starts it for the first time, he is sent to a Create Account Activity, where he needs to fill information about himself: age, weight, height, gym experience etc. The software is capable of generating trainings for the user according to the data introduced by him in previous activity. If he is an advanced gym client, he is allowed to customize the training by his own will, for example the user is able to work more than three muscle groups in a single training, or only one.

The user also has the possibility to see all the exercises stored in the database, organised by muscle groups. When he selects an exercise, a description activity starts and the user can see information about each one of them: a picture or a tutorial movie about how the exercise should be made, a short description, recommended number of reps.

The last option is the Progress Status, where the user is able to see how many trainings he already made and how many he still needs to make. Also, he could see his personal progress, if he lost weight, or got stronger or faster and compare his results with previous weeks.

Furthermore, through the settings panel user can change details about his profile, update the exercises list from our remote database and more other slightly changes to the interface.

2.3.1 Create Account Activity



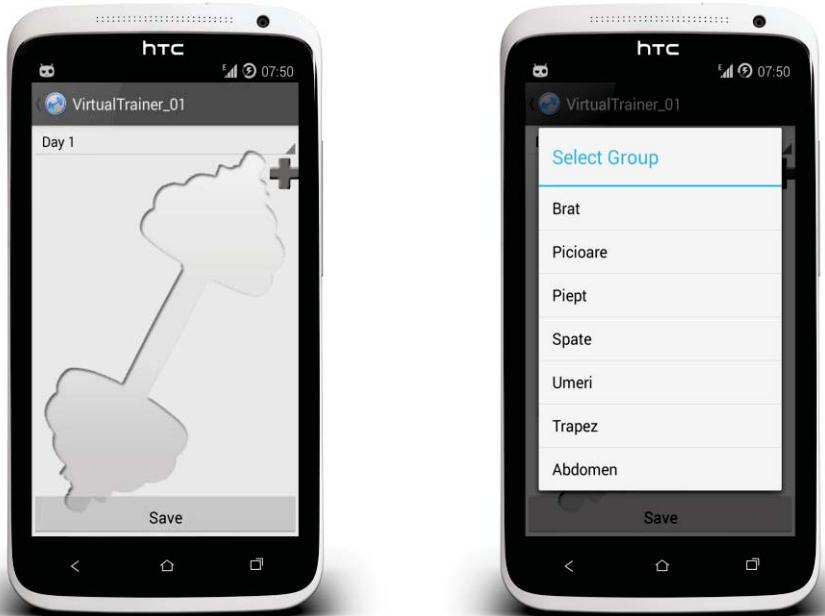
The user is sent on this activity only when he starts the application for the first time. He should complete the indicated fields with his personal information: Name, weight, height, sex, available days for gym, age. After he fills the data, he needs to press Confirm button and his profile will be created.

2.3.2 Generate Training Activity



In this activity, the user will see the training generated for him by an algorithm based on the informations he provided at the first launch. Here exercises are sorted by their muscle groups but also by days. Furthermore, the user can select any exercise and he will be sent to an activity where he can see informations about that specific exercise.

2.3.3 Setup Training Activity



This activity is recommended for more experienced users who went to the gym before and they already know to set their training. It is really simple for a user to setup his own training; he just has to follow few steps: first he needs to press the plus sign then, a pop-up with all the muscle groups will appear from where he can choose. After that he will need to select the desired exercises for each of the muscle groups previously chosen and press Save when ready.

2.3.4 Exercises Activity



In this activity called Exercises user is able to see the full list of exercises existing in our database and additional information if he chooses to select one of them. This serves well as documentation.

2.3.5 Current progress Activity

Here user can see his progress from the point when he first started to use this application until the time he enters this activity. He will see the exercises he finished at that point and those that still need to be made. User will also be able to see here his current growth (if he lost or gained some weight and also a photo gallery with pictures of himself he will be asked to optionally take monthly).

3 References

- [1] Bruce Eckel, *Thinking in Java*, Prentice Hall, 2006.
- [2] <http://developer.android.com/training/index.html>
- [3] Apress Android Apps for Absolute Beginners (2011)

Bogdan-Mihai FIERARU
Transylvania University
of Brașov
Departament of Mathematics and
Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
Romania
E-mail: fieraru_bogdan2000@yahoo.com

Vasile-Sebastian VIDREA
Transylvania University
of Brașov
Departament of Mathematics and
Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
Romania
E-mail: vidreasebastian@gmail.com

Sorina-Gabiela IRIMIE
Transylvania University
of Brașov
Departament of Mathematics and
Informatics
Str.Iuliu Maniu Nr. 50 Brasov , 500091
Romania
E-mail: sorina_irimie@yahoo.com

Plagiarism detector – Document Processing

**Floreia Andrei George, Marica Constantin Ionuț, Liță Alexandru Marian, Dragomir Ionuț
Teacher Coordinator : Constantinescu Zoran, Nicoară Simona, Moise Gabriela**

Abstract

In the academic world, the most common type of fraud is plagiarism, which means copying certain parts (paragraphs, pictures, ...) belonging to other authors into your own document and not declaring (or correctly declaring) the author.

The application was built entirely by students belonging to the T.I.M.F department, under the supervision of the teachers from the Computer science section.

The purpose of the application is to provide an accessible, on-line method of checking various documents in an attempt to deconspire the above mentioned type of fraud.

This article describes the architecture and basic functioning of Plagiarism detector, and focuses on document processing.

1 Introduction

Plagiarism, according to [1] originates from the Latin word *plagiarius* 'kidnapper', and means *the practice of taking someone else's work or ideas and passing them off as one's own*.

The act of plagiarism is not considered a criminal act,

The traditional way of checking documents for plagiarism was for an experienced reviewer to read the document page by page and notice where there were discrepancies between the author's writing style and a certain paragraph. Afterwards the reviewer must search that paragraph on different sources in order to confirm.

This method is error prone, and time consuming, and thus a better solution needed to be found.

Any automated detection system has the following general requirements:

- A document processor that is capable of filtering out characters or symbols that could lower the precision rate
- One or more search solutions and a search database
- One or more comparison methods capable of generating a viable score

This document focuses on different methods used to filter documents of any elements that might disrupt the proper functioning of search and comparison algorithms alike.

The article continues with *Related Work* where other plagiarism detection tools and auxiliary software are described. Afterwards, *Architecture* is the section where the basic structure of the application is presented.

All processing stages and short examples are presented in the *Document processing system* section. A detection example alongside application parameters and a list of software dependencies are

presented in the 5-th section, *Software Usage*.

The final section, *Conclusions* tries to draw a line between the application's features, current limitations and possible extensions!

2 Related Work

There have been numerous attempts to automate the detection process, and for that reason, today there are many software solutions designed in this purpose. There are also many auxiliary software solutions that must also be mentioned because they enhanced and in some cases made the process possible.

Plagiarism detection tools :

1. Plagiarism Detector : payed product (made by SkyLine, LLC), installed on the user's computer, that according to it's website[2] has the following features:
 - has a search database of over 7 billion on-line sources
 - uses Google, Bing and AltaVista as search engines
 - Microsoft Office integration
 - Automatic references detection

Compared to our application, the only significant advantage of Skyline's product is automatic reference detection which improves score accuracy by skipping referenced frazes.

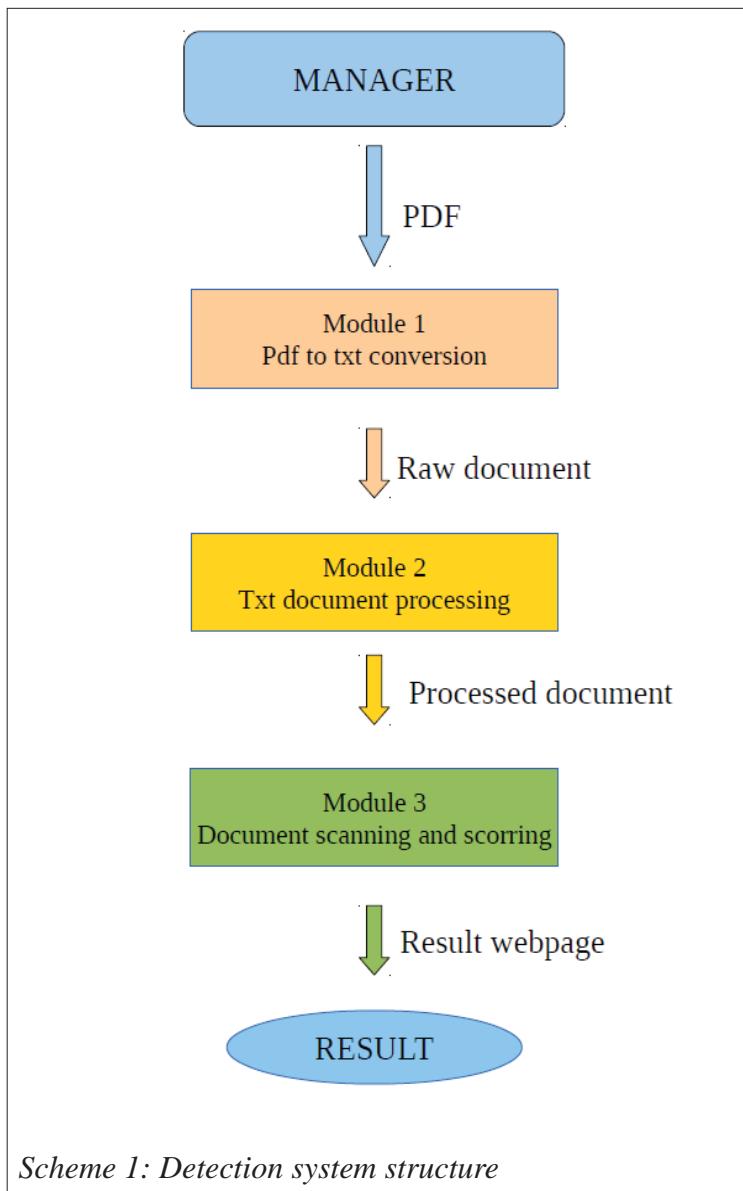
2. PlagTracker [3] offers a simple detection system, based on web search, and unlike SkyLine's Plagiarism Detector, is completely free.

The auxiliary software that is widely used in plagiarism software, is listed below :

1. Pdf – Initially a proprietary only software owned intended to ease written document transfer between computers using different hardware / software sets, Adobe Pdf has become an open ISO standard (ISO 32000-1:2008). Some parts of the Pdf standard still remain proprietary (e.g. Adobe XML Forms Architecture, Adobe JavaScript).
2. BLEU - According to article [4], BLEU stands for BiLingual Evaluation Understudy, and proposes a method of evaluating the quality of automatic machine translation. The advantages of this method are speed, low usage cost and language independence. The article proves the advantages of the *n-gram* comparison algorithm.
3. Regex [wikipedia] – The most widely used string pattern matching system, REGular Expressions were conceived in 1956 by American mathematician Stephen Kleene with *regular sets*. One on the first software implementations was made by Ken Thompson with the *qed* text editor, and afterwards came into public use with a series of Unix applications like *ed*, *grep*, *vi*, *awk*.

3 Architecture

The detection system is composed of several interdependent modules that take a given input .pdf file through several processing stages as shown in Scheme 1 :



Manager refers to a Python script responsible for running the appropriate script (module) on a given file, in order to ensure the quality of the end-result.

The first task at hand is converting the initial Pdf document into a simple text file, job done by *Module 1* which is actually the PdfToText converter application,

Text document processing is done by *Module 2*, a Python script later described in greater detail in this document.

Module 3 uses the processed document as input and scans it with 3 different search engines and writes a Html file containing the initial text and the similarity score of each sentence. (if greater than 50%).

Scheme 1: Detection system structure

4 Document processing system

The output of *PdfToText*, and for that matter any other pdf conversion tool can be unexpected because not all Pdf exporter applications fully respect the *ISO-300** standard released by Adobe.

Various types of errors may appear after conversions, these are just the most common :

- Missing characters due to various font errors
- Surplus characters, often outside of the current char-set, for the same reason as above
- Big gaps between paragraphs or table elements
- One letter words (x, y, z) that generally appear after converting scientific documents

In order to fulfill the objectives mentioned above, the processing module (2) makes full use of the *regular expression* implementation available native in Python2 as the **re** module. The end result is called a processed document.

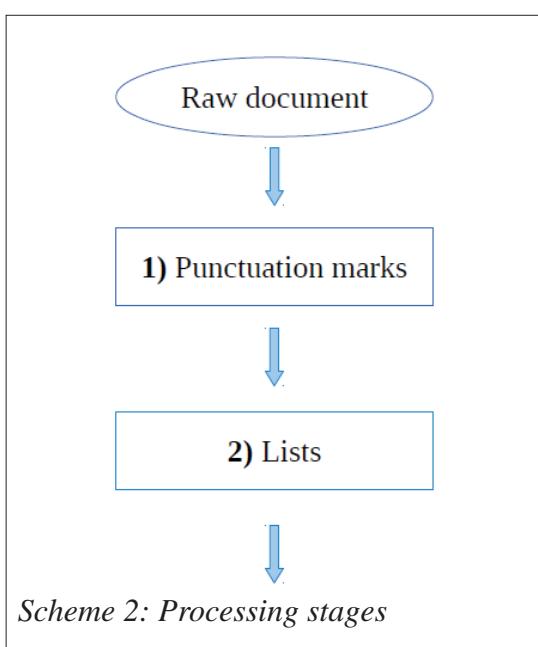
Definition 1 : A text document which contains only 1 sentence (no matter what length) per line is called a processed document

The module is intended to filter out any text components that might prevent text searching or change the accuracy of the score system.

Document formatting is a key element for the application, because the detection module calculates the document score based on a arithmetic mean of the contained sentences. Thus, it is vital for the sentences to be clearly delimited one from another.

The most useful, in terms of score, are longer sentences, but they also have the downside of being separated across more than 1 row. Because of that, the *main* task of the module is to **join rows** that span across multiple rows.

However, in order to achieve this, the document must undergo a series of transformations in order to ensure that sentences are correctly separated. Scheme 2 shows the processing stages :

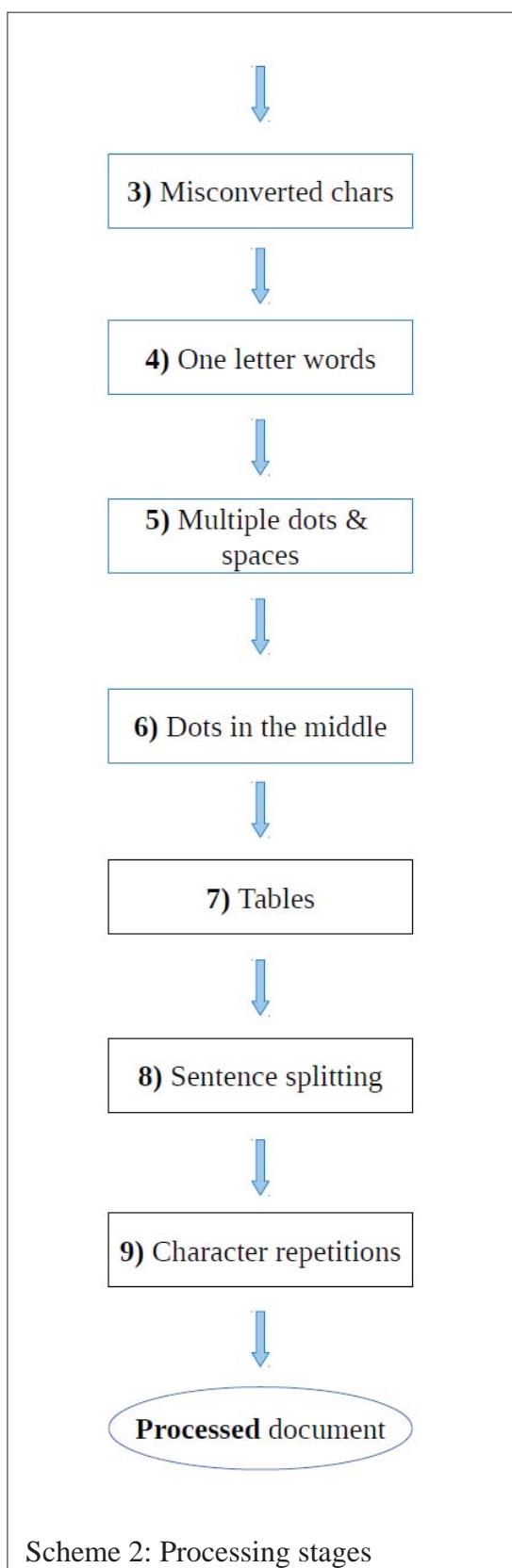


1) The first transformation to be performed is to replace all end-sentence punctuation marks (; : ! ?) with . (dot) in order to simplify the splitting phase. The regex expression used to detect punctuation marks is : '\s*(;|:|!|?)'

2) List elements can also contain significant information and therefore must be added to the general score and that can only be done by treating each element as a separate sentence. In order to avoid any detection errors, lists have to be trimmed of headers. The processing script uses *regex* substitutions in order to remove the headers of the following types of lists :

- digit numbered lists, ex : 1) 1. (1)
- letter numbered lists, ex : a) A) a. (a) ...
- roman numbered lists, ex : I) II) V) ...

- bullet type lists



3) During the conversion process from Pdf to txt, various characters can be misconverted due to font incompatibility, like in the example :

This is an incorer example

In order to prevent encoding errors, such characters must be detected and eliminated (by replacing with a space or null character). In the regex standard, this is a straight-forward task simply because it can detect characters outside of a given range. The following example was obtained after processing the above sequence with : '([^w\.-\[\]\]|_){}'

This is an incorer example

It is obvious that neither this version of the sentence is lexically correct but it can be correctly encoded in UTF-8 and it can be processed by search engines.

4) Mathematical formulas are often encountered in scientific papers and after conversion, these will rarely resemble the original form, mostly because operators and brackets are removed at step 3). Given the following expression :

f x sin x 3 cos x

after step 3) it would be transformed to :

fx sin x 3 cos x

and after passing through `\s+[^aeio\d.]\s+` :

sin 3 cos

The objective of this step is not to produce a readable mathematical expression, but to reduce the number of meaningless (a, e ,i ,o are the only accepted words in the Romanian language) one letter words.

This step is explained in greater detail in the *Functioning* section of the document.

5) Removing multiple dots next to each other is also a vital filtering stage, given the fact that sentence splitting is done at dots. The regex expressions `\.{2,}` and `\s{2,}` are used to clear repeated dots and spaces respectively in char sequences like :

This is an error prone example.

That after the regex replacement resemble :

This is an error prone example

6) Abbreviations use dots to separate letters and if they are not removed, the sentence splitting phase will result

in one letter sentences, which are useless for detection purposes. The regex used in this case, `\.(?=\\w)` detects and replace sequences like the following :

Most Unix systems use the G.R.U.B boot-loader.

with :

Most Unix systems use the GRUB boot-loader.

7) Tables are also important data containers and therefore the data in each cell must be scanned! The output of *PdfToText*, generally resembles the source table, cells being read left to right, top to bottom, and written on separate lines with an empty line in between. The processing script removes the empty lines and adds a dot at the end of the line in order to mark them as separate sentences. The following table better illustrate the changes :

input	output	
	Pdf - > txt	processed
<i>line1_col1 line1_col2 line1_col3</i>	<i>line1_col1</i>	<i>line1_col1.</i>
<i>line2_col1 line2_col2 line2_col3</i>	<i>line1_col2</i>	<i>line1_col2.</i>
<i>line3_col1 line3_col2 line3_col3</i>	<i>line1_col3</i>	<i>line1_col3.</i>

8) The separation of sentences is the most important step in the processing module, as it offers a way to search and calculate a differential score on every integral sentence. This step is executed in 2 stages, simply because all the steps above processed lines stored in a array. The first stage is to join the array of lines into a single string (pdftext) and afterwards split it again at dots and store the results in a sentence array (frazz) :

```
pdftext = " ".join(linie);
frazz = pdftext.split(".");
```

9) The last step before writing the processed document is to check for 2 or more characters next to each other. These characters are most often the side effects of previous replacements.

It must be noted that characters refers to anything outside the interval {letter,digit,dot,minus,space '[', ']' } which is expressed in the regex language as: `([^\\w\\.\\-\\[\\]]_\\s){2,}`

5 Software Usage

The *manager* script provides each module listed in Scheme 1 a set of 2 parameters :

- file_id – the index in the pp_files database table.
- input_file – the file that will be processed
- -v – the verbosity level, used mostly for debugging

Under normal usage, the *manager* requests a list of files that have not passed through all processing stages i.e there is no result.

The status of each file is saved in the database and can have one of the following values :

0. YYYY-MM-DD HH-MM-SS.pdf - file uploaded by the user and renamed with the current timestamp
1. file[id].txt – raw text conversion from the pdf file with the index of *id*
2. file[id].txt1 – processed document, result of running the processing module on the file with index *id*
3. file[id].html – the result of running the detection module on the file with index *id*

The end result of the detection module can be defined as :

Definition 2 : A web page that contains the contents of the scanned document and tags that offer information about the source of inspiration and the score is known as a result webpage.

A small case study will be conducted using the following figures, where the example document test.pdf is taken through all the modules.

Figure 1, illustrates the test pdf in it's original, unprocessed state.

TEST DOCUMENT TITLE

This is the first paragraph. It was written by the author, without any inspirational sources whatsoever!

In the second paragraph, there is a list :

- This is the first element in the list
- This is the second element in the list
- This is the third element in the list

The contents of the next paragraph were copy pasted from Wikipedia and should appear red in the final result webpage :

PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page. PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

$\text{sqrt}(r^2 * (\cos(t)^2 + \sin(t)^2)) = R^2$ – an example of a matlab / octave expression.

Now there will be a table :

Cell1	Cell2
Cell3	Cell4

The dots must be removed

So should these ,-----

Figure 1: Original PDF document

After passing the original pdf through module 1 (PdfToText) we obtain the following result :

```
[TEST DOCUMENT TITLE
This is the first paragraph. It was written by the author, without any inspirational sources
whatsoever!
In the second paragraph, there is a list :
.

This is the first element in the list

.

This is the second element in the list

.

This is the third element in the list

The contents of the next paragraph were copy pasted from Wikipedia and should appear red in the
final result webpage :
PHP code is interpreted by a web server with a PHP processor module, which generates the
resulting web page. PHP commands can be embedded directly into an HTML source document
rather than calling an external file to process data. It has also evolved to include a command-line
interface capability and can be used in standalone graphical applications.
sqrt(r ^ 2 * (cos (t) ^ 2 + sin(t) ^ 2) ) = R ^ 2 - an example of a matlab / octave expression.
Now there will be a table :
Cell1
Cell2
Cell3
Cell4

The dots must be removed .....
So should these .....
```

Figure 2: Output of module 1 - PDF to txt conversion

As it can be noticed in Figure 3, the output of module 2 (processing stage) is more condensed and search friendly.

```
[TEST DOCUMENT TITLE This is the first paragraph
It was written by the author without any inspirational sources whatsoever
In the second paragraph there is a list
This is the first element in the list
This is the second element in the list
This is the third element in the list
The contents of the next paragraph were copy pasted from Wikipedia and should appear red in the final result webpage
PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page
PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data
It has also evolved to include a command-line interface capability and can be used in standalone graphical applications
sqrt(2 cos(2 sin(2)) ^ 2) an example of a matlab octave expression
Now there will be a table
Cell1
Cell2
Cell3
Cell4
The dots must be removed So should these.
```

Figure 3: Output of module 2 (processing stage)

The output of module 3 (scanning) represents the end result of the application which is presented to the user as a web page.

The red lines shown in Figure 4 are paragraphs that after being searched over the web returned a similarity score greater than 50%. These lines can be clicked in order to view search details, as shown in Figure 5.

PlagiarismDetector

designed for education purposes only

TEST DOCUMENT TITLE This is the first paragraph
It was written by the author without any inspirational sources whatsoever
In the second paragraph there is a list
This is the first element in the list
This is the second element in the list
This is the third element in the list
The contents of the next paragraph were copy pasted from Wikipedia and should appear red in the final result webpage
PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page
PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data
It has also evolved to include a command-line interface capability and can be used in standalone graphical applications
sqrt 2 cos 2 sin 2 2 an example of a matlab octave expression
Now there will be a table
Cell1
Cell2
Cell3
Cell4
The dots must be removed So should these

Figure 4: Output of module 3 (scanning) - end result

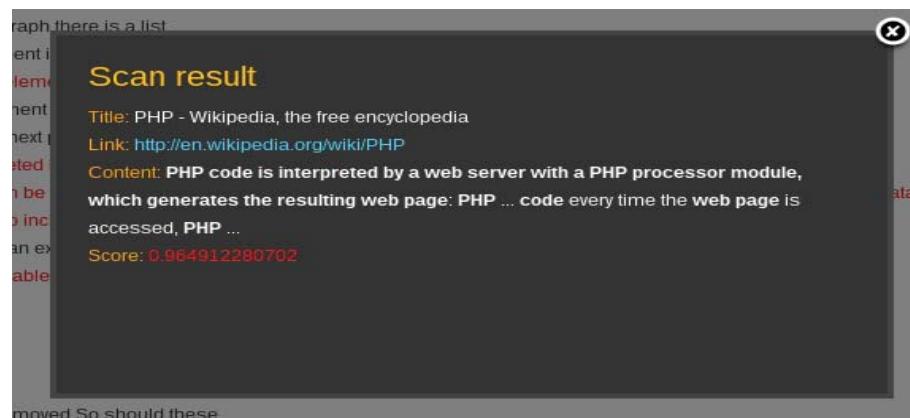


Figure 5: Result details popup

Software used for development and usage

Plagiarism detector.. can be run on any system that meets has the following software dependencies :

- For the front-end, browser that can interpret the following :
 - Php 5
 - Javascript
 - Jquery
- For the back-end, Linux machine that has the following applications installed :
 - PdfToText
 - Python2
 - Apache
 - Mysql

Although the application was designed to run on a Linux workstation, the portability of Php and Python scripts ensures compatibility with other operating systems with minimal modifications.

Current limitations

The current limitations of the project generally refer to document formatting, score calculation and of course web search.

The major limitation regarding document formatting is directly proportional to PdfToText's font limitations, especially when dealing with eastern European languages like Romanian, whose meaning depends on diacritics like š, ţ, ā, ī, â. Most western encodings like ISO-8859-1 do not include eastern diacritics and the above-mentioned letters are most often trimmed of diacritics and thus most of the meaning is lost (when passed through a search engine).

The second limitation is related to the first, and is also frequently encountered in text editors as it refers to variations in diacritic notation. The letters ţ, š can be written either with a short or long mark underneath. For the human reader, this is barely noticeable, but for a search engine these appear as 2 different characters.

Possible extensions

One way of increasing detection precision, as suggested in [5], is implementing a lematization algorithm, that can return the root of a given word. For example, the following sentences would have a low score (4/10 = 40%) :

This sentence has a low score and won't be detected. / That sentence had the lowest scoring and can't be detected.

The score was calculated by successively matching words (1-gram) [4]. The next 2 sentences, obtained after word rooting have a similarity score of 7/10 = 70% :

This sentence have a low score and won't be detect / That sentence have the low score and can be detect

This is one method of increasing the precision of score calculations in an attempt to lower the

influence of word variations.

A preliminary implementation of this method has been tested during the project, but not included in the processing steps due to the error frequency.

The implementation consisted of a pair of root-flexioned database tables extracted from DexOnline [6]. The two tables were linked by a common column (root_id) that allowed extracting the root word of a given extended word. Unfortunately the root table did not contain unique root words, as many of the words were extended.

6 Conclusions

During previous years, quantity dominated against quality in almost every domain including academic research. In recent years however the tide has turned in favor of quality, due to a series of economical, social and political changes.

Various software solutions appeared in order to assist research reviewers in ensuring the quality and originality of research papers. For the moment most of these are paid services, and the price depends almost always on the type of sources used for comparison.

Although the current project is not flawless, it represents a step forward in an attempt to offer a simple, free solution to a never ending problem!

Bibliography

- [1] Oxford English Dictionary, <http://www.oxforddictionaries.com/definition/english/plagiarism?q=plagiarism>
- [2] Plagiarism Detector, <http://www.plagiarism-detector.com/>
- [3] PlagTracker, <http://www.plagtracker.com/>
- [4] Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu , BLEU : A Method for Automatic Evaluation of Machine Translation, 2011
- [5] Miranda Chong, Lucia Specia, Ruslan Mitkov : Using Natural Language Processing for Automatic Detection of Plagiarism, Research Group in Computational Linguistics, University of Wolverhampton, UK, <http://www.plagiarismadvice.org/research-papers/item/using-natural-language-processing-for-automatic-detection-of-plagiarism>
- [6] DexOnline, <http://dexonline.ro/>

FLOREA Andrei George
Article & application author
Petroleum-Gas University of
Ploiești
Information Technology,
Mathematics and Physics
ROMANIA
E-mail :
george91ro@yahoo.com

MARICA Constantin Ionuț
Application author
Petroleum-Gas University of
Ploiești
Information Technology,
Mathematics and Physics
ROMANIA
E-mail:
m_costyn_i@yahoo.com

LIȚĂ Alexandru Marian
Application author
Petroleum-Gas University of
Ploiești
Information Technology,
Mathematics and Physics
ROMANIA
E-mail:
alexandrumarianlita@gmail.com

DRAGOMIR Ionuț
Application author
Petroleum-Gas University of
Ploiești
Information Technology,
Mathematics and Physics
ROMANIA
E-mail:
ionutz_dimy@yahoo.com

QuickThink

**Daniela Ilie, Ionuț Hodină, Lucian Oprea
Teacher Coordinator: Daniel Hunyadi**

Abstract

We think that mobile devices will be in short time the center of what means high level technology and the best way to use it daily.

Technology will be more and more present in our life and people will assimilate it as 'must-use' or 'must-have' to make easier and safer our life. With these said, entertainment can't be left behind.

When we speak about making an app it is not important anymore the mobile platform chosen. The real utility and simplicity to use we think it's the main reason for starting developing. There are a lot of apps that facilitate fetching and sharing information, but the ordinary user should be concerned by the level of thinking involved when using these. There is a big difference between having a car with sensors for collisions, parking assistant and a full self-driving one.

Our application represents a small game that provokes your quick thinking and shows you how fast your brain reacts at an easy question. We are students, we like to race each other so, for us, this game is catchy and challenging.

1 Introduction

In everyday life fast reaction time can protect you from small accidents - like a ball in your face. It brings you sometimes big rewards. Usually the ones who answer fast are the ones who have advantages in life - "to be the first".

Frequent studies have shown that stress and pressure can significantly affect decision-making and memory. With this application we want to stimulate the thinking of person, to help them to pay attention to details which can be seen at a first view. [1]

2 Design and implementation

Mobile apps were originally offered for people needs to have everywhere their preferred app, like email, calendar, games or contacts. The popularity of mobile apps has continued to rise, now a lot of app are created for user entertainment. From an OS to another, developers must obey certain rules in order to ensure that the applications they have written are accepted to be published on the Store. [2]

2.1 Technologies used

Visual Studio represent a collection of tools and services for developing applications for windows, web, devices and cloud. Offers a development environment for building apps targeting the full range of Microsoft platforms, Windows Phone apps and others. Microsoft Visual Studio has an editor, a compiler and a designer for many programming languages. [3]

The Programming Languages included in Visual Studio are: Microsoft Visual C++, Microsoft Visual C#, Microsoft Visual Basic, Microsoft Visual Web Developer and Team Foundation Server.

Microsoft Visual C# is a programming Language object oriented, every function, method, attribute you write it has to be defined in the interior of a class. It uses the Microsoft .Net Framework, a collection of classes the users can use and many data structures. Concepts like class, object, polymorphism, inheritance must be known by a programmer in C#.

When you choose to develop an app on windows phone you have two choices, XAML/C# or HTML/JavaScript. For our application we choose the first choice XAML/C# because it is better object oriented, a better community knowledge support, easier debugging, easier UI Binding.[4]

Extensible Application Markup Language, or XAML, is a declarative markup language. As applied to the .NET Framework programming model, XAML simplifies creating a UI. It can initialize objects, set properties of objects and defines the visual appearance.

Azure is Microsoft's application platform for the public cloud. It can be used in many different ways: to store data for app, to create virtual machines for development and test or to run applications or you can use Azure to build massively scalable applications with lots and lots of users. Azure is based on Windows ,so is easy to use. You can easily create app that run reliably and scale from 10 to 10 thousand to 10 million users, without any additional coding, you can set the number of processors your app need and you pay just for what you use (if your app is active one day per month then you are not bound to pay for the whole month, but only for a day). [5]

When we want to create a Mobile App, Azure Mobile Services helps by providing turnkey way to store data in the Cloud, users authentication, and send push notifications. Because of the Native client libraries for Android, iOS, HTML/Java Script, Windows Phone and Windows Store it is easy to power apps available on all major mobile platforms.

Using Mobile Services you can do a couple of things like:

- To store data in the cloud using Azure SQL database, blob storage, table storage, also it has the option to connect to on-premises databases so that sensitive data can be accessed on mobile devices without saving that data in the cloud. This variety of data makes it easy to add a global leader board to a game.
- To authenticate users easily through the well-known identity providers like Facebook, Twitter, Microsoft or Google
- To send push notifications: (Push notifications happen when a service sends an update to the device, they can happen at any time with a great frequency).
- Consume your favorite services. [6]

2.2 UI

In the Main Page of the app are located the logo and 2 buttons as described:

- new game/continue game - this button will introduce user in the game, beginning with new game data or continuing from last save

- high scores - in this page it's shown a table with best saved scores and the rank of the current user

Game Page, basically, contains several blocks which show the state of the game:

At the bottom of the page:

- score block - indicates the current score of the user calculated on the level and number of correct questions answered

- time block - indicates the time left for the user to give correct answers
- correct questions / total questions answered block

Content panel, on which user eye will focus, contains:

- question block - straightforward question or tip to indicate the right answer
- a Stack Panel where are placed 3 buttons from which only one is correct and other 2 buttons with quite random content generated, appropriate enough content to make choosing right answer hardly

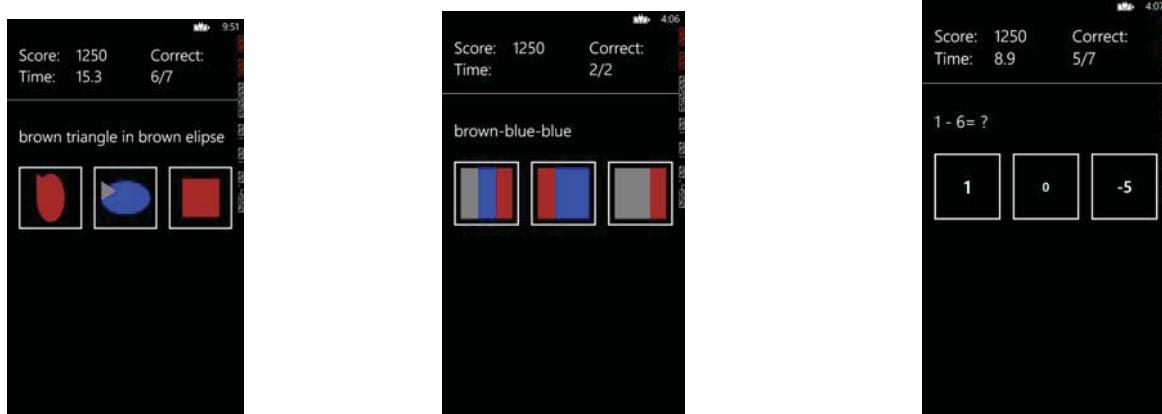


Figure 1. Game Page

2.3 Game rules

The game starts with score set to 0 and a given left time of 10s. Every time when the user chooses a right answer, the user receives 3 more seconds. Every time when the user chooses wrong answer, time left it's decreased by 2 seconds. In this way, the user is constrained to pay attention to every single one question.

Game starts with the level set to 1. After a number of right answers given, the level can increase up to 5. After every right answer given the score it's increased with an amount of points dependent of current level. The game ends when the time it's elapsed.

2.4 Code design

The below UML class diagram shows the basic structure involved in the app. Every question class inherits from a base class, named "QuestionBase" which consist in:

- Question - string property with specific template (example given:"1+2=?")
- UpdateDifficulty (int level) - current level is sent to update difficulty of questions
- GetRightContent() - the button with the right content owns this method result content

- GetRandomContent() - similar content as the right answer it's built and retrieved
Abstraction, Encapsulation, Inheritance and Polymorphism are Object Oriented Programming principles used.

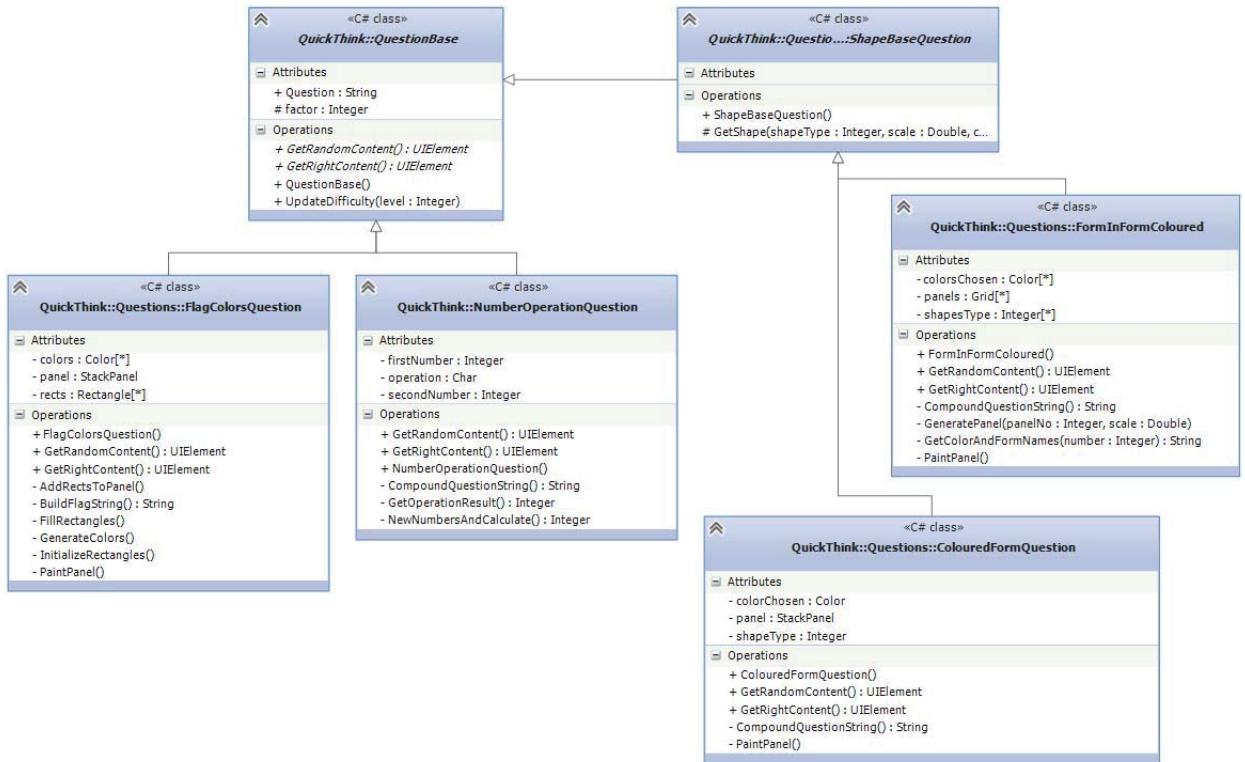


Figure 2 UML Diagram

3 Conclusion and future features

“Things don't have to be complicated in order to be attractive...”

QuickThink is a Windows Phone App in which people have the possibility to answer at different types of questions, which are created to lead somebody in a wild-goose chase.

In the near future, the app will be improved with new template question, more attractive design, a better user "sign in", new high score table and upload it to Store. We also want that application to have support for multiple languages.

4 References

- [1] <http://www.scientificpsychic.com/graphics/index.html>
- [2] http://en.wikipedia.org/wiki/Mobile_application_software
- [3] <http://www.visualstudio.com>
- [4] [http://msdn.microsoft.com/en-us/library/fx6bk1f4\(v=vs.90\).aspx](http://msdn.microsoft.com/en-us/library/fx6bk1f4(v=vs.90).aspx)

[5] <http://www.techrepublic.com/blog/10-things/10-reasons-to-use-azure-for-your-cloud-apps/1282/>

[6] <http://azure.microsoft.com/en-us/services/mobile-services/>

Daniela ILIE
“Lucian Blaga” University of
Sibiu
Faculty of Science
Departament of Informatics
Str. Dr. I. Ratiu, No. 5-7, Sibiu,
550012,
ROMÂNIA

Ionuț HODINĂ
“Lucian Blaga” University of
Sibiu
Faculty of Science
Departament of Informatics
Str. Dr. I. Ratiu, No. 5-7, Sibiu,
550012,
ROMÂNIA
Email: ionut.hodina@gmail.com

Lucian OPREA
“Lucian Blaga” University of Sibiu
Faculty of Science
Departament of Informatics
Str. Dr. I. Ratiu, No. 5-7, Sibiu,
550012,
ROMÂNIA
Email: ionut.hodina@gmail.com

Measuring the Similarity of Images Compressed Using JPEG Algorithm

Milos Jordanski, Nebojsa T. Peric, Radmila Sekulic

Teacher Coordinator: Milan Tuba

Abstract

The purpose of this paper is to present the similarity of images compressed using JPEG image compression algorithm. The measure of image similarity is very important for image compression algorithm. JPEG is commonly used method of lossy compression for digital images. The degree of compression can be adjusted, allowing a selectable compromise between image quality and size. JPEG uses a discrete cosine transform on blocks of 8*8 pixels transforming spatial domain into the frequency domain. The main lossy operation in the whole process is done by neglecting less important frequency coefficients. Which coefficients are less important is determined by quantization matrices. Matrices can be adjusted for different types of images. The similarity of the original and compressed image depending of different quantization matrices are described. Software system for the evaluation of the quality of compressed JPEG images is developed with corresponding graphical user interface.

Keywords: Digital image, Image compression, JPEG algorithm, Measure of similarity

1 Introduction

Digital images are very popular and extensively used. Today, there is almost no area of technical world that is not impacted in some way by digital image processing. The problem with digital images is that the representation of images in digital form requires a lot of data. One solution for this problem is the use of compression techniques. Lossy compression algorithms have very high degree of compression with minimal loss in quality during decompression.

The term “JPEG” is an acronym for the Joint Photographic Experts Group, which created the standard. In computing, JPEG is commonly used method of lossy compression for digital images. It is not a fixed standard, rather it is a set of recommendations. The quantization is the most varying part of the algorithm, because there are different ways of doing quantization that depend of an image. There is not a unique best way to doing quantization. Quantization can be specifically adjusted to some picture. It is possible to have different quantization in different areas of the image. Some quantization matrices are recommended with the JPEG standard. Subjective experiments involving the human visual system have resulted in the JPEG standard quantization matrix.

The software system for experimenting with quantization is developed in C#. This object-oriented system is easy for maintenance and modifications and has appropriate graphical user interface.

2 Digital images

2.1 Digitized image

Digital images are becoming more popular and there is a growing number of situations where they are used. Having images in digital form allows processing by many mathematical operations that significantly increases their overall quality. An important feature in a digital image is the projection of the 3D world to a 2D rectangle. The rectangle is horizontally and vertically discretized, into small size rectangles, usually squares, called pixels. Pixel is the smallest controllable element of a picture represented on the space. Pixels should be small enough so that human eye would not identify them as separated elements. The number of horizontal and vertical pixels determines the resolution of image.

The number of distinct colors that can be represented by a pixel depends on the number of bits per pixel. For black and white images one bit is enough, 0-white and 1-black. Each additional bit doubles the number of colors available, so two bits per pixel can have 4 colors, and a three bits per pixel can have 8 colors, etc. Most often used is RGB model where each color is represented as a combination of three basic colors: red, green and blue. The total number of bits used for an RGB color is typically called the color depth.

2.2 Compression

In computer science, data compression or source coding involves encoding information using fewer bits than the original representation. All data compression algorithms can be classified into two categories: lossless compression and lossy compression. Lossless compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression. Lossy compression is a data encoding method that compresses data by losing some of it. Lossy compression sacrifice some data so the decompressed data is not exactly equal to the original. The differences usually are not visible to the human eye.

Lossless compression is used in cases where it is important that the original and the decompressed data be identical or where deviations from the original data could be deleterious. Some image formats, like PNG or GIF, use only lossless compression. On the other hand, the image compression algorithms with losses are used in cases when some degree of information loss can be tolerated.

3 JPEG algorithm

Very often the storage and the use of digital images requires a big amount of memory that can make its use difficult. It is necessary to apply the data compression algorithms to reduce the space an image needs, requiring additional computational time for both compression and decompression operations. JPEG compression is lossy compression, but reduce significantly the space occupied by a digital image, compared with a loss of information, which is usually perceived by the human eye.

In image processing, digital images are stored in a matrix of pixels and make the structure suitable for compression, since neighboring pixels are often identical or differ by a small value. JPEG compression

is divided into several steps, and bases itself mainly on the limits of human visual perception system. The human eye is more sensitive to small change in brightness than to small changes in color. Photos in full color achieve the best compression for the human eye. The compression of images in the shades of gray is lower, but still good. The lowest compression is archived for the text and simple drawings, where visible defects appear. There are various level of JPEG compression. High compression produces lower quality and smaller files. Lower compression equals higher quality and higher file sizes. User can choose the level of compression.

JPEG compression can reduce the file size by factor of twenty without apparent loss in image quality to the human eye. It is based on DCT (discrete cosine transform) and inverse DTC which do not induce any losses of data, besides the fact that integer DTC does rounding of real value to the nearest integer.

JPEG compression is divided into several steps:

1. Transform the image by sampling chrome from RGB to YCbCr color model;
2. Down-sampling – reducing the spatial resolution of the Cb and Cr components;
3. Division of the raster image into 8x8 pixels blocks. Each 8x8 block of each component (Y, Cb, Cr) is converted to a frequency domain representation using DCT;
4. Quantization of frequency components;
5. Further compression of each block using a variant of Huffman encoding.

3.1 Color space transformation

The first step in JPEG compression of an image is color space transformation. There are different color models for image representation. Most often used is RGB model where each color is represented as the combination of red, green and blue color. YCbCr is color system developed for television. Each pixel is represented with one intensity (Y) and two hue (chroma) components (Cb and Cr). The value of this components is in the range 0 to 255.

First, the image should be converted from RGB system to YCbCr color system. The main reason for this separation is that it allows to make a reduction in information on chromatic components, possibly leaving intact those relation to the brightness. This is because the human eye is more sensitive to the variations of luminance component. YCbCr is derived from RGB system as follows:

$$\begin{aligned} Y &= 0.299*R + 0.587*G + 0.114*B \\ Cb &= -0.16874*R - 0.33126*G + 0.5*B + 128 \\ Cr &= 0.5*R - 0.41869*G - 0.08131*B + 128 \end{aligned} \quad (1)$$

3.2 Down-sampling

The transformation into YCbCr color model enables the next , which is to reduce the spatial resolution of the Cb and Cr components. Jpeg reduce chromatic components Cb and Cr by the ratio 2:1 horizontally and 2:1 vertically. The square blocks of four pixels are averaged in Cb and Cr matrices and less bits are used for Cb and Cr components. This already reduces the size of image file by 50% - a lossy

transformation. The down-sampling is one of the reasons why color images show higher compressions factor with respect to the same grey level image.

3.3 Discrete cosine transform

Jean Baptiste Joseph Fourier was a French mathematician and physicist. The Fourier transform, named after Joseph Fourier, is a mathematical transformation employed to transform signals between time domain and frequency domain. It is reversible, being able to transform from either domain to the other. Fourier discovered that any periodic function can be represented as a combination of cosine and sine functions with corresponding amplitude coefficients. This can be applied to non-periodic function in some interval, because interval can be repeated.

Discrete cosine transform is similar to the Fourier transform but more appropriate for this problem. Before computing the DCT of the 8x8 blocks, its value are shifted from a positive range to one centered zero. For an 8-bit image, each value falls in range [0,255], so that the modified range is [-128,127]. After shifting, into 8x8 blocks two-dimension DCT is applied to each block. Two-dimension DCT is given by:

$$D(u, v) = \frac{1}{4} \alpha(u) * \alpha(v) * \sum_{x=0}^7 \sum_{y=0}^7 d(x, y) * \cos\left[\frac{(2x+1)u\pi}{16}\right] * \cos\left[\frac{(2y+1)v\pi}{16}\right] \quad (2)$$

where:

- u is the horizontal spatial frequency, for integers $0 \leq u < 8$;
- v is the vertical spatial frequency, for integers $0 \leq v < 8$;
- $\alpha(u) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } u = 0 \\ 1, & \text{otherwise} \end{cases}$;
- $d(x,y)$ is the pixel value at coordinate (x,y);
- $D(u,v)$ is the DCT coefficient at coordinate (u,v);

The first coefficient (0,0) on the outputs of direct DCT corresponds to the average intensity values in blocks 8x8. That coefficient is the most important coefficient and contains most of the image information. Others coefficients which are located near (0,0) coefficient, correspond to lower frequencies, and the coefficients which are located near (7,7) coefficient correspond to the high frequencies. Most of the higher frequency coefficients are close to zero.

The matrix form of direct DCT is given by following equation:

$$T(i,j) = \begin{cases} \frac{1}{\sqrt{N}}, & \text{if } i = 0 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2j+1)i\pi}{2N}\right], & \text{if } i > 0 \end{cases} \quad (3)$$

For N=8, it results matrix T:

T =	.3536	.3536	.3536	.3536	.3536	.3536	.3536	.3536
	.4904	.4157	.2778	.0975	-.0975	-.2778	-.4157	-.4904
	.4619	.1913	-.1913	-.4619	-.4619	-.1913	.1913	.4619
	.4157	-.0975	-.4904	-.2778	.2778	.4904	.0975	-.4157
	.3536	-.3536	-.3536	.3536	.3536	-.3536	-.3536	.3536
	.2778	-.4904	.0975	.4175	-.4175	-.0975	.4904	-.2778
	.1913	-.4619	.4619	-.1913	-.1913	.4619	-.4619	.1913
	.0975	-.2778	.4157	-.4904	.4904	-.4157	.2778	-.0975

T is an orthogonal matrix. When doing the invers DCT the orthogonality of T is important, as the inverse of T is T' which is easy to calculate.

We are now ready to perform the direct DCT on matrix M (which values falls in range [-128,127]), which is accomplished by matrix multiplication:

$$D = TMT' \quad (4)$$

DCT does not make any loss of image information, because inverse DCT applied to matrix D should result in matrix M. Some errors would appear as a result of the rounding of a real value to the nearest integer.

3.4 Quantization

Quantization is the main part of compression. The human eye is good at seeing small differences in brightness over a relatively large area, but not so good at distinguishing the exact strength of a high frequency brightness variation. This allows reduction of the amount of information in the high frequency coefficients. This is done by dividing each coefficient in the frequency domain by a constant, and then rounding to the nearest integer. As a result of this, it is typically the case that many of higher frequency components are rounded to zero, and many of the rest become small numbers. After quantization, we need many fewer bits to represent 8x8 blocks.

Subjective experiments involving the human visual system have resulted in the JPEG standard quantization matrix. With a quality level of 50, the standard quantization matrix is as follows:

Q ₅₀ =	16	11	10	16	24	40	51	61
	12	12	14	19	26	58	60	55
	14	13	16	24	40	57	69	56
	14	17	22	29	51	87	80	62
	18	22	37	56	68	109	103	77
	24	35	55	64	81	104	113	92
	49	64	78	87	103	121	120	101
	72	92	95	98	112	100	130	99

Quantization is achieved by dividing each element in the transformed image matrix D by the corresponding value in the quantization matrix, and rounding to the nearest integer value. The quantized DCT coefficients are computed with:

$$B(i,j) = \text{round} \left(\frac{D(i,j)}{Q(i,j)} \right) \text{ for } i = 0,1, \dots, 7; j = 0,1, \dots, 7 \quad (5)$$

3.5 Coding

After quantization, it is quite common for most of the coefficients to equal zero. JPEG takes advantage of this by encoding quantized coefficient in the zig-zag sequence shown in Figure1.

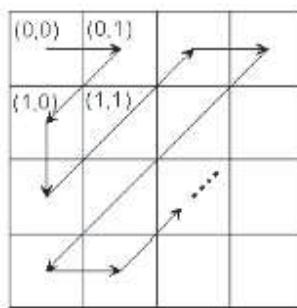


Figure 1. Zigzag order

Entropy coding is a special form of lossless compression, and the last step of the JPEG compressions. This coding of quantized coefficients is based on their statistics values. The JPEG standard allows two ways for coding- Huffman's coding and arithmetical coding. Huffman's coding uses a specific method for choosing the representation for each symbol, resulting in a prefix code. The basic idea is that the most common appeared number have shorter code, and those that appear less often have longer code.

3.6 Decoding

Decoding consists of doing all the above in reverse. The first step is converting Huffman codes in the sequence. After that, elements from sequence are extended to one 64 element long array of quantized DCT coefficients for each pixel 8x8 blocks. Each element of quantized DCT matrix is then multiplied by the corresponding element of the quantization matrix originally used.

$$R = Q \times B \quad (6)$$

The inverse DCT is next applied to matrix R, which is rounded to the nearest integer. Finally, 128 is added to each element of that result. Then, we have the decompressed JPEG version DI of our original image d.

$$DI = \text{round}(T'RT) + 128 \quad (7)$$

4 Measure of similarity

When comparing two images, it is a rather easy task for a human to render a qualitative judgment of images being close or far apart. But, the efficient computation of the similarity between two images is more difficult.

A digital image is a discrete function defined on a bounded regular grid of points in the plane and taking values in a set of color values. A digital image A can be regarded as a set $\{A_{ij}\}$ of pixels, where each pixel $A_{ij} = (i, j, R_{ij}, G_{ij}, B_{ij})$ is defined by its spatial coordinates (i, j) , red levels R_{ij} , green levels G_{ij} , and blue levels B_{ij} . The most widely used distance-based measure for image comparison is the metric based on the Euclidean distance. The Euclidean distance between $x, y \in R^n$ is computed by:

$$d(x, y) = \|x - y\|_2 = \sqrt{\sum_{i=1}^n (xi - yi)^2} \quad (8)$$

We will use an adapted measure of similarity between two images A and B, where image resolution is MxN:

$$d(A, B) = \sqrt{\sum_{i=1}^M \sum_{j=1}^N (A.R(i, j) - B.R(i, j))^2 + \sum_{i=1}^M \sum_{j=1}^N (A.G(i, j) - B.G(i, j))^2 + \sum_{i=1}^M \sum_{j=1}^N (A.B(i, j) - B.B(i, j))^2}$$

We will experiment with different quantization matrices and calculate the similarity between the original and compressed images. Quantization is done by dividing each coefficient in the frequency domain by a corresponding value in the quantization matrix, and then rounding to the nearest integer. As a result of this, it is typically the case that many of higher frequency coefficients are rounded to zero, and many of the rest become small numbers. After quantization, we need many fewer bits to represent block of 8x8.

Experimentation with the quantization matrices will be based on the number of bits to encode a block 8x8 of matrix after quantization. How many bits we need to use to encode numbers is dependent on the interval to which numbers belong. The number of bits required to represent an integer n is $\lfloor \log_2 n \rfloor + 1$.

If a number in quantization matrix is bigger than 128, after dividing corresponding coefficient matrix with that number, the result after quantization can be 0 or 1, i.e. we need only one bit to encode that coefficient; if a number in quantization matrix is between 64 and 127, the result after quantization can be 0, 1, 2 or 3, i.e. we need only two bits to encode that coefficient, etc.

We will experiment with different quantization matrices, so that the quantized blocks 8x8 of matrix may encode with a certain number of bits. For example, if we want to encode quantized block 8x8 of matrix with 6 bits, quantization matrix should have 58 coefficients with value 255, and 6 coefficients with values greater than 128.

The number of bits for encoding block 8x8 after quantization	The level of compression	Distance between original and compressed image
6	100	10138.184874556
6	80	13319.3397688628
6	60	16846.1038310509
6	50	18586.3646124033
6	40	19820.0290926356
6	20	28575.0661428358
6	1	43429.8857231921
12	100	9872.88633334928
12	80	12606.6860817518
12	60	15189.1206315331
12	50	16001.2531690273
12	40	16873.8914997951
12	20	20495.1973236192
12	1	29362.3782630475



Figure 2. Original image



Figure 3. Quality 100% (6 bits)



Figure 4. Quality 80% (6 bits)



Figure 5. Quality 60% (6 bits)



Figure 6. Quality 50% (6 bits)



Figure 7. Quality 40% (6 bits)



Figure 8. Quality 20% (6 bits)



Figure 9. Quality 1% (6 bits)



Figure 10. Quality 100% (12 bits)



Figure 11. Quality 80% (12 bits)



Figure 12. Quality 60% (12 bits)



Figure 13. Quality 50% (12 bits)



Figure 14. Quality 40% (12 bits)



Figure 15. Quality 20% (12 bits)



Figure 16. Quality 1% (12 bits)

5 Conclusion

System software is developed that facilitate access to all internal parameters of the JPEG algorithm. Since quantization process is the most important in JPEG algorithm, different ways of handling quantization are presented. If we use more bits to encode the quantized DCT coefficients, then the similarity between the images is bigger.

References

- [1] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Third Edition, Prentice Hall.

Milos Jordanski
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail:
jordanski90@hotmail.com

Nebojsa Peric
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail:
nebojscha.perich@gmail.com

Radmila Sekulic
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail:
radmilasekulic90@gmail.com

Prof. Milan Tuba
University Megatrend Belgrade
Faculty of Computer Science
Bulevar umetnosti 29N.Belgrade
SERBIA
E-mail: tuba@ieee.org

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Computing Functions on Android

**Gerhard Konnerth
Teacher Coordinator: Delilah Florea, Adrian Florea**

Abstract

“Algebra Helper” is an application developed for the operating system Android, used by mobile devices. It is applicable in the scientific area through the following mathematical functions: Calculation of a complex mathematical expression containing no variables, simplifying mathematical equations containing one or more variables up to a certain stage and definition of personalized mathematical functions.

1 Introduction

1.1 Brief description

In this paper we describe the functionality and advantages, and present software details about our original work, the Android-based application entitled „Algebra Helper”. For developing the application we required mathematical concepts of high school level but also knowledge of object oriented programming using the Android platform. Nevertheless, among others, certain sources of information must be mentioned. They served not as a source of inspiration for the composition of the paper, but as learning material for certain aspects related to the programming part. The work on several code-based problems, such as managing text files on the operating system Android, development of the layout, etc. was supported by information obtained from the following websites [1], [2] and [3]. Also, as start point for working on Android it is used [4] where authors developed a memory game for student’s entertainment.

In the following pages, the application shall be presented, beginning with a brief introduction about the intended purpose of the software. Furthermore, section 2 presented in detail the functionality of the application, including information about the interface and the features the application offers. In section 3 we described background details, including general information about implementation and code. In section 4 we shortly review the related work in the field of educational platforms. Finally, information about future development and possible areas of application shall be provided.

1.2 Development History

The first impulse to the development of the application „Algebra Helper” came through a theoretical description (by 12th grade student Eduard Marius Pop) of a potential application with appliance in the mathematical area. In detail, the concept is composed of two parts. Firstly, the potential application was to convert the photograph of a hand written mathematical expression into text format for further processing. Secondly, the obtained text would be transformed

repeatedly until a certain target is reached (ex. calculating an expression or simplifying an equation), thus solving the respective mathematical problem. I, the author and developer of the application „*Algebra Helper*”, decided to take over the implementation of the second part,

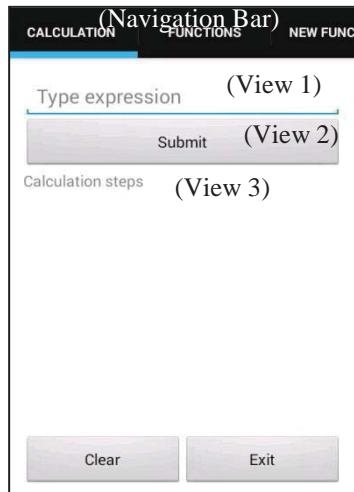


Fig. 1 – Main interface

processing of a mathematical expression in virtual text format.

2 Implementation details

2.1 Application Interface

The application should be used through the touchscreen of the Android operating system. Fig. 1 shows the main application interface, which from now on shall be referred to as a *Fragment*. A Fragment includes the so-called Views, as referred in Fig. 1 as follows:

View 1 An Edittext. Here text can be introduced using the Soft Keyboard, which appears on the screen once the Edittext is focused per touch.

View 2 A Button. It can be touched (“pressed”) to trigger a certain activity.

View 3 A TextView. Here text is displayed through the application background. Most TextViews in the application are not editable by the user.

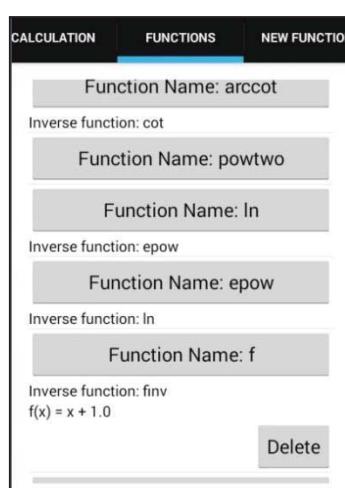


Fig. 2 – Functions part

The application consists of three main menus, each one including certain views. These shall be referred to as follows:

Fragment 1 *The Calculation-Fragment, which is presented in Fig. 1*

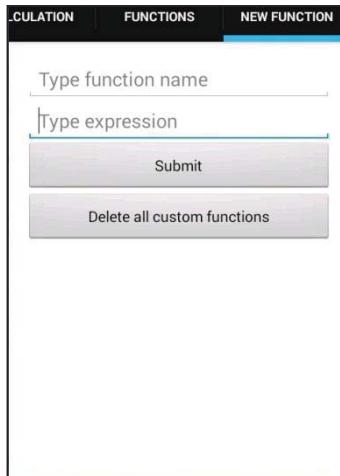


Fig. 3 – New Functions

Fragment 2 *The Functions-Fragment (Fig. 2)*

Fragment 3 *The New-Function-Fragment (Fig. 3)*

The user can navigate through the fragments either by touching the respective tab in the navigation bar (marked in Fig. 1) of each fragment, or by wiping horizontally across the screen in the opposite direction of the targeted fragment (ex. the user can switch from the Calculation-Fragment to the Functions-Fragment by wiping horizontally from right to left).

This terminology (Fragment, View, Edittext, TextView, Button, Calculation-Fragment, Functions-Fragment, New-Functions-Fragment) shall be used in the following sections and subsections in order to describe the application features.

2.2 Application Features

The functionality of the application can be resumed to three main features: Calculation, Equations and definition of personalized Functions. These shall be presented in detail in the following.

2.2.1 Calculation

This feature can be accessed in the Calculation-Fragment (Fig. 1). It consists of transforming a mathematical expression into an equivalent number. Therefore the expression must not contain variables, but only numeric values, as in the following example:

$$3+4*\sin(3.14) \quad (1)$$

After introducing the expression (1), the output would be 3.0064, the number obtained from the calculation of the expression, alongside the particular steps for the calculation. Fig. 4 shows the

particular case for example (1). The Expression (1) is introduced in the Edittext on the top of the Calculation-Fragment. After the button “Submit” is pressed, the calculation steps are displayed in the Textview below the button, where “E” is the name given to the expression. The last entry in this Textview (“E = 3.0064”) represents the result of the calculation.

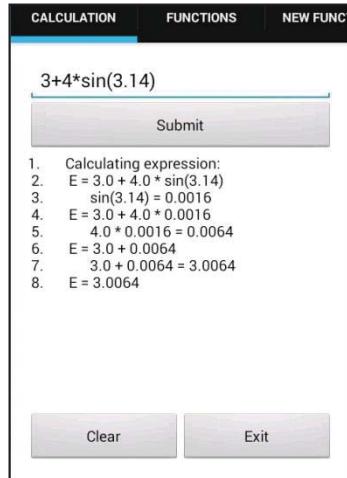


Fig. 4 – Calculating expression

Calculation is operated with an accuracy of 4 decimals. The steps remain displayed in the Textview until they are deleted (see sub-subsection 2.2.3 “Other Features in the Calculation-Fragment”), even if another expression is inputted. The exact way of processing the data for this certain example shall be depicted in the section “Sections of Code”.

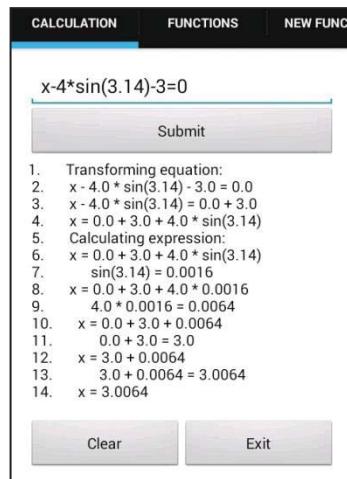


Fig. 5 – Transforming equation

2.2.2 Equation Solving

As the calculation feature, the solving of an equation takes place in the Calculation-Fragment. It consists of expressing a variable from a simple mathematical equation. The following example shall be used for presentation purpose:

$$x - 4 * \sin(3.14) - 3 = 0 \quad (2)$$

The target of this feature is to obtain a result similar to the following equation (3). In this example “ a ” is a rational number and “ x ” is a variable.

$$x = a \quad (3)$$

Therefore, the equation will be transformed step by step, with the aim of transferring all numbers to the right side, and keeping all variables on the left side. As depicted in Fig. 5, the equation (2) will be brought to the following form:

$$x = 0 + 3 + 4 * \sin(3.14) \quad (4)$$

At this point (where all the transformations the application can perform are done and the right side of the equation contains no variables), the Calculation-Feature takes place as described in the subsection “2.2.1 Calculation”. The right part of the equation (4) is equivalent to the expression (1) from the subsection “2.2.1 Calculation”, consequently the output will be similar and the result will be exactly the same.

2.2.3 Other Features in the Calculation-Fragment

As clearly visible in Fig. 1, Fig. 4 and Fig. 5, there are two buttons on the bottom of the Calculation-Fragment: the Clear-button on the left side and the Exit-button on the right side. The former is used in order to delete the calculation steps displayed in the Textview and the latter to close the application.

2.2.4 Personalizing Mathematical Functions.

In the New-Function-Fragment, the user has the possibility to define a personalized (in the application referred to as “custom”) function. Therefore, he/she has to input a name for the function and an expression, as in the example below:

$$f(x) = x + 1 \quad (5)$$

In this particular case, the function name would be “ f ”, and the expression “ $x+1$ ”. This information must be introduced in the New-Function-Fragment (Fig. 3) as follows: The function name in the first Edittext with the input hint: “Type function name”, and the expression in the second Edittext with the input hint: “Type expression”. After the button “Submit” is triggered, the function f is saved (see details in sub-subsection “2.2.5 Function List”) and it can be used in contexts as the features presented in sub-subsections 2.2.1 and 2.2.2. Simultaneously, the application tries to create an inverse function for the saved function. If it succeeds, it will save the inverse function as well creating its name as follows (ex. see sub-subsection “2.2.5 Function List”):

$$\text{Inverse name} = \text{function name} + \text{"inv"} \quad (6)$$

Each custom function can be deleted individually (details in sub-subsection “2.2.5 Function List”), or together with all other custom functions all at once. This is done by triggering the button “Delete all custom functions” in the New-Function-Fragment.

2.2.5 Function List

All the mathematical functions available for the features 2.2.1 and 2.2.2 and for the expressions inputted in feature 2.2.4 are listed in the Functions-Fragment (Fig. 2). There, each function is defined by an entry consisting of a name, the name of its inverse function (if the application can create one), and the respective expression, if the function is a custom one. In Fig. 2 for instance, the custom function “f” is listed along with the automatically created inverse function “finv” and the expression “x+1”, which was inputted by the user earlier (see sub-subsection 2.2.4). The function “f” can be deleted by triggering the button “Delete”. This will not affect other custom functions, not even “finv”, with the exception of the case, where “f” is a part of other functions, as in the example below:

$$g(x)=f(x)+1 \quad (7)$$

In this case, function “f” would not be recognized as a function anymore. Therefore, the function “g” would not work. On this particular aspect, further work might be done, such as an alert message in order to warn the user about functions which would stop working after another function is deleted.

Besides the custom functions, a set of standard functions are provided by the application (inter alia several trigonometric functions, the square root, the natural logarithm, etc.). These cannot be deleted.

In the Functions-Fragment, the names of each function are displayed via a button, as visible in Fig. 2. When this button is triggered, the details (inverse function, expression) are hidden (as in Fig. 2 the function “powtwo”). Another click on the button allows them to be displayed again.

3 Software Development and Implementation

3.1 Programming Environment

The application was developed in the IDE eclipse, using the object-oriented programming language Java. For compatibility reasons, the Android SDK was used in order to deploy the application for the operating system Android. Thus, a major part of the code is based on certain android libraries, which are not included in pure Java.

3.2 Sections of Code

In the following, two major components of the application are explained through sections of code. These exact sections play no part in the application itself, but represent simplified versions of certain parts of the application background, for explanation purposes.

3.2.1 Decoding of data input

Every mathematical expression inputted into the application is received in text format, thus, forming a String, a Java class, which can be used to store a text. This String has to be evaluated in order to be worked on later (ex. see sub-subsection 3.2.1 “Performing Calculations”). Therefore, every mathematical operator (+,-,*,/,^) must be recognized as such. In simplified version, the following code represents such an operation:

```
void decode(String input) {
    for (int i = 0; i < input.length(); i++) {
        char ch = input.charAt(i);
        if (ch == '+' || ch == '-')
            addAddition(ch);
        else if (ch == '*' || ch == '/')
            addMultiplication(ch);
        else if (ch == '^')
            addPower(ch);
    }
}
```

The methods “addAddition()”, “addMultiplication()” and “addPower()” process the data according to the particular case. Obviously, the actual algorithms used in the application are much more complex, because, besides mathematical operators, functions, variables, brackets and numeric values have to be processed appropriately. Nevertheless, the whole decoding process is based on this simple character-by-character analysis.

3.2.2 Performing Calculations

The basic structure the application follows in the calculation process is the analysis of each particular function and operator. Therefore, these have to be processed as certain data types. For presentation purposes, the following example only illustrates the processing of operators in a decoded input text (see sub-subsection “3.2.1 Decoding of data input”).

```
void calculate(Operator[] operators, int maxNrBrackets) {
    int nrBrackets = 0;
    while(nrBrackets <= maxNrBrackets) {
        for (int i = 0; i < operators.length(); i++) {
            if (operators[i].getNrBrackets == nrBrackets)
                operators[i].trigger(getLeftNumber(i); getRightNumber(i));
        }
        nrBrackets++;
    }
}
```

```

Number getLeftNumber(int i) {
    /* Returns the number on the left side of the operator at the index "i".
Ex: In the expression "1+2", for the operator "+" it would return the number 1"
    */
}

Number getRightNumber(int i) {
    /* Returns the number on the right side of the operator at the index "i".
Ex: In the expression "1+2", for the operator "+" it would return the number 2".
    */
}

class Operator {
    int getNrBrackets() {
        /* Returns the number of brackets the operator is surrounded by.
Ex: In "2*(1+3)", the operator "*" is surrounded by one set of
brackets. Thus, the method would return 1.
        */
    }
}

```

In this algorithm, the operators are processed according to the number of brackets they are surrounded by. Thus, the mathematical accuracy is assured. This example is only a draft of the actual application background. The actual structures are much more complex.

4 Related Work

During last decade many international e-Learning platforms were developed that targeting on large scale education for every person not necessarily students. Such platforms are IBM Education Assistant [5], Cisco Academies [6], Khan Academy [7] and Oracle [8]. Besides learning modules, these platforms encompass evaluation components based on quizzes, partial and final exams, and practice exam developed in Flash. Unfortunately, not all of these courses are free, the cost of these courses ranges from hundreds to thousands of dollars. Therefore, “*Algebra Helper*” – a free software tool – is dedicated for students and teachers from academia especially belonging to “Samuel von Brukenthal” National College and “Lucian Blaga” University, both from Sibiu, eliminating the cost impediment and providing learning opportunities for everyone.

Another reason that makes our work valuable is the low number of educational applications developed on the Android platform. It represents a step forward for the development of Java applications for any student passionate about games and mobile programming.

5 Conclusions and further work

5.1 Compendium

In conclusion, the application „*Algebra Helper*” is a promising component in the area of scientific computing. The user-friendly interface makes it easily accessible by the non-professional user, but nevertheless some of the features may be valuable for specialists in a certain scientific domain. Therefore, the application is applicable both for simple, mundane mathematical problems and for professional scientific use.

5.2 Further Work

As an individual piece of software, the features of the application might be limited. Nevertheless, the area of application would be wide if the application was supplemented by other features. The following paragraphs describe possible options for the further development, which are not accessible in the application at the time being.

5.2.1 Optical Character Recognition

As mentioned in the subsection “1.2 Development History”, the application name is meant to be combined with an OCR-feature (Optical Character Recognition). It would have the possibility to work both on paper and on the mobile device if it will be combined with this feature. A photograph of a hand written mathematical equation or expression could be processed immediately without the necessity to re-write it on the Soft Keyboard. This feature could be used for every function described in the subsection “2.2 Application Features” (Calculation, equation solving and definition of personalized/“custom” mathematical functions). The following example should depict the functionality if the ORC-feature was implemented.

$$myfunction(x)=3*x+4 \quad (8)$$

Should the user write the example (8) by hand, the application could be able to process a photograph of this particular line. Using the New-Function-feature (as described in the sub-subsection “2.2.4 Personalizing Mathematical Functions”) for example, the application could recognize the word “*myfunction*” as a function name, and “ $3*x+4$ ” as the respective expression. In conclusion, the output/result would be the same as if the input was written through the Soft Keyboard, for every feature described in sub-subsections 2.2.1, 2.2.2 and 2.2.4 (Calculation, equation solving and personalized functions). With the OCR-feature, the application would be of great utility and speed, especially for the amateur user.

5.2.2 Database Input

Another possible area of application for the application „*Algebra Helper*” could be the processing of large amount of data. In a particular case, where the same operation has to be executed with multitudinous sets of input, the application could be of great utility. At the time being, the application is not able to process functions with more than one variable. Nevertheless, the implementation of this feature is possible and could be advantageous, as the following example depicts.

$$average(x, y, z) \quad (9)$$

In the example (9), “*average*” is a function which calculates the arithmetic mean of the three arguments. Applied on a database, the application could repeatedly perform the calculation of the function (9) for every entry in a given table, thus, completing the database with the respective result for each set of input. This feature could be very useful for the professional user. He/she would be able to perform statistical calculations, analysis of physical experiments and other similar operations on a mobile device.

References

- [1] <http://stackoverflow.com/>
- [2] <http://www.vogella.com/tutorials/AndroidListView/article.html>
- [3] <http://android-developers.blogspot.ro/2012/05/using-dialogfragments.html>
- [4] Florea Delilah, Florea Adrian, *Matching Game*, Proceedings of the International Symposium "Creativity and technology in addressing the educational process", Hunedoara, România, June, 2013, ISBN 978-973-0-11754-7.
- [5] IBM Education Assistant, <http://publib.boulder.ibm.com/infocenter/ieduasst/stgv1r0/index.jsp> (accessed 05 April 2014)
- [6] Cisco Academies, <http://www.cisco.com/web/learning/netacad/index.html>, (accessed 05 April 2014)
- [7] Khan Academy, <https://www.khanacademy.org/>, (accessed 18 April 2014).
- [8] Oracle Academy, <https://academy.oracle.com/>, (accessed 18 April 2014).

Gerhard Konnerth
„Samuel von Bruckenthal“ National College
Piața Huet, 550182 Sibiu, România ROMANIA
E-mail: gerhard.konnerth@yahoo.de

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Theoretical Bases of Building a Decision Support System (Dss) for Developing Regional Innovation Program

**Narine Manasyan
Teacher Coordinator: Vladimir Chernov**

Abstract

In the transforming economic environment the prevalence of the class of ill – structured problems, among them also the development of an innovation program, demands introduction of DSS in the decision making process which shall maximally meet the problem being solved, as the modern computer technologies perform not only computing operations, but also structuring etc.

1. Introduction

The current processes of economic development show that the rapid pace of development requires application of operative and adequate systems of economic management in the process of making decisions on the class of complex, multi – criteria problems, when it is necessary to take into consideration a wide range of influencing factors. As a basis of such systems the information technologies (particularly cognitive ones) are which are developing rapidly during recent years.

2. Problem description

The innovative development of a region, as any other qualitative change, takes place by leaps and bounds which conditions the ill – structured, sometimes non-structured character of this process, which is accompanied by high level of uncertainty and is characterized by the uniqueness of the decision maker, by dominance of qualitative parameters and their verbal evaluations. Though the solution of ill – structured problems, which include considerable amount of heterogeneous, fuzzy information and a great number of interrelated factors and criteria, is prepared by a group of experienced experts, it shall be accompanied by DSS to increase its quality.

3. Decision support systems (DSS)

Under DSS an information system is implied, which takes part in the decision making process for formalization of ill – structured and non – structured problems. In more complex situations, when the problems incompletely defy modeling, are of multi – criteria character and do not let define fully the criteria for decision quality evaluation until a certain point of time, the

main predetermination of instrumental tools of decision making support situation analysis and assessment, generation of possible scenarios of development, creation of criteria and evaluation of criterion correspondence, their agreement, alternatives' arrangement, finding out satisfactory solutions, consequence analysis etc. are.

When realizing computer support it is necessary to choose the main component of DSS correctly, i.e. to define the method of solutions. The DSS application possibilities on the basis of fuzzy mathematics (fuzzy logic) are connected with the fact that in uncertainty conditions the fuzzy systems are able to reflect the human thinking and to find out a solution that is close to his perception, i.e. the fuzzy – set models are based on the semantics of object domain and brings computer logic near to the human thinking [3].

Consequently, the models, formally reflecting the process of innovation program (IPm) development (formation), and the methods of decision making in the DSS architecture shall be built on the basis of fuzzy – set descriptions. The main task of such DSS the formalization of fuzzy information, its processing and realization of different fuzzy operations are.

Stepwise formalization and algorithmization of the problem of IP development and realization using DSS can be organized according to the following scheme (Fig. 1).

Block I – the analysis of the regional innovation politics demands conduction of regional economy monitoring, when its real economic base, innovation potential, level of innovative development and the external environment effect are assessed. Consequently, definitely circle of problems is revealed which are to be solved. Here, above all, the solvability of the problem is evaluated, then its verbal drafting takes place, as well as the goals, restrictions and main directions of the development of the region are set and the composition of instruments of regional innovation policy is determined.

The given stage can be conducted using the fuzzy logic and neural network combinations, i.e. hybrid networks which are based on the artificial neural – fuzzy networks and can be used as the main instrument of evaluation and forecasting of innovation potential of a region.

Success of the development strategy realization is based on the created IPm and its constituent projects which conditions the transition to the block II.

Block II – sphere of profile IPm generation. In this cluster the starting IPm are formed with potential portfolios of typical IPts. As the uniqueness of IPm makes the decision making by means of computer impossible, the fuzzy preferences of decision makers and experts suppose stage by stage method of problem solution with informal (heuristic) methods, creation of knowledge base of expert system and possible scenarios.

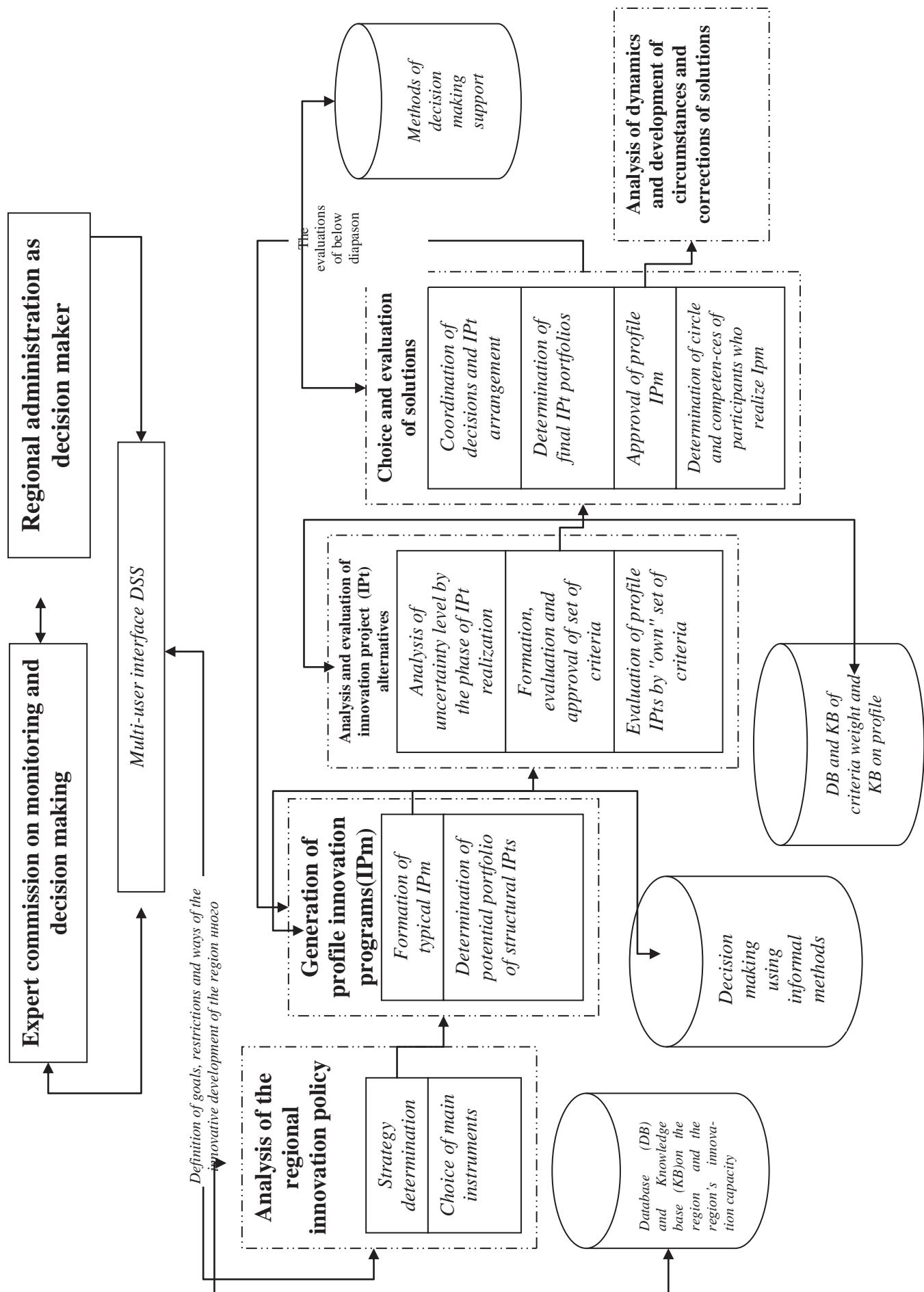
Block III – The sphere of analysis and evaluation of alternatives covers the whole process of IPt sampling and is the most responsible stage during which the alternative arrangement takes place:

1) using expert methods the uncertainty level of the project is analyzed in the realization stage,

2) on the basis of database (DB) and knowledge base (KB) of criteria hierarchy the set of criteria is made up, evaluated and approved, and their formation can be conducted on the basis of fuzzy sets and expert inquiries, in case of full uncertainty, which propose a certain set of criteria and assess their importance by the IPt form,

3) a principally important stage the assessment of profile IPts is by means of "own" set of criteria.

We would like to point out that in this block the iteration possibility is foreseen, because the projects which don't meet the organizational requirements may be corrected and transferred to the next sampling stage.



Block IV – choice and evaluation of solutions using the methods of decision making support generalizes the expert opinions and carries out the projects' arrangement. Taking into account the arrangement the decision maker determines the final IPt portfolio, approves the typical IPm, and defines the participants' circle and competences, who realize the program. The stage of program approval and adoption also have iteration character, as two outcomes are possible: when there are satisfactory results and the process finishes, and, on the contrary, when the projects' evaluations don't overcome the decision threshold and the process starts again, i.e. it goes to block II.

Block V – analysis of development dynamics of circumstances. In changing conditions of realization of any action plan the DSS shall give the decision maker the possibility to correct the plan at any moment and make decision taking into consideration the changes. In this case the decisions can be made after the analysis of computer game with the possibility of rule changes.

4. Conclusion

The scheme of DSS functioning is made up to solve such ill – structured problems, as the complex object domain of innovation program development, the introduction of which decreases the laboriousness and increases the efficiency of made decisions. The main approach to the inadequate architecture of DSS is given, in which the DSS subsystems blocks are realized on the basis of fuzzy sets.

References

- [1] Andreichikov A.V., Andreichikova O.N. *Kompiuternaya podderzhka izobretatelstva (metody, sistemy, primery primeneniia)*. – M.: Mashinostroenie, 1998. ISBN 5– 217 – 02886 – 6
- [2] Simankov V.S., Vladimirov S.N., Denisko A.O., Cherkasov A.N. *Metodologicheskie aspekty postroeniia sistem podderzhki priniatii reshenii* // Vestnik DGTU, 2008. T.8. №3(38) , 258-268 pp.
- [3] Sokolova A.Iu. *Razrabotka modelei mnogokriterialnogo vybora alternativ na osnove nechetkikh mnozhestv vtorogo poriadka dlia reshenii ekonomicheskikh zadach* [online at: <http://www.scienceforum.ru/2013/pdf/5917.pdf>].
- [4] Trakhtengertc E.A. *Kompiuternaya podderzhka priniatii reshenii: nauchno-prakticheskoe izdanie*. / E.A. Trakhtengertc // Ser. "Informatizaciia Rossii na poroge XXI veka". – M.: SINTEG, 1998

Narine MANASYAN
Vladimir State University
Management and Computer Science in Technical and Economic Systems
Gorky St., 87, Vladimir
RUSSIA
narinemanasyan@mail.ru

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Alarm modules, a necessity for industrial software

**Istvan Benjamin Pallo
Teacher Coordinator: Dana Simian**

Abstract

The aim of this paper is to emphasize the importance of alarm modules and to present the design and implementation of such a module embedded in GeoTrekker¹ software. GeoTrekker is a SC Dafllog SRL software solution used for drilling installations.

The alarm modules are used for monitoring complex systems and must be able to notice using visual; audio or other kind of signals the appearance of an unwanted event. The alarm modules are necessary in the case of systems dependent of many parameters which must be kept in a given interval. The simultaneous monitoring of all parameters cannot be made efficiently and rigorous by human operators. The values of parameters can be vital not only for the functioning of system itself but also for the safety of operators and of the environment in which the system is located

The task of GeoTrekker software is to trace about 145 of drill parameters in each second of time. Monitoring so many parameters with such time frequency and acting properly in each situation is impossible to be made only by human operator.

1. Introduction

Complex systems dependent of many parameters can be found in many field of activity, from industrial field to medicine and aeronautics and space administration. The apparition of different events in these systems shall be perceivable at any moment the possibility of emitting different kind of alerts must be available. Software solutions for monitoring complex system are utilised at large scale. As critical an unexpected event in this system is as rapid the answer of the system must be. If some parameters of the system are monitored the events consist in reaching a certain threshold for parameters. In order of acting in time for correcting the parameters many alarms can be automatically emitted by the software before reaching this threshold.

The aim of this paper is to present an alarm module for a monitoring software system used for drilling installations, but it can be adapted for parameters monitoring in any other software systems.

Our module is design and implemented within GeoTrekker software. GeoTrekker is a SC Dafllog SRL software solution used for drilling installations.

¹ GeoTrekker is property of SC Dafllog SRL. Code used with permission.

The paper is organized as follows. In Section 2 we present the basic notions relating to alarm's types and the object oriented design of these types of alarms. Section 3 details any alarm type. The interface of our module is presented in Section 4 and Section 5 contains conclusions and further directions of work.

2. Design of alarm module

The task we face out is to define many alarm type capable to solve all the situations which can appear on the drilling installation running. The following situations must be solved: parameter value is equal to a given value, parameter value is lower or higher than a given value, parameter value is situated inside or outside of a given interval. Also there are situations when it is necessary to know when a parameter changes its value whit a specific threshold. A specific alarm for our case appears in the situation when no signal from DAQ board (acquisition board) is received by computer.

An alarm has both visual and audible flags. Visual flag lasts as long as the monitored parameter meets the conditions to raise the alarm. After those conditions are no longer met the visual flag disappear. The audible flag lasts until the operator stops it even if the visual flag is no longer enabled. The visual and the audible flags depend also of the alarm level. There are four alarm levels: without alarm, announce warning and emergency. For first level there are no visual or audible flags. The second level of alarm comes with a slow sound and a light shade of red. For warning level a dark shade of red than that for announce level is used and also a new sound as audible flag. For emergency level it is used the red shade and most powerful audible flag.

Also there are different types of alarms. For some parameters is required to monitories if their value exceeds a threshold or maybe is needed to monitories if parameter value is inside or outside of an interval. So there where designed more alarm types. Those types are specified in Fig. 1.

```
public enum AlarmTriggerType
{
    EqualTo,
    LowerThan,
    HigherThan,
    InsideInterval,
    OutsideInterval,
    Change,
    AquisitionAlarm
}
```

Fig. 1 – Alarm types

To define all those alarm types we used object orientated programming. Through OOP we could design all those types of alarms starting from an abstract class and using polymorphism. The abstract class is called “AbstractSingleAlarm”. The derived classes are used for any type of alarm as we can see in Fig. 2 where is depicted the class diagram.

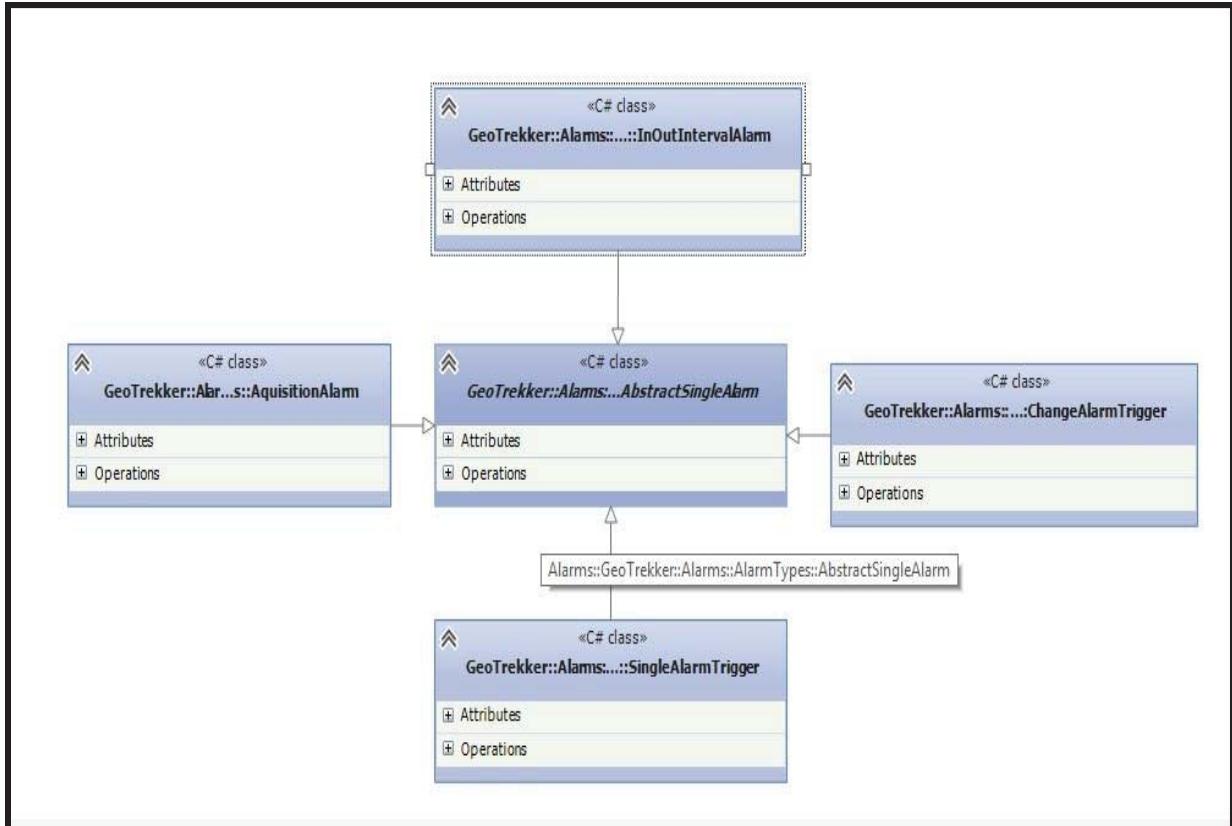


Fig. 2 – Class diagram

The abstract class “AbstractSingleAlarm” incorporates information common to all alarm types. This class contains a field for the type of the alarm. Each alarm must have a type because for an alarm without a type the operations that must be made are unknown. The class also contains a property that specifies if the value of the parameter can be calculate in PPM (parts per million). For some parameters the value is calculated in PPM. Also the class has a flag for messages. Each alarm shows a message in a history box when it starts. For each alarm a check is made in each second. If an alarm starts the start message is showed only one time. In order to see if an alarm which started sent a message to the history box we need to check the value of the flag. The class have also a field where is saved the name of the parameter monitored by the alarm. To know which visual an audible the alarm will use the class contain a field for the state of the alarm: without alarm, announce, warning or emergency. The source code for “AbstractSingleAlarm” class is presented in Annex 1.

Another important class is “AlarmCalculationServer”. This class represents the “engine” of the alarm module. The responsibility of this class is to call in each second and for each alarm that was set a method named “CheckAlarm”. This method makes the necessary verifications to determine if the conditions to start the alarm are met.

There are two general types of alarms: permanent alarms and occasional alarms. Permanent alarms are set automatic when GeoTrekker starts. The operator could not delete any alarm which is constructed as a permanent alarm. He can only to modify the details (type of alarm, level of the alarm, threshold value). To set an occasional alarm the operator must introduce alarm’s details:

parameter, alarm type, alarm level, threshold value. After an occasional alarm is set the operator can delete the alarm if it is no need to monitors alarm corresponding parameter.

3. Alarm types implementation

The alarm types given in Fig. 1 are all inherited from the abstract class “AbstractSingleAlarm”. This class contains basic details common to all types of alarms. In the following we will present each type of alarm. For the implementation of each module it was used as programming language C# [1].

Acquisition Alarm

The acquisition alarm is a permanent alarm. Also this alarm isn't related of any parameter that exists on a drill installation. This alarm was implemented to monitors the connection between the DAQ board and the PC. A DAQ board (or an acquisition board) is a board with analogical and digital channels which receives electrical impulses from different sensors and know how to convert those impulses into real values and send them forth to GeoTrekker or any other software solution for calculations. There can be only an alarm of this type on each instance of GeoTrekker (on a PC it can run only one instance of GeoTrekker a time). A signal is transmitted from DAQ board to the PC. If there is no signal that means there is no connection between the PC and DAQ board and no calculations are made. Because we have no output results it is impossible to appreciate the progress that the drill installation has made. So it is crucial to notify the operator if there is no connection. The principle behind this alarm is the following: if there are more seconds than a threshold value the alarm stats. To count the seconds a Timer object with an interval of a thousand milliseconds is used. The counter starts if the signal from DAQ board is missing (in this case the signal is a message that is send to all modules of the software). If the connection is restored the counter is reset and the counting stops. However if the connection isn't restored the counter starts over and over again until the connection is restored or the program is restarted.

Change alarm.

Change alarm is used when it is necessary to monitor parameters whose values can increase or decrease by several units. The principle is the follow: if the parameter's value changes (increase or decrease) with more than a threshold value the alarm starts. For example: the operator set a change alarm on pressure and he want to know when the pressure of a tank was changed with more than five bars. This type of alarm is used as permanent and occasional alarm.

EqualTo, HigherThan, LowerThan

EqualTo, HigherThan and LowerThan alarms have the same data that compose the objects. The difference between those types of alarms is given by the type and the comparison which is made between the threshold value of the alarm and the parameter's actual value. Because the differences are so small it was built a class named “SingleAlarmTrigger” that can manage all three types of alarms. “CheckAlarm” method is the one that decide what type of comparison to make depending on the alarm type specified inside of the “SingleAlarmTrigger” object. Also it can be set permanent alarms with those types. Most of permanent alarms set are *HigherThan* alarms.

InsideInterval and *OutsideInterval*.

Inside and outside have also the same data which compose the objects of those types. The difference between those types of alarms and the previous ones is that parameter's value is compared with one of the limits of an interval. Also permanent alarms of those types can be set.

An alarm of any type of those that were presented before shows a message when the alarm starts or stops. Those messages are useful for the moments when he comes back the message that tells him what parameter raised the alarm so that he can verify what happened. The start message tells to the operator that the parameter's value meet the condition to start the alarm. For example if an EqualTo alarm starts the start message will be "Parameter current value is equal to threshold value." The start message for each type of alarm is different. The stop message for all types of alarms is same: "The alarm was stopped." The acquisition alarm has different message for start and stop. When the acquisition alarm starts there is no connection and the message is "Data acquisition is stopped!" If the alarm stops the message is "Connection restored." That means that now the signal send from DAQ board to the PC reaches its destination. The class that manages those messages is called "AlarmMessages"

4. The interface

Taking into account the stress level of the operators working on a drilling installation, the simplicity of the interface is required them to keep an interface with as few details as possible. The interface was built using forms. A form is a window used to create the interface for the user. GeoTrekker contains twelve modules. Because there are too many to be showed in a single window each module is minimized and they are arranged in a stack in the left bottom of the window. When the operator needs to consult a module it can maximize it with a double click on the module. Also the module can be maximized by clicking the module name in the menu from the top of the window.

The window space of this module is organised in two parts. In the left part it is draw a grid with three columns. This grid contains the list with alarms that where set. By default the grid contains the list with the permanent alarms. To this list the operator can add other alarms by setting up new ones. In the right side it is draw a multiline textbox in which is written when an alarm starts or stops. This textbox is used as a history for the start or stop of different types of alarm.

In the grid it was introduced a row double click event. Through this event the operator can see the details for the alarms that where set and even to delete them. For all alarms the details can be modified but only occasional alarms can be deleted. When the operator press a double click on a grid row the alarm object which is represented by that row is transmitted as a parameter for a new alarm window and all the details of the alarm are set on the new alarm window controls (textboxes, dropdown lists, labels, group boxes). After this the operator is free to modify the details of the alarm.

In the following we show the interface of the alarm module.

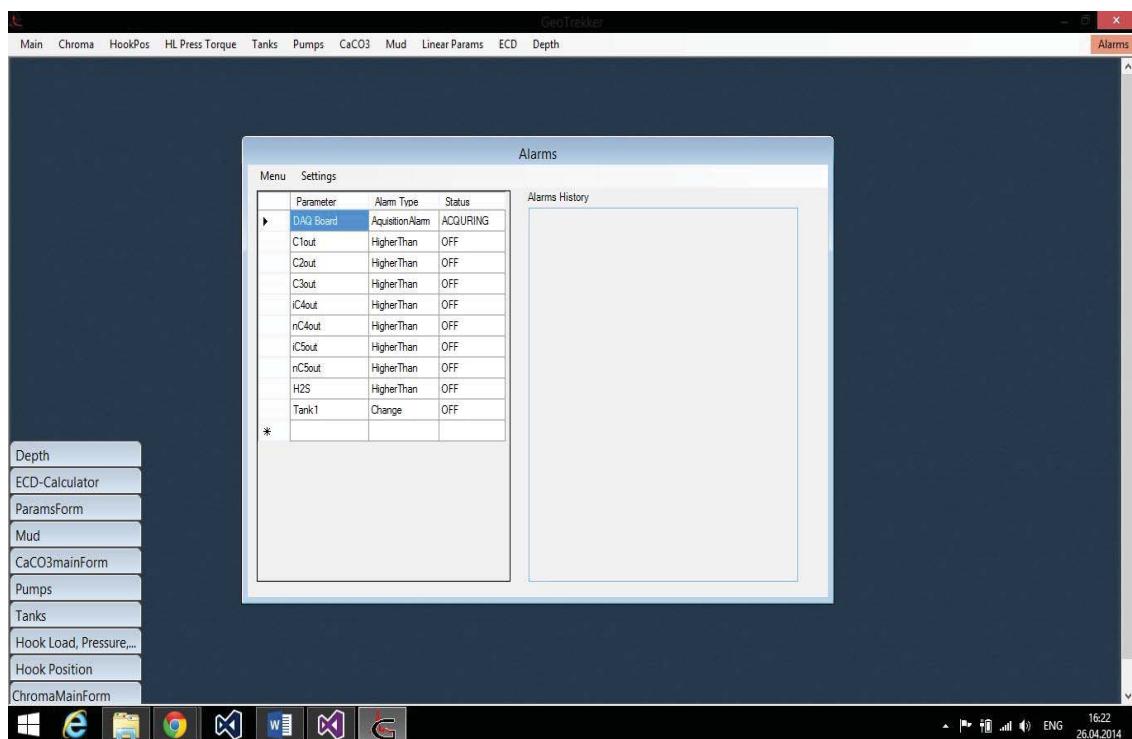


Fig. 3 – Main window for alarms

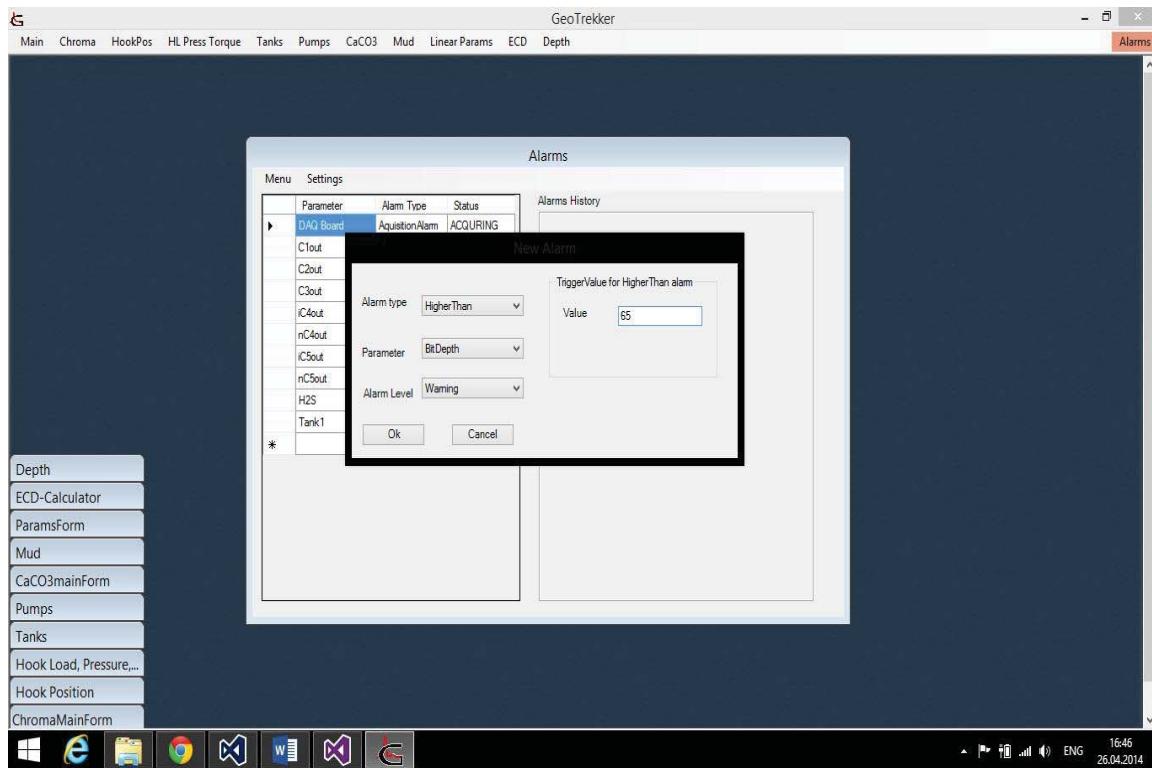


Fig. 3 – New alarm window

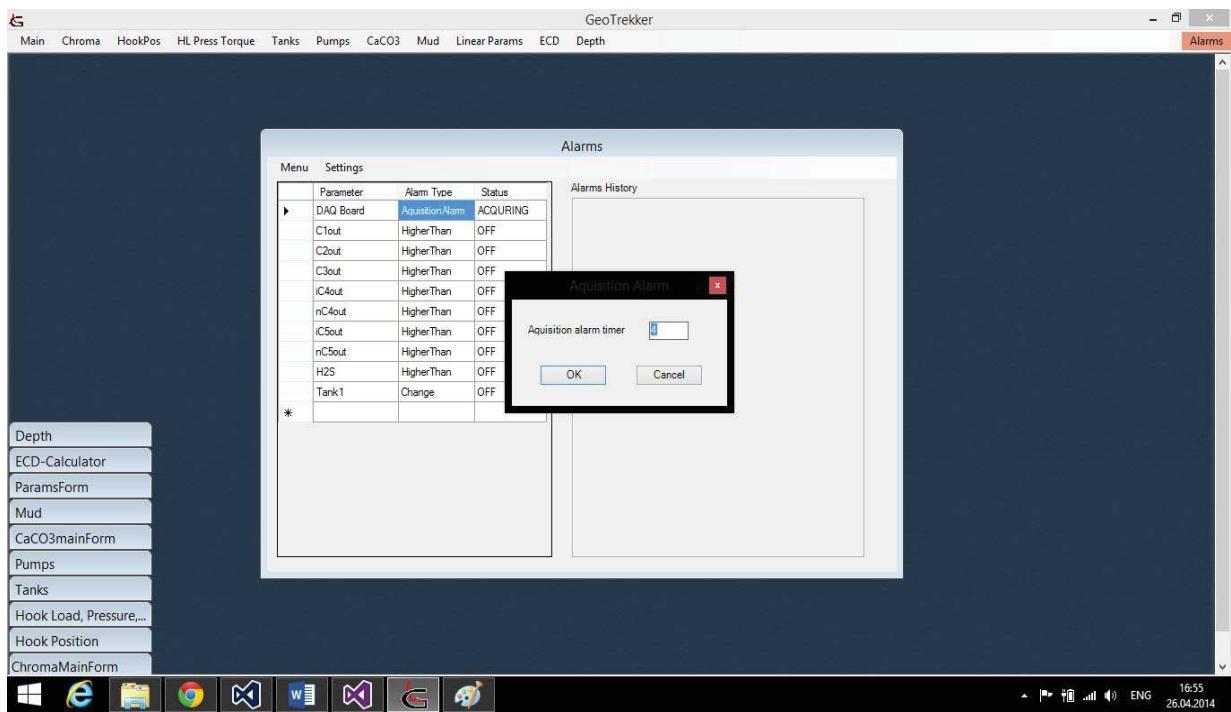


Fig. 4 – Acquisition alarm window

5. Conclusions

GeoTrekker is software used on drilling installations to monitor drilling parameters. There are also other software solutions that can do this job. But those software solutions like DAQFactory[2] or WinDAQ[3] are quite expensive. For small companies it is necessary to find solutions that are cheaper but can assure the necessary functionalities that are also offered by the others software on the market.

This is the case of the SC Defog SRL company that I work for and which has projected and developed GeoTrekker, software designed to monitor efficient de drill process.

As future developments for alarm module we will implement a class that can play the sound of the alarms outside of the cabin where the PC is through different devices like megaphones.

The alarm module is integrated into GeoTrekker. In present the company's Daflug team that I am a part of works to develop a new version of GeoTrekker: GeoTrekker2. This new version will require some modifications to the alarm module. Through those modifications there are: a new interface developed with WPF (Windows Presentation Foundation) and a new calculation class.

The existence of an alarm module is required in any software whose purpose is to monitor parameters and to attention the operator when situations that may endanger the safeties of the operations that are made on a drilling installation.

The alarm module was built on demand. The designing of this module required documentation about other software solutions that are available on the market and also a documentation about the drilling installations. The design of the module can be applied to other software solutions in this field.

Annex 1 - Source code of class “AbstractSingleAlarm”

```
public abstract class AbstractSingleAlarm
{
    protected bool HasChangedInLastSeccond;
    protected string Status;
    public string AlarmStatus
    {
        get { return Status; }
        set { Status = value; }
    }
    public bool BoolStatus
    {
        get; set;
    }

    public bool MessageRegistered;

    protected string Parameter;
    public string ParameterName
    {
        get { return Parameter; }
        set { Parameter = value; }
    }

    protected AlarmTriggerType TriggerType;
    public AlarmTriggerType AlarmType
    {
        get { return TriggerType; }
        set { TriggerType = value; }
    }

    public bool WasChange
    {
        get { return HasChangedInLastSeccond; }
        set { HasChangedInLastSeccond = value; }
    }
    public abstract bool CheckAlarm();

    protected Results ResultType;
    public Results AlarmResultType
    {
        get { return ResultType; }
        set { ResultType = value; }
    }

    protected AlarmLevel AlarmLevel;
    public AlarmLevel AlarmLevelType
    {
        get { return AlarmLevel; }
        set { AlarmLevel = value; }
    }
    public bool IsChromaParam()
    {
        Array chParams = Enum.GetNames(typeof(ChromaNew.Component.ComponentName));
        return chParams.Cast<string>().Any(param => Parameter == param);
    }
}
```

Annex 2 – Source code of class “AlarmsCalculationServer”

```
internal class AlarmsCalculationServer : CalculationServerAbstract
{
    private AlarmsSettings _alarmSettings;
    public AlarmsCalculationServer(AlarmsSettings alarmSettings)
    {
        _alarmSettings = alarmSettings;
    }
    public AlarmsSettings AlarmsSettings
    {
        get { return _alarmSettings; }
        set { _alarmSettings = value; }
    }
    protected override void Compute()
    {
        if (_alarmSettings.AlarmsList != null)
            foreach (var alarm in _alarmSettings.AlarmsList)
            {
                alarm.CheckAlarm();
            }
        if (_alarmSettings.PermanentAlarmsList == null) return;
        foreach (var alarm in _alarmSettings.PermanentAlarmsList.Where(
                    alarm => alarm.AlarmType != AlarmTriggerType.AquisitionAlarm))
        {
            alarm.CheckAlarm();
        }
    }
}
```

References

- [1] John Sharp. *Microsoft Visual C# 2010 Step by Step*. Microsoft Press. Redmond, Washington. 2010
- [2] <http://www.azeotech.com/j/products/what-is-daqfactory.html> 01.05.2014
- [3] <http://www.dataq.com/products/software/> 01.05.2014

Istvan Beniamin PALLO
Faculty of Science “Lucian Blaga”
Department of Informatics
Str. Dr. I. Ratiu, No. 5-7, Sibiu, 550012, Romania
ROMANIA
Email: eu.stefy@yahoo.com

Real-time edge detection for video capture

Nebojsa T. Peric, Radmila Sekulic, Milos Jordanski
Teacher Coordinator: Milan Tuba

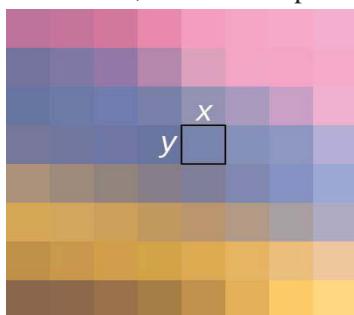
Abstract

In this paper we will introduce a way how to detect edges in real-time for video capture. Edge detection is a fundamental part of many algorithms, both in image processing and in video processing. Therefore it is important that the algorithm is efficient and fast to carry out. Efficiency is of particular importance when processing is performed on live coverage of an event where we need to make some conclusions. One example that motivated us is live coverage of a football match when we need to decide if the ball has crossed the line or if we need to track player movement or calculate some statistics like ball possession and other. In this paper we will describe edge detection methods for image processing and how will we use it for video capture. Also we shall present a fast working solution for this problem.

Keywords: edge, detection, real, time, video, image, processing

1 Introduction

An image may be defined as two-dimensional function $f(x, y)$, where x and y are spatial coordinates, and the amplitude of f at any pair of coordinates (x, y) is called intensity of gray



level of the image at that point. When x , y and the intensity values of f are all finite, discrete quantities, we call the image a digital image. We should notice that every digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are called *picture elements*, *pels*, and *pixels*. *Pixel* is the term used most widely to denote the elements of digital image.

Now we work with a digital image composed of pixels. Our goal is to do edge detection for video capture. The main “problem” is that the video capture seems to be much more complicated than

Fig 1: Pixel in an image

a digital image. The problem of detecting edges in video capture can be reduced to the problem of detecting edges in a image by representing the video capture as a series of consecutive images, which will then be processed individually. The main challenge of this approach is speed, because we need to process and display the “edge” image for almost the same time that’s needed to display it in the video capture. In the next section we will describe how to detect edges in a digital image.

2 Edge Detection

2.1 Background

Edge pixels are pixels at which the intensity of an image function changes abruptly, and edges (or edge segments) are sets of connected edge pixels. Edge detectors are local image processing methods designed to detect edge pixels. A line may be viewed as an edge segment in which the intensity of the background on either side of the line is either much higher or much lower than the intensity of line pixels.[8]

The purpose of edge detection is to localize variations such as discontinuities, local extrema and 2D features where at least two edges meet. There are many ways to perform edge detection. However, the most may be grouped into two categories: gradient and Laplacian.

The gradient method detects edges by looking for the maximum and minimum in the first derivative of the image. The Laplacian method searches for zero crossings in the second derivative of an image to find the edges [11].

However, differentiation of an image is an ill-posed problem; image derivatives are sensitive to various sources of noise and quantification effects[9]. An edge detector accepts discrete, digitalized images as input and produces an edge map as output. In addition, we will propose several types of edge detectors.

2.1.1 Laplacian edge detector

Laplacian operator is second derivative operator used in edge detection. The Laplacian of an image highlights regions of rapid intensity change.

The Laplacian $L(x,y)$ of an image $f(x,y)$ is given by:

$$L(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$

Two commonly used kernels are :

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

2.1.3 Roberts Cross operator

The Roberts Cross operator is one of first edge detectors . In its most common usage, the input to the operator is a grey scale image, as is the output. As differential operator, the idea

behind the Roberts cross operator is to approximate the gradient of an image through discrete differentiation which is achieved by computing sum of the squares of differences between diagonally adjacent pixels.

Roberts proposed following equations:

$$y_{i,j} = \sqrt{x_{i,j}}$$

$$z_{i,j} = \sqrt{(y_{i,j} - y_{i+1,j+1})^2 + (y_{i+1,j} - y_{i,j+1})^2}$$

where x_{ij} is intensity of an image. The result of an operation will highlight changes in intensity in diagonal direction. One of most appealing aspects is its simplicity; the kernel is small and contains only integers, but this filter suffers greatly from sensitivity noise.

2.1.5 Sobel operator

The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial gradient that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grey scale image.

The operator consists of pair 3x3 convolution kernels as shown. One kernel is simply the other rotated by 90°.

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these G_x and G_y). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

Typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy|$$

which is much faster to compute.

The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$\theta = \arctan(Gy/Gx)$$

In this case, orientation 0 is taken to mean that the direction of maximum contrast from black to white runs from left to right on the image, and other angles are measured anti-clockwise from this [12].

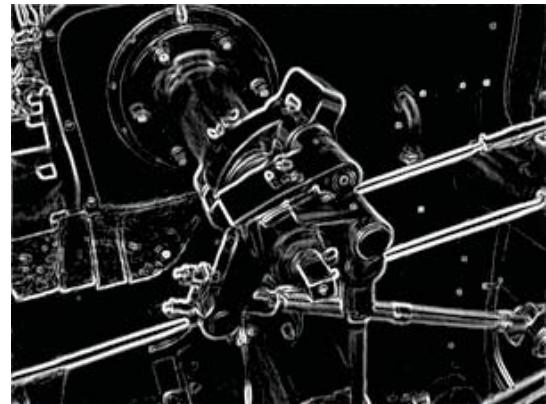
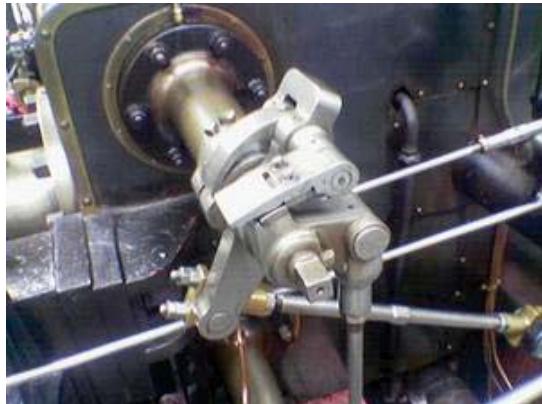


Fig 2,3: Image before and after Sobel edge detection algorithm applied.

2.1.4 Prewitt operator

Prewitt Operator is another operator that can be used for edge detection. However Prewitt Operator is more inclined towards finding the horizontal and vertical edges and then using both of these intermediate result to generate a final image [7]. The Masks used for this operator are as below.

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	-1



Fig 4.5: Image before and after Prewitt edge detection algorithm applied.

2.2 Canny Edge detection algorithm

The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only once when a single edge exists for minimal response [1]. According to Canny, the optimal filter that meets all three criteria above can be efficiently approximated using the first derivative of a Gaussian function[5].

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

$$\frac{\partial G(x, y)}{\partial x} \alpha x e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad \frac{\partial G(x, y)}{\partial y} \alpha y e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2)$$

The first stage involves smoothing the image by convolving with a Gaussian filter. This is followed by finding the gradient of the image by feeding the smoothed image through a convolution operation with the derivative of the Gaussian in both the vertical and horizontal directions. The 2-D convolution operation is described in the following equation:

$$f_{filtered}(x, y) = g(k, l) \otimes f_{original}(x, y) = \sum_{k=-N}^{N} \sum_{l=-N}^{N} g(k, l) \otimes f_{original}(x-k, y-l) \quad (3)$$

Where: $g(k, l)$ = convolution kernel, $f_{original}(x, y)$ = original image, $f_{filtered}(x, y)$ = filtered image, and $2N + 1$ = size of convolution kernel.

Both the Gaussian mask and its derivative are separable, allowing the 2-D convolution operation to be simplified. This optimization is not limited to software implementation only, but applies to hardware implementation as well, as shown in the next section.

The non-maximal suppression stage finds the local maxima in the direction of the gradient, and suppresses all others, minimizing false edges. The local maxima is found by comparing the pixel with its neighbors along the direction of the gradient. This helps to maintain the single pixel thin edges before the final thresholding stage.

Instead of using a single static threshold value for the entire image, the Canny algorithm introduced hysteresis thresholding, which has some adaptivity to the local content of the image. There are two threshold levels, t_h , high and t_l , low where $t_h > t_l$. Pixel values above the t_h value are immediately classified as edges. By tracing the edge contour, neighboring pixels with gradient magnitude values less than t_h can still be marked as edges as long as they are above t_l . This process alleviates problems associated with edge discontinuities by identifying strong edges, and preserving the relevant weak edges, in addition to maintaining some level of noise suppression. While the results are desirable, the hysteresis stage slows the overall algorithm down considerably[10].

The performance of the Canny algorithm depends heavily on the adjustable parameters, σ , which is the standard deviation for the Gaussian filter, and the threshold values, t_h and t_l . σ also controls the size of the Gaussian filter[3]. The bigger the value for σ , the larger the size of the Gaussian filter becomes. This implies more blurring, necessary for noisy images, as well as detecting larger edges. As expected, however, the larger the scale of the Gaussian, the less accurate is the localization of the edge. Smaller values of σ simply a smaller Gaussian filter which limits the amount of blurring, maintaining finer edges in the image[4]. The user can tailor the algorithm by adjusting these parameters to adapt to different environments with different noise levels.

Results can be further improved by performing edge detection at multiple resolutions using multi-scale representations, similar to the Marr-Hildreth algorithm [2]. This is achieved using different standard deviations, which correspond to different resolution versions of the image. Edges have zero crossing at multiple scale values. Combining maxima information from different scales allows better classification of true edges. Convolution at multiple resolutions with large Gaussian filters require even more computation power. This may prove to be challenging to implement as a software solution for real-time applications.

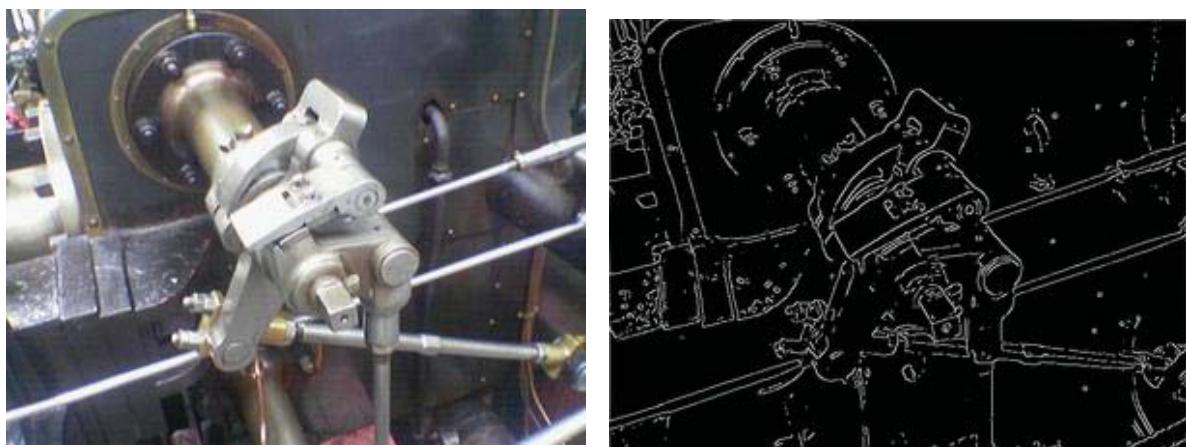
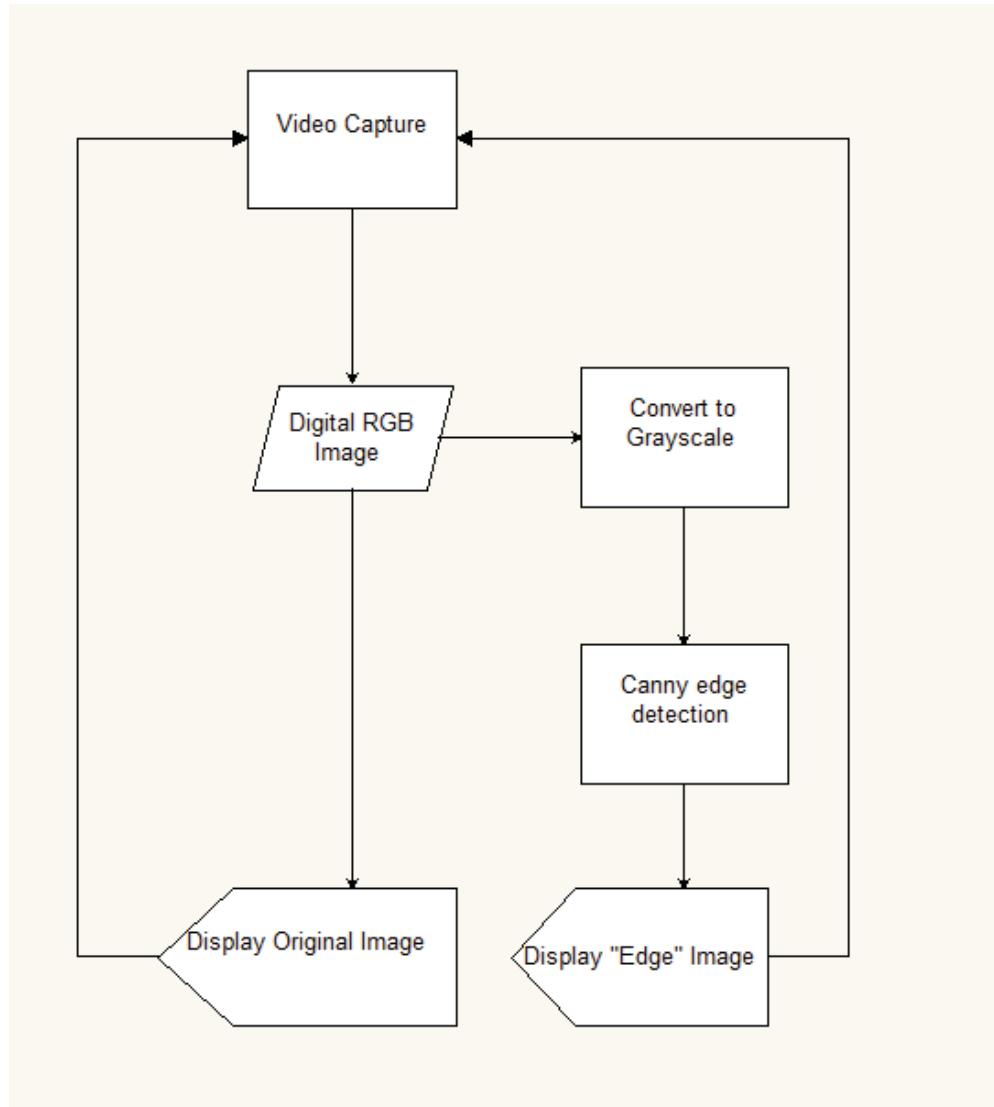


Fig 6.7: Image before and after Canny edge detection algorithm applied.

3 The scope of application

The input of the application is a video capture from camera. Output of the application are two displays of the video capture, the real video and the “edge” video capture. The input video capture is read frame by frame. Every frame is converted to grayscale image. On the grayscale image is the canny filter applied. At the end the frame is displayed.



3.1 Technology used

The application was developed using C# programming language 4.0 on .NET platform and OpenCV library. C# is an OOP language which appeared in 2000, used by software developers for building .NET applications.

4 Conclusion

This paper introduced the concept of edge operators, and how they are used for detection in real-time application. Edge detection algorithms in general have a very important limitation and that is the video quality. If the quality of the video capture is not good enough, then the quality of the result will not be good enough. This approach for edge detections for real-time video capture can later be used for more complex transformations of video capture like shape recognition and other

References

- [1] E. Argyle, "Techniques for edge detection," Proc. IEEE, vol. 59, pp.285–287, 1971.
- [2] A. Rosenfeld, "A nonlinear edge detection technique," Proc. IEEE, vol.58, pp. 814–816, 1970
- [3] A. Rosenfeld and M. Thurston, "Edge and curve detection for visual scene analysis," IEEE Trans. Comput, vol. C-20, pp. 562–569, 1971.
- [4] Lindeberg T., "Edge Detection and Ridge Detection with Automatic Scale Selection", International Journal of Computer Vision, vol. 30, number 2, 117-154, 1998
- [5] J. Canny, "Finding edges and lines in images," MIT AI TR-720, 1983
- [6] F. van der Heijden, Edge and line feature extraction based on covariance models, IEEE Trans. Pattern Anal. Mach. Intell. 17 (1), 16–33, 1995.
- [7] W. E. Higgins and C. Hsu, Edge detection using 2d local structure information,Pattern Recognit. 27 (2), 277–294, 1994.
- [8] V. S. Nalwa and T. O. Binford, On detecting edges,IEEE Trans. Pattern Anal. Mach. Intell.PAMI-8, 1986, 699–714
- [9] Scott Konishi, Alan Yuille, James Coughlin, and Song Chun Zhu, "Statistical edge detection:Learning and evaluating edge cues." IEEE Transactions on PAMI , 25(1):57–74, 2003.
- [10] Z. Wu and R. Leahy, "Image segmentation via edge contour finding: A graph theoretic approach," in Proc. IEEE Comput. Soc. Conf. Computer Vision and Pattern Recognition, 1992, pp. 613–619
- [11] Gonzalez, Rafael C., and Richard E. Woods. "Digital image processing." (2002).
- [12]Baxes, Gregory A. *Digital image processing: principles and applications*. New York: Wiley, 1994.

Nebojsa T. Peric
University of Belgrade
Mathematical Faculty
Studentski Trg 16, Belgrade
Serbia
E-mail:
nebojscha.perich@gmail.com

Radmila Sekulic
University of Belgrade
Mathematical Faculty
Studentski Trg 16, Belgrade
Serbia
E-mail:
radmilasekulic90@gmail.com

Milos Jordanski
University of Belgrade
Mathematical Faculty
Studentski Trg 16, Belgrade
Serbia
E-mail:
jordanski90@hotmail.com

Prof. Milan Tuba
University Megatrend Belgrade
Faculty of Computer Science
Bulevar umetnosti 29, N.Belgrade
SERBIA
E-mail: tuba@ieee.org

Probabilistic model for quantitative risk assessment of traffic accidents in Bulgaria

**Kristina Petkova
Lyudmila Dimitrova**

Abstract

Bayesian Networks (BNs) are well suited tool for uncertain knowledge representation. In the present work a BN model for risk assessment of traffic accident in Bulgaria is proposed. AgenaRisk software is applied to evaluate the model parameters. Statistical data is used for traffic accidents in our country. Usage of our model for quantity risk assessment is demonstrated in various situations.

1 Introduction

Traffic safety is an important social problem. A lot of people are injured and/or killed all over the world each day on highway traffic related crashes. The highest price we pay for car crashes is in the loss of human lives, however society also bears the costs associated with motor vehicle accidents. In order to have higher traffic security, careful analysis of the causes and the consequences of all events that can lead to traffic accidents is needed. These accidents are result of a combination of human, technical, organizational and environmental factors. The amount of information that must be taken into account is huge and thus hardly manageable without computer aid. On the other hand, modelling such complex system involves many uncertainties originated from randomness of the most of the factors that contribute to traffic accidents.

An efficient, compact and intuitive knowledge representation for handling uncertainty can be done with a Bayesian networks (BNs). BNs are a powerful modeling and analysis tool especially due to their ability to consider the stochastic variations of the data and to perform bi-directional induction in decision-making. BNs in general and works related to traffic accidents are explained in section 1.

In section 2 the available statistics data is represented and the BN model is created and explained. In the last section 3 examples of the model usage are listed.

1.1. Bayesian Networks

Bayesian networks belong to a more general class of models called probabilistic graphical models[1] that arise from the combination of graph theory and probability theory. These models are widely used due to the option for complex probabilistic models to be decomposed into smaller, susceptible units. A probabilistic graphical model is defined by a graph where nodes represent stochastic variables and arcs represent dependencies among such variables. These arcs are annotated by probability distribution mapping the interaction between the linked variables. A

probabilistic graphical model is called a Bayesian network when the graph has its variables connected in a directed acyclic graph (DAG).

The nodes in a Bayesian network [2,3,4] represent a set of random variables $\{X_1, X_2, \dots, X_n\}$ from the domain. A set of directed arcs (or links) connects pairs of nodes $X_i \rightarrow X_j$ and represents the direct dependencies between variables. Assuming discrete variables, the strength of the relationship between variables is quantified by conditional probability distributions associated with each node. The quantitative part of the network, attached to each node, is known as conditional probability table (CPT). Nodes without incoming arcs, i.e. without parents, are called root nodes and have marginal prior probability distributions. A simple BN with CPT for each node is represented in Figure 1.

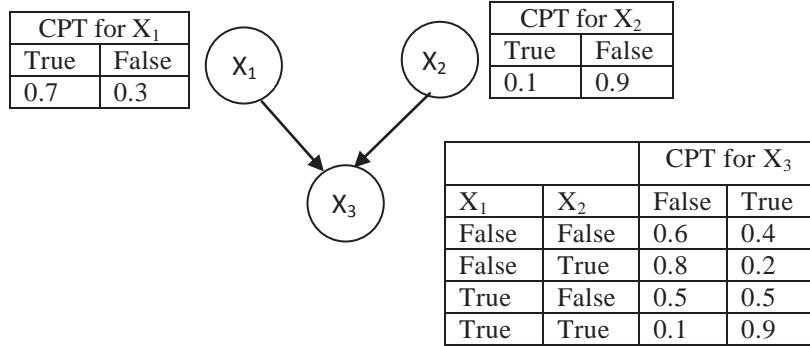


Figure 1. BN describing impact the variables X_1 and X_2 on a third variable X_3

The structure, or topology, of the network should capture qualitative relationships between variables. In particular, two nodes should be connected directly if one affects or causes the other, with the arc indicating the direction of the effect. The experience and knowledge of experts domain can be used to design the structure and the initial values of the relationship coefficients (i.e. CPT) between any two causal nodes of the BN, and the final appropriate values can be revised through learning algorithms[5,6,7]. Generally, learning a Bayesian network from data consists of the induction of its two different components:

- 1) The graphical structure of conditional dependencies (model selection);
- 2) The conditional distributions quantifying the dependency structure (parameter estimation).

The graphical structure should form the relationships between the nodes. This is done by an expert in the subject. The conditional distributions can be also created by an expert, or statistical data could be used. The onward dependencies should be calculated by the probabilities of the root nodes. These relationships can be represented in the DAG by using the local and global Markov properties [1].

The local Markov property states that each node is independent of its non descendant given the parent nodes and leads to a direct factorization of the joint distribution of the network variables into the product of the conditional distribution of each variable X_i given its parents. Therefore, the joint probability (or density) of the network variables can be written as:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{parent}(X_i)) \quad (1)$$

The probability distribution of a particular random variable is determined by marginalizing the joint probability distribution with respect to this random variable. In the presence of evidence (i.e., the state of one or more random variables is known with certainty), the probabilities propagation algorithms determine the marginal posterior probability for each variable given the evidence.

Two types of inference support of the BN can be used: predictive support for node X_i , based on evidence nodes connected to X_i through its parent nodes (also called top-down reasoning), and diagnostic support for node X_i , based on evidence nodes connected to X_i through its children nodes (also called bottom-up reasoning). In this way, a Bayesian network becomes a complete

simulation system able to forecast the value of unobserved variables under hypothetical conditions and also able to find the most probable set of initial conditions leading to observed situation.

1.2. Related works

BNs have been used by many authors for modelling and analysis of different aspects in traffic accidents modelling. Such model is proposed by Juan de Oña and coworkers who developed a BN to classify traffic accidents according to the severity of the injury [8]. Authors presented an analysis of 1536 accidents on rural highways in Spain, where 18 variables representing contributing factors were used to build 3 different BNs that classified the severity of accidents into slightly injured and killed or severely injured. In their next work [9] they demonstrated a simultaneous use of Latent Class Clustering (LCC) and BN for road safety analysis. LCC is used as a preliminary tool for segmentation of 3229 accidents on rural highways in Granada (Spain) between 2005 and 2008. Next, BNs are used to identify the main factors involved in accident severity for both, the entire database and the clusters previously obtained by LCC.

A BN for extracting data for a real accident has been proposed by Ozbay and co.[10]. They created a dynamic incident duration estimation tree that serves the problem when data might be partially available. The dependency relations among all variables in the BN served as input data for quantitative and qualitative analysis.

Sun [11] has used the BN to form a model for traffic flow predictions. He proposed a very effective model whose advantage is that its' forecasts are still valid both for complete and incomplete input data. Davis and Pei [12] observe traffic accidents as a problem which can be solved with uncertain reasoning. They apply artificial intelligence and physical principles to combine the structural model for the accident and the development of uncertain reasoning. In this model an expert assessment of prior uncertainty for the root variables is used. The evidence is collected at the scenes' accident and then the posterior probabilities are computed by Bayes theorem.

An incident detection algorithm for urban arterial roads model has been proposed by Zhang and Taylor [13]. Their algorithm uses the BN to represent and detect the patterns of deterioration in traffic conditions which lead to lane-blocking incidents. Lane volume and occupancy is captured by newly proposed detector configuration. An incident detection algorithm is proposed for traffic signal scheduling. The BN is dynamic – it is constructed and trained to detect incidents. Algorithm performance is assessed by offline tests using simulated data. These tests showed promising results which leads to the conclusion that the BN modeling approach is successful. William Marsh [14] at all. used a BN model to describe the organizational causes of traffic accidents.

Deublein M at all. [15] proposed a new methodology for road accident occurrence prediction. The observed frequencies of accident occurrence are used as an input for multivariate regression analysis to establish the probabilistic BN. With this as a base and using updating algorithms, parameter learning is done. Using this learning the model becomes conditional on the values of the risk indicating variables. The authors illustrated the methodology with the use of a case study based on data from the Austrian rural motorway network.

Antony Cyril Mbakwe [16] developed a BN model for possible traffic safety responses to lower the high traffic related injury and fatality rates in Nigeria. He used data obtained from Delphi database and included the major factors for accident occurrence – poor road condition, road obstruction, poor traffic control, drunk driving, car failure and driver fatigue.

BN model for traffic accident risk assessment is the purpose of the current paper. The accident depends on different intrinsic and extrinsic factors. This model will be used for prognostic reasoning, based on the specific factors which cause vehicle accidents in Bulgaria.

2 Model development

2.1 Statistical data for traffic accidents on the roads in Bulgaria

Accident data is present on the site of public consultative committee on issues of road safety in Bulgaria [17]. In this paper we have used data from the year 2013. During that year 47488 traffic accidents have happened, 45851 (or 96.55%) of which are caused by the driver.

2.1.1 Statistics by accident cause

The causes for traffic accidents sorted decreasingly by their influence are represented in Figure 2.

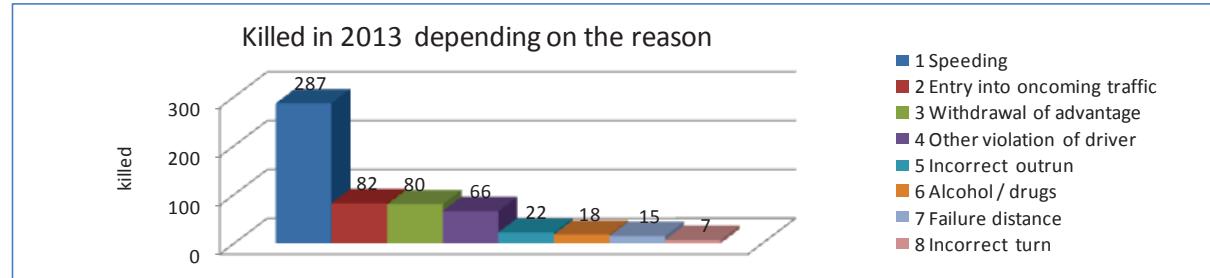


Figure 2. Statistical data sorted by traffic accident cause

Violations of the traffic rules, starting with speeding, are the main reason for traffic accidents. Alcohol and drug abuse are on 6th place. The extent of the rule violation is expressed by the ratio of people killed to all the accidents sorted by the accident cause. This is represented in Figure 3.

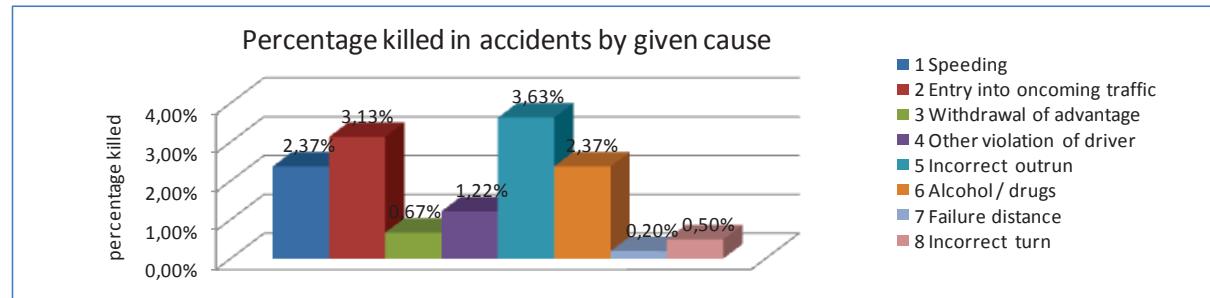


Figure 3. Fatality rate for different accident causes

2.1.2 Statistic by region

These data give us the ability to evaluate the traffic conditions in the 28 regions in Bulgaria. In order to simplify the parameters in our model these regions were coupled in 7 groups as Figure 4. represents. The ratio of killed to overall accidents count has top value in the region with the hardest traffic conditions – Plovdiv. In Plovdiv this ratio is 8.4 times higher than in the safest region.

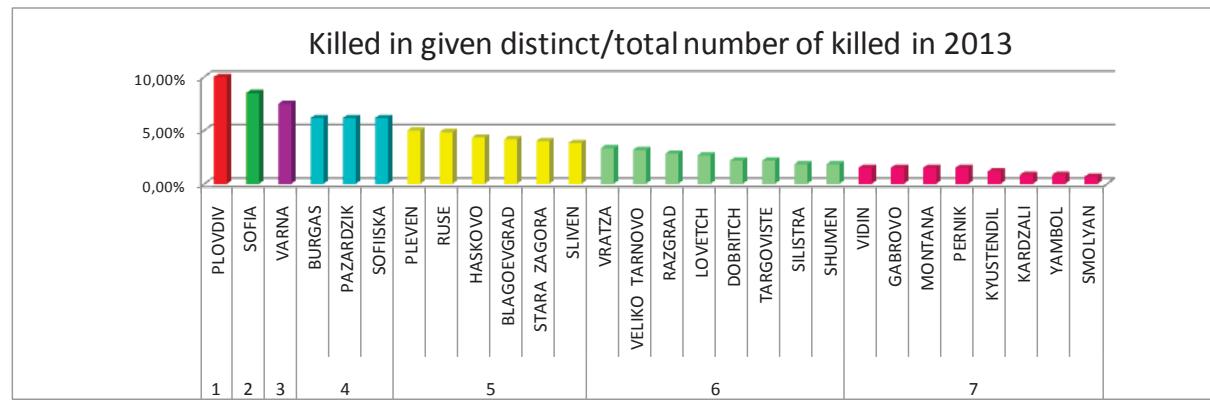


Figure 4. Percentage ration of killed to overall accidents count sorted by region

2.1.3 Statistics by hour and weekday

These data visualizes the influence of traffic flow on the accidents occurrence. Data is grouped in five time zones as presented on Figure 5. The percentage of killed in a traffic jam between 18h and 19h is 7.5 times higher than in the safest zone.

In Figure 6 data is grouped by weekday. The percentage of killed is highest on Friday and in the weekend - 1.4 times more people are killed at the end of the week compared to the rest of the week.

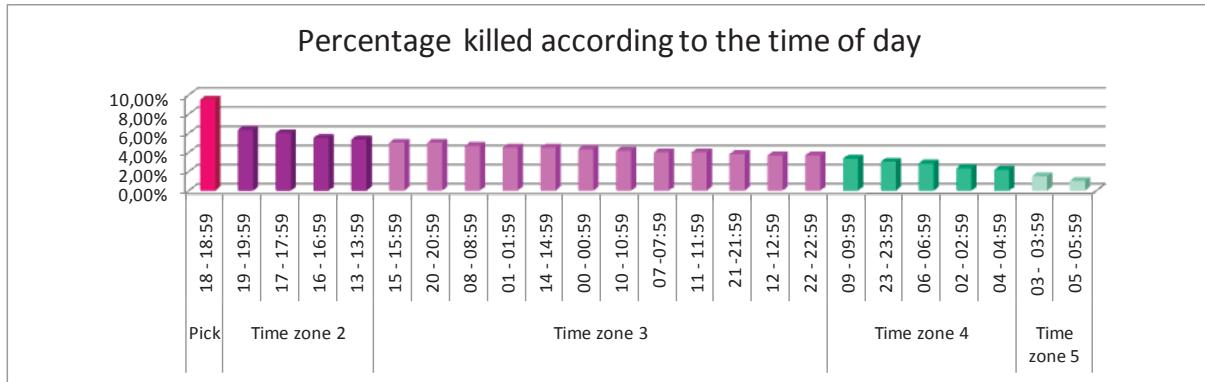


Figure 5. Ratio killed to overall accidents count sorted by time of the day

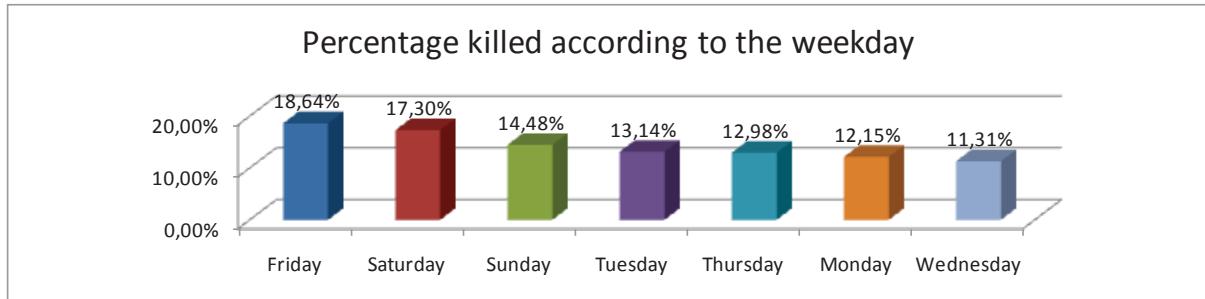
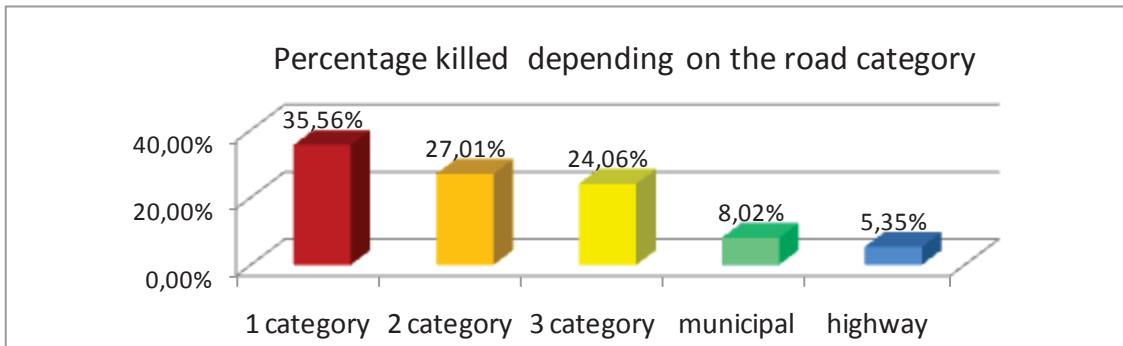


Figure 6. Ratio killed to overall accidents count sorted by day of week

2.1.4. Statistics by road condition category

On the category 1 road there is 6.65 times more killed than on the highway. This ratio is represented in Figure 7. The category with best driving conditions is highway and with worst is category 1.



2.1.5 Statistics by road lighting condition

The highest ratio of killed to overall accidents count is during daylight because this is the time with highest traffic. This is represented in Figure 8. In figure 9 you can see the ration killed to overall accidents for the current light. This means if there are 100 accidents in darkness 5 of them are with fatal outcome, whereas if there are 100 accidents in daylight only one is fatal. By these statistics data we could conclude that driving in the darkness is 5 times more risky than driving in daylight or in artificial light.

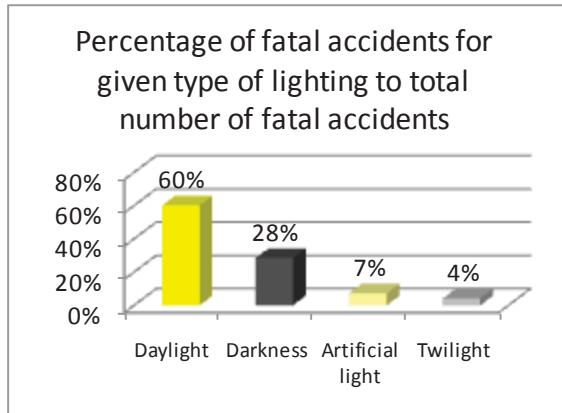


Figure 8. Percentage of fatal accidents for given type of lighting to total number of fatal accidents

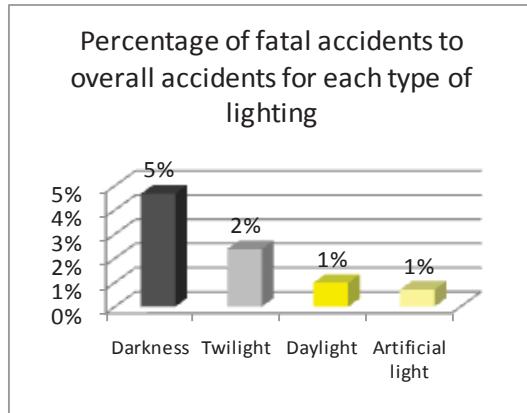


Figure 9. Percentage of fatal accidents to overall accidents for each type of lighting

2.1.6 Statistics by drivers' sex

Figure 9 and 10 represent that male drivers are causing more accidents than female drivers. At Figure 9 the ratio of killed female drivers and passengers to all victims is presented, whereas in Figure 10 the ratio only of female passengers killed to all victims.

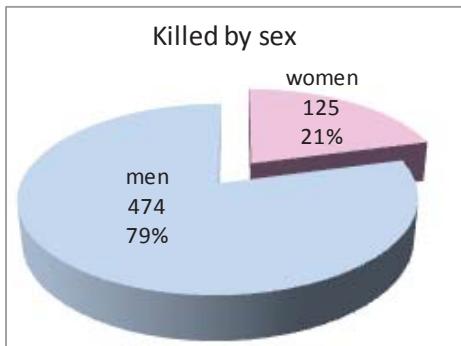


Figure 9. Killed drivers and passengers by sex

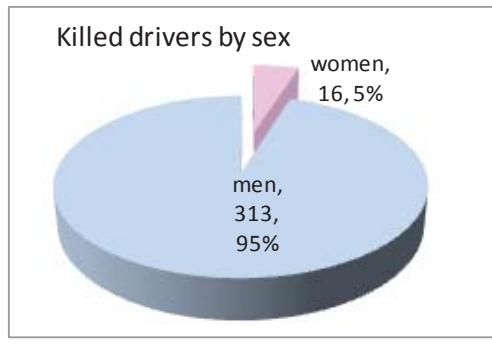


Figure 10. Killed drivers by sex

2.1.7 Statistics by drivers' age

Figure 12 represents ratio of drivers killed to overall drivers killed sorted by age group. The low number of drivers killed in the age group 18-20 can be explained with the lower number of drivers in this group. Drivers killed in the age group of 21-24 are 2 times more than the drivers killed in the age group above 64.

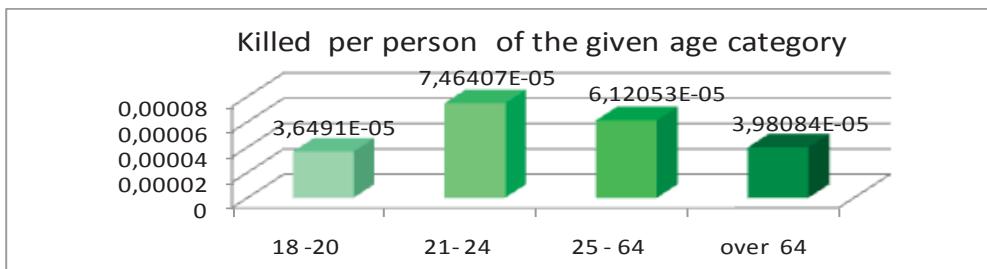


Figure 12. Ratio killed to overall drivers killed sorted by age group

All the data discussed in these seven types of sorting is used to evaluate the influence of different factors during elicitation of the network parameters.

2.2. Bayesian network model for traffic accident risk assessment

2.2.1 Model structure definition

Before defining the structure of the model we have to clarify the root causes for the accidents, the consequences and the relations among them.

Along with the factors for which statistics is available, we also included other underlying factors, as explained in the literature. Personal psychological characteristics are believed to classify a driver as reliable or unreliable, thus capable or incapable for driving [18]. Unreliable drivers commit the following mistakes:

- Unintentional errors – lack of attention, incorrect perception, lack of proper reaction, improper judgement for traffic conditions in general. These errors are caused by insufficient level of specific psychological qualities.
- Conscious decisive errors – the driver is aware that his decision can lead to a dangerous traffic situation but still commits the action. These decisions are caused by personal psyche.

Behavioral factors are believed to be the main reason for three out of five accidents and to highly contribute to the remaining two as stated by Pertidou at all. [19].

In the present paper we have summarized the underlying factors grouped in the following four categories:

1. Factors caused by behavioral mistakes. This group consists of four subcategories:
 - a. Physical and individual personal characteristics: *Sex, Age, Psychological characteristic, Discipline*. Going through casual paths, shown on the network, these factors lead to the driver becoming dangerous for the others on the road – node *Dangerous behavior*
 - b. Drivers' qualification: *Experience, Skills and Past incidents*
 - c. Drivers' current condition: *Physical condition* of the driver: *Lack of sleep, Fatigue, Illness*

Combined influence of these factors determines the first factor considered risky on the road: *Risk human factor*.

2. Factors caused by travel time - node *Risk travel time*.

Statistics give us relations between the chosen hour of the day and day of the week and the risk for accident. The season is also included as a factor – research was conducted with conclusion that accidents percent rises with 8% between November and March [20].

3. Factors causing the state of node *Driving condition*.

First sub-category of factors influencing the driving conditions – region, city, road category, determines the complexity of road network in the country. Influence rate of region and category was discussed in section 2.1. As cited from [20] due to speeding violations accidents outside cities are much more than inside cities – 1.78 times more people are killed. Climate also affects driving conditions.

4. Factors determining vehicle condition: brakes condition, tires condition, lights – node *Technical malfunction*.

Finally we can calculate the weighted average risk influenced by all the four groups mentioned above – node *Overall risk*. The outcome of the accident - node *Outcome*, may be less dangerous based on the precautions determining the security level - node *Protection*. The resulting model is represented in Figure 13 in its' four sub-networks – one per group.

2.2.2 Defining model parameters

The model is developed with AgenaRisk software [21]. Very convenient feature of this tool is the broad possibility to define CTP (in this product referred as NPT – node probability table), with the usage of functions. Possible number of functions depends on the node type - Labelled, Boolean, Continuous Interval, Ranked, Integer Interval or Discrete Real. In this model nodes are of type Ranked. Ranked nodes are mapped to underlying numerical scale. This means that no matter what the state labels are and how many states a node has, there is an assumption that there is an underlying numerical scale that goes from 0 to 1 in equal intervals. By defining the ranked scale the user must be sure that the labeling of the states is consistent from worst to best. In our case in the beginning of the scale are the values of the parameters that have the highest influence over the risk of an accident. For example, to model *Risk travel time* in terms of factors *Day of the week* in consistent way, days must be ordered from worst to best, as shown on Figure 6. Thus, states of the node *Risk travel time* should be ordered from highest to lowest. This is done in a 3 state scale – high, medium, low.

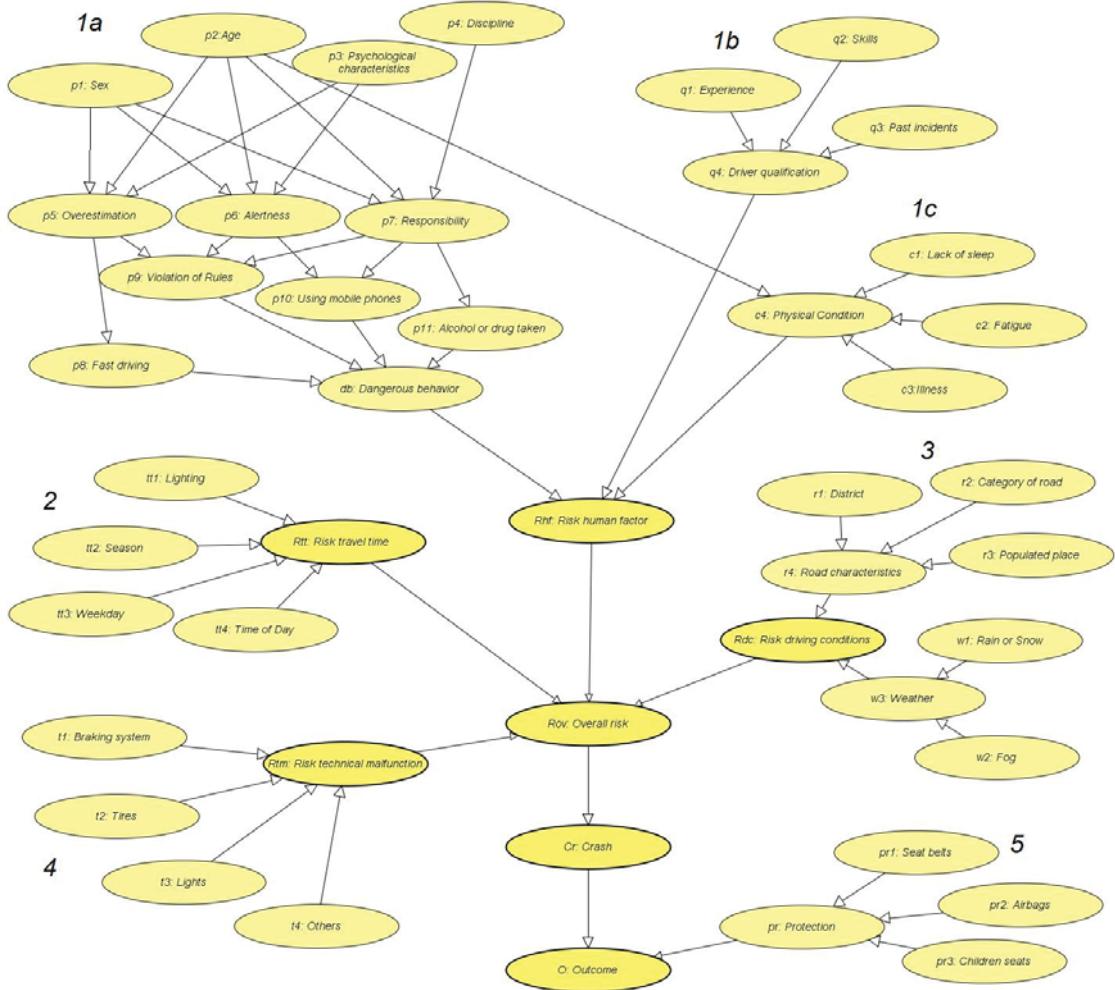


Figure 13. BN model for traffic accidents risk assessment

Due to the underlying numerical scale of the ranked nodes, we can define numerical statistical distribution expressions on it. Especially useful for defining NPT is truncated Normal distribution (TNormal). Unlike the regular Normal distribution TNormal has finite end-points. For ranked nodes whose endpoints are 0 and 1, respectively. Like the Normal distribution, the TNormal is characterized by two parameters: the mean and variance. This enables us to model variety of distribution shapes.

Node Probability Table	
NPT Editing Mode	Expression
Expression parameters take the form of standard mathematical expressions and can include node names (available by right-clicking in the parameter's text field).	
If a parameter is badly formed, the text field will have a red border. You can find out the problem by holding the mouse over the field.	
Expression Type	TNormal
Mean	$(8*r1+6*r2+2*r3)/16$
Variance	0.01

Figure 14. Setting NPT for the node Risk driving conditions

In our model factor influence is evaluated as a ratio of killed in the worst situation to killed in the best situation, which is given with the corresponding weight factor. Figure 14 represents parameter settings of node *Risk driving conditions*. Parent nodes are given by r1, r2 and r3 - district, road category and populated place for this specific node (Fig. 13). Weights of r1 and r2 are determined

based on the statistical data, as represented in Figure 4 and Figure 7. Weight of r3 is determined based on the data cited by [20].

Using the same technique the other NPT of all internal and leaf nodes are evaluated. Prior probability distributions of the root nodes are calculated or estimated. States of each node and the parameters for NPT determination for different risk factors are listed in Tables 1 to 5:

Table1. Properties of node *Risk travel time* and its parents

Node name	Node ID	Node states	Node CPT
Lighting	tt1	darkness, twilight, daylight, artif.light	(0.1, 0.05, 0.8, 0.05)
Season	tt2	nov-mar, apr-oct	(0.5,0.5)
Weekday	tt3	fri-sun, mon-thu	(0.43, 0.57)
Time of the day	tt4	tz1, tz2, tz3, tz4, tz5	(0.04, 0.17, 0.5, 0.21, 0.08)
Risk travel time	tt5	high, medium, low	TNormal(2tt1+1.5tt2+1.5tt3+5tt4)/10, 0.01)

Table 2. Properties of node *Risk human factor* and its direct and undirect parents

Node name	Node ID	Node states	Node CPT
Sex	p1	male, female	(0.49, 0.51)
Age	p2	18-20, 21-24, 25-64, over 64	(0.01, 0.1, 0.74, 0.15)
Psych.characteristics	p3	unreliable, medium, reliable	(0.3, 0.5, 0.2)
Discipline	p4	low, medium, high	(0.3, 0.5, 0.2)
Overestimation	p5	yes, no	TNormal((p1+2p2+3p3)/6, 0.01)
Alertness	p6	low, medium, high	TNormal((p1+2p2+4p3)/7, 0.01)
Responsibility	p7	low, medium, high	TNormal((p1+p2+3p4)/5, 0.01)
Fast driving	p8	yes, no	((0.6, 0.4), (0.05, 0.95))
Violation of rules	p9	yes, no	TNormal((p5+p6+p7)/3, 0.01)
Using mobile phone	p10	yes, no	TNormal((p6+p7)/3, 0.01)
Alcohol/drug taken	p11	yes, no	((0.1,0.9),(0.03,0.97),(0.001,0.999))
Dangerous behavior	p12	yes, no	TNormal((p8+2p9+p10+3p11)/6, 0.01)
Experience	q1	low, medium, high	(0.2, 0.6, 0.2)
Skills	q2	low, medium, high	(0.15, 0.5, 0.35)
Past incidents	q3	high, medium, low	(0.2, 0.4, 0.4)
Driver qualification	q4	low, medium, high	TNormal((q1+q2+2q3)/4, 0.01)
Lack of sleep	c1	yes, no	(0.3, 0.7)
Fatigue	c2	yes, no	(0.25, 0.75)
Illness	c3	yes, no	(0.1, 0.9)
Physical condition	c4	bad, good, excellent	TNormal((p1+c1+c2+c3)/4, 0.01)
Risk human factor	Rhf	high, medium, low	TNormal((3*p12+2*q4+c4)/6, 0.01)

Table 3. Properties of the node *Risk driving condition* and its parents

Node name	Node ID	Node states	Node CPT
Distinct	r1	d1, d2, d3, d4, d5, d6, d7	(0.07, 0.08, 0.06, 0.15, 0.22, 0.21, 0.21)
Category of road	r2	class2-3, class1, municipal, highway	(0.5, 0.25, 0.15, 0.1)
Populated place	r3	yes, no	(0.2, 0.8)
Road characteristics	r4	difficult, normal, easy	Tnormal((8r1+6r2+2r3)/16, 0.01)
Rain or snow	w1	yes, no	(0.15, 0.85)
Fog	w2	yes, no	(0.2, 0.8)
Weather	w3	unfavorable, favorable	Tnormal((w1+w2)/2, 0.01)
Risk driving cond.	Rdc	high, medium, low	Tnormal ((2*r4+w3), 0.01)

Table 4. Properties of node *Risk technical malfunctions* and its parents

Node name	Node ID	Node states	Node CPT
Breaking system	t1	yes, no	(0.02, 0.98)
Tires	t2	yes, no	(0.1, 0.9)
Lights	t3	yes, no	(0.02, 0.98)
Other	t4	yes, no	(0.05, 0.95)
Risk technical malfunction	Rtm	high, medium, low	TNormal((5t1+3t2+t3+t4)/10, 0.01)

Table 5. Properties of node *Protection* and its parents

Node name	Node ID	Node states	Node CPT
Seatbelts	pr1	yes, no	(0.1, 0.9)
Airbags	pr2	yes, no	(0.3, 0.7)
Children seats	pr3	yes, no	(0.05, 0.95)
Protection	pr4	high, medium, low	TNormal((pr1+pr2+pr3)/3, 0.01)

NPT for the *Overall risk* node is defined as TNormal ((2Rh_f+Rdc+Rtt+Rtm)/5, 0.01).

3 Model usage example

Prior marginal probability distribution in the network for given approximated marginal probability distribution of the root nodes is represented on Figure 15. Marginal distribution for node *Overall risk* in percents is (high=4.42%, medium=60.39%, low=35.19%). This risk distribution leads to crash and outcome with marginals (high=12.89%, medium=48.15%, low=38.96) and (severe injury=0.65%, mild injury=5.325%, lucky=94.007%), respectively.

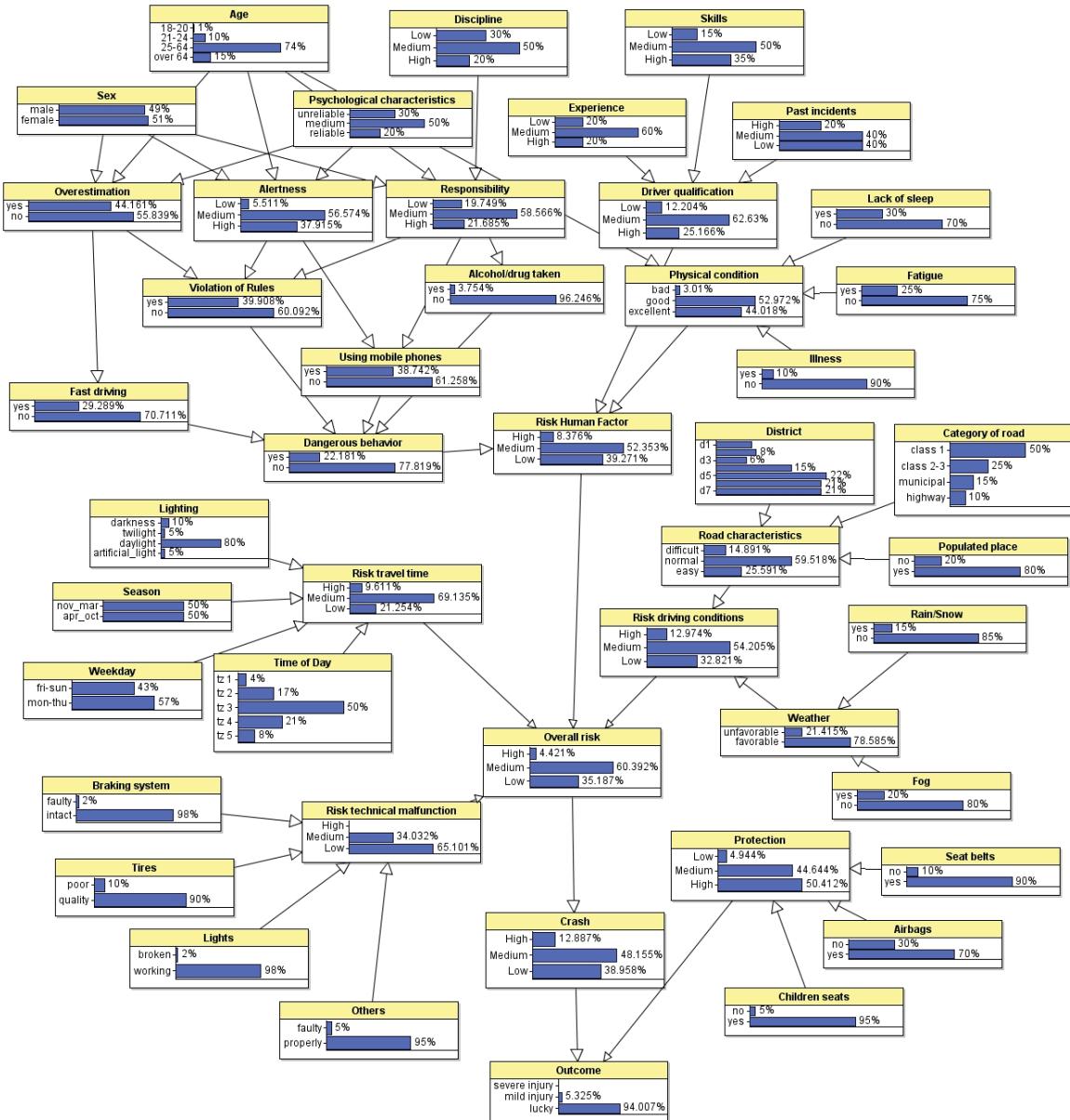


Figure 15. Prior marginal probability distribution of the network

Figure 16 gives the posterior marginal probability distribution in two scenarios - unfavorable (Scenario 1) and favorable (Scenario 2). In Scenario 1 parameters leading to higher incident risk are set - *Sex=male*, *Age=21-24*, *Discipline=medium*, *Skills=low* etc. In Scenario 2 parameters leading to lower incident risk are set - *Sex=female*, *Age=25-64*, *Discipline=high*, *Skills=medium* and so on. The model shows considerably lower risk in the second scenario.

In Scenario 1 the probability for the variable value for *Overall risk* to be *high* rises up to 25.875%, which is 5.81 times higher than the apriori value of 4.427%. This leads to rise of accident risk probability with *Outcome = severe injury* from 0.65% up to 18.254% while *Outcome=lucky* is only 42.625%. In Scenario 2 the probability for the variable value for *Overall risk=high* drops to 0.89%, the probability for *Outcome=lucky* rises up to 99.69%, while *Outcome=severe injury* drops to a value close to 0. These results are presented in Figure 16.

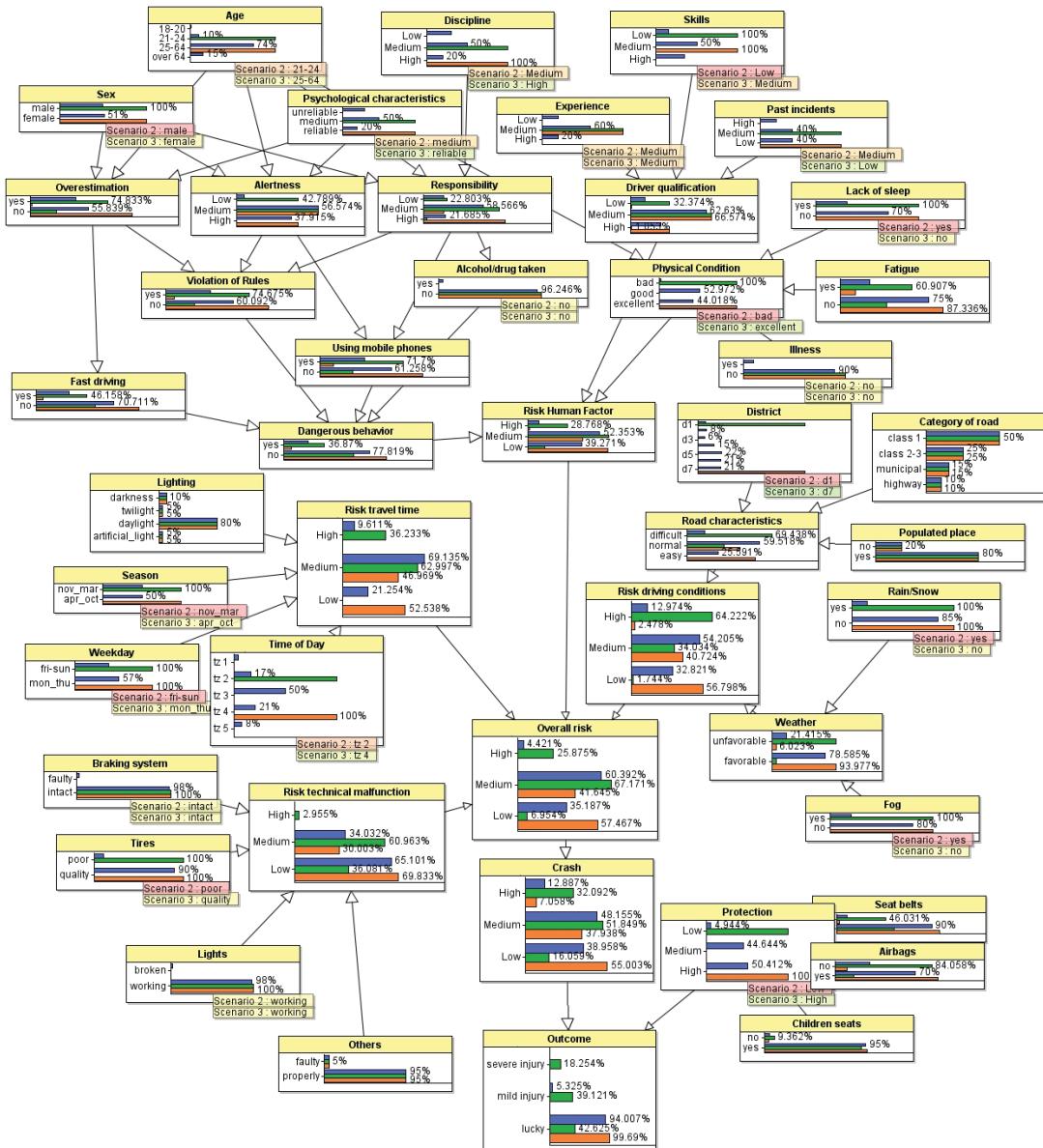


Figure 16. Posterior marginal probability distribution for two scenarios

4 Conclusions and future work

We have proposed a BN model for traffic accident risk assessment in Bulgaria. Truncated Normal distributions - TNormal (mean, variance) are used to generate CPTs of the intermediate and leaf

nodes of the model. The first parameter - mean of each node distribution - is determined using statistical data for evaluating the weight of the causal factors influencing this node. The proposed model can be used for quantitative assessment of accident occurrence based on given intrinsic and extrinsic travel factors – this could change the decision of a driver to start or defer a journey. Refining of the interpretation of statistical data for underlying factors weight, as well as learning and validation of the model, is needed for more accurate predictions.

References

- [1] S. L. Lauritzen, *Graphical Models*. Oxford University Press, Oxford, UK, 1996
- [2] U.B. Kjaerulff, A.L. Madsen, „*An Introduction to Bayesian Network and Influence Diagram*”, 2005 [Online] Available: <http://people.cs.aau.dk/~uk/papers/pgm-book-I-05.pdf>
- [3] U. B. Kjærulff, and A. L. Madsen, “*Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*”, Information Science and Statistics, vol. 22, 2013
- [4] A. Darwiche, “*Modeling and Reasoning with Bayesian Networks*”, Cambridge University Press; 562 p., 2009
- [5] D. Heckerman, D. Geiger, and D. M. Chickering, “*Learning Bayesian networks: The combination of knowledge and statistical data*”, Machine Learning , vol. 20 (3), pp 197-243, 1995
- [6] D. Heckerman, “*A Tutorial on Learning with Bayesian Networks*”, Innovations in Bayesian Networks”, Studies in Computational Intelligence, vol. 156, pp 33-82, 2008
- [7] S.L. Lauritzen, “*The EM algorithm for graphical association models with missing data*”, Computational statistics and data analysis, vol.19(2), pp.191-201, 1995
- [8] Juan de Oña, Randa Oqab Mujalli, Francisco J. Calvo, „*Analysis of traffic accident injury severity on Spanish rural highways using Bayesian networks*”, Accident Analysis & Prevention Volume 43, Issue 1, January 2011, pp 402-411
- [9] Juan de Oña, Griselda López, Randa Mujalli, Francisco J. Calvo, „*Analysis of traffic accidents on rural highways using Latent Class Clustering and Bayesian Networks*”, Accident Analysis & Prevention, Volume 51, March 2013, pp 1-10
- [10] Kaan Ozbay, Nebahat Noyan, „*Estimation of incident clearance times using Bayesian Networks approach*” Accident Analysis & Prevention, Volume 38, Issue 3, May 2006, pp 542-555
- [11] Shiliang Sun, „*A bayesian network approach to traffic flow forecasting*”, Intelligent Transportation Systems, IEEE Transactions on (Volume:7 , Issue: 1), March 2006, pp 124 – 132
- [12] Gary A. Davis, Jianping Pei, „*Bayesian networks and traffic accident reconstruction*”, ICAIL '03 Proceedings of the 9th international conference on Artificial intelligence and law, ACM New York, NY, USA ©2003 pp 171-176
- [13] Zhang, K, Taylor, M A, „*A new method for incident detection on urban arterial roads*”, World Congress on ITS, 11th, 2004, Nagoya, Aichi, Japan
- [14] William Marsh, George Bearfield, „*Using Bayesian Networks to Model Accident Causation in the UK Railway Industry*” Online: <http://www.researchgate.net/publication/228789914>
- [15] Deublein M, Schubert M, Adey BT, Köhler J, Faber MH. „*Prediction of road accidents: A Bayesian hierarchical approach*”, Accident Analysis & Prevention, 2013, 51, pp 274-291
- [16] Cyril Mbakwe, „*Modeling highway traffic safety in Nigeria using Bayesian network*”, ProQuest Dissertations and Theses, 2011
- [17] <http://dokkpbdp.mvr.bg/AboutUs/mission.htm>
- [18] Republic of Bulgaria Ministry of transport, informational technology and messaging Executive agency Mobility administration – Psychology in traffic safety <http://www.rta.gov.bm/images/Image/rtoldsite/novini/nk/dvd.pdf>
- [19] Petridou E, Moustaki M. „*Human factors in the causation of road traffic crashes*” Eur J Epidemiol. 2000;16(9), pp 819-826
- [20] Statistical data for traffic accidents <http://press.mvr.bg/NR/rdonlyres/899919CF-716C-4848-B418-48F599FBCDDF/0/statistika.doc>
- [21] Bayesian Networks and Simulation Software for Risk Analysis and Decision Support, online: <http://www.agenerisk.com/>

Kristina PETKOVA
 Technical University Sofia
 Faculty of Industrial Technology
 bul.Kliment Ohridski 8
 Sofia 1756, Bulgaria
 E-mail: kristinapetkova.tu@gmail.com

Lyudmila DIMITROVA
 University”Prof. A. Zlatarov”
 Dep. of Computer and Information Technology
 Prof. Yakimov street 1
 Burgas 8010 Bulgaria
 E-mail: lyudim@gmail.com

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Screen Resolution Invariant Pattern Matching

**Kristina Petkova
Teacher Coordinator: Assoc. Prof. PhD Milena Lazarova
Consulting Expert: Assoc. Prof PhD Hristo Aladjov**

Abstract

Pattern matching is a common problem in navigation through window-based software. An operation is required each time a button, link or slider should be located in the user interface. For example, the mouse has to be positioned over the item, then clicking, dragging or dropping should be executed. The navigation process could further be automated in order for software to be able to navigate for the user. Automatic navigation in the user interface is very convenient when defining scripts for regularly executed routines. Software systems do not provide possibility for a person to find the desired navigation elements without interaction. This paper proposes a simple but still efficient algorithm for automatic navigation in a software system user interface. The automatic navigation is based on a predefined target template of the user interface layout that is used for pattern matching and positioning over required navigation element. The suggested approach is implemented as a software solution that does not use external software libraries and thus is easily portable and platform independent. It is also optimized to find a fast solution to the pattern matching problem and to provide flexible and efficient automatic navigation.

1 Introduction

Automatic navigation is the ability of a system to find a predefined image in the screen, execute some actions based on that information, follow a sequence, etc. That approach is very useful for systems which lack pointing peripheral devices – mouse for example - but yet need to provide navigation. If there is no automatic navigation the coordinates of desired images could be set explicitly to static values. This, though, would not work if the same program runs on another screen resolution.

Thus the following problem needs to be solved: find a targeted image, referred as template, in a larger image, denoted here as a screenshot, for example the capture of the current screen. In addition the described location process needs to be independent on the screen resolution and the operating system. The solution should also be computationally effective since the automatic navigation required a real-time processing speed; otherwise it is not useful for navigation.

One of the objectives of the solution presented in this paper is to achieve a lightweight software implementation which is independent from external software libraries since it will be the most efficient way to resolve the problem. The other aims of the proposed solutions are to provide colour and scaling invariance.

Automatic navigation is nowadays often used in user interface testing. Several example software systems that are serving this purpose are mentioned in the next sections.

2 Image Matching Software Libraries

In the next paragraphs an overview of the existing solutions for the pattern matching problem are described and analyzed with emphasizing on their advantages and disadvantages for solving the automatic navigation problem:

2.1 Sikuli Script

Sikuli Script is an OpenCV-based UI testing library. It is very easy to use but it contains dependencies to pre-compiled OpenCV libraries which makes its footprint larger than is required for the automatic navigation solution. [3]

2.2 Java OpenCV MatchTemplate

Java OpenCV MatchTemplate is a precompiled library using only the template matching parts of the OpenCV. It would be perfect to use but it is OS-dependant and its' dll-s should be precompiled for all architectures before reusing it. [4]

2.3 AutoIt

AutoIt is a scripting language for automatic navigation in windows desktop applications. It provides automation of all of the user actions, but has the disadvantage to only work under Windows OS [5]

Based on the evaluation of the possibilities to use existing implementation of a solution to the pattern matching problem for automatic navigation the following conclusion has been done – the existing solutions are dependant on large libraries and most of them are platform-dependant. Therefore a new solution should be provided to solve the problem for platform-independent and screen-independent pattern-matching for automatic navigation.

In order to solve the operating system independency the solution will be implemented in Java. Java is a modern widely used object oriented language, which provides cross platform functionality due to its virtual machine basis – the "write once, run anywhere" ability. [8]

3 Image Matching Algorithms

In order to solve the automatic navigation system an overview of the template matching algorithms is done having in mind the required platform flexibility and the colour. In order to solve the basic task for automatic navigation that assumes when the color schemes and the resolutions of the UI template and the screenshot match a correlation based methods can be applied. These are a set of methods for direct image matching that compare the two images on a pixel by pixel basis. The comparison requires to iteratively cropping parts of the screenshot image with the size of the template image that is the target navigation control and evaluating their correspondence and match. The result is a map with differences, the smaller the difference the better the chance a match is found. The comparison can be done using several algorithms, a summary of them in presented on Fig. 3.1

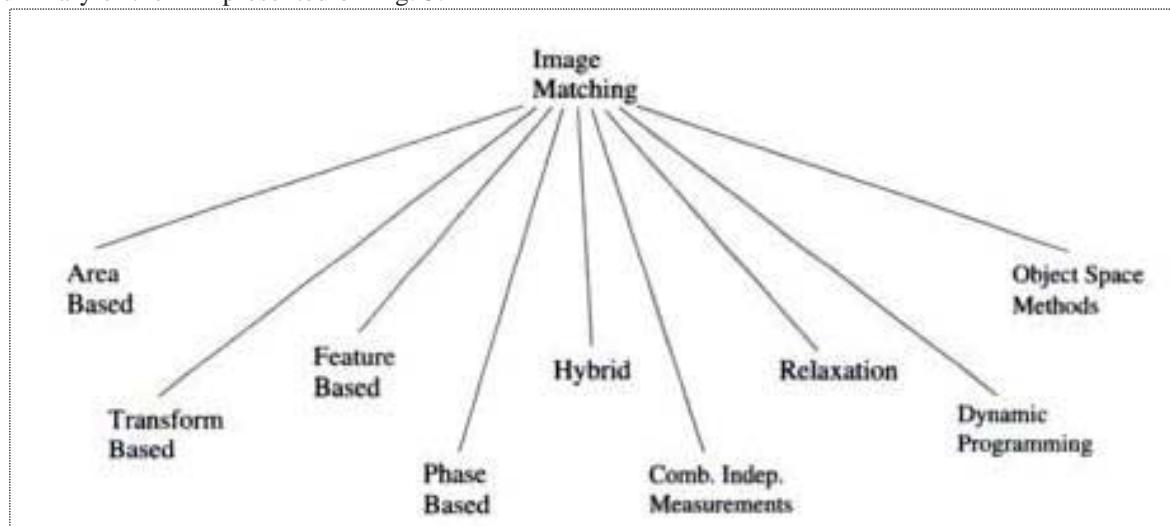


Figure 3.1 – Taxonomy of image matching techniques.

The most basic algorithm for image matching uses SAD – Sum of Absolute Differences. Formula (3.1) gives the relation used for the matching coefficient for this algorithm. In the formula the intensity of a pixel at position (i,j) in the template is given with $I_1(i,j)$ and with $I_1(x+i, y+j)$ – the intensity of the corresponding pixel in the screenshot. Then a sum of all differences is calculated. The lower the sum of all differences is, the better the match. The algorithm returns zero if the images match completely. [1]

$$\sum_{(i,j) \in W} |I_1(i,j) - I_2(x + i, y + j)| \quad (3.1)$$

Although this algorithm is good enough to solve the most basic task, it should be optimized. This is due to the fact that iteration with a step of one pixel is time consuming. Optimization would be achieved if the step is larger before finding a relatively good match thus localizing the matching in an area where it is more likely to occur. This algorithm is not returning good results when the luminosity and color are different in the two images. These are subjects of the next two tasks. Nevertheless this algorithm could be combined with edge detection algorithms to achieve better results [6].

We have also executed experiments with the SSD area based algorithm. The correlation coefficient is calculated by the formula (3.2), where again the intensity of a pixel at position (i,j) in the template is given with $I_1(i,j)$ and with $I_1(x+i, y+j)$ – the intensity of the corresponding pixel in the screenshot.

$$\sum_{(i,j) \in W} (I_1(i,j) - I_2(x + i, y + j))^2 \quad (3.2)$$

Due to the need to calculate with multiplication this algorithm is more time-consuming than SAD. [2] More accurate but again more time-consuming than SAD is the NCC – Normalized Cross Correlation algorithm. The correlation coefficient is given with formula (3.3) where $t(x,y)$ is function of the template, $f(x,y)$ is function of the screenshot, n – number of pixels in the template, \bar{f} average value of f and σ_f is the standart deviation of f .

$$\frac{1}{n} \sum_{x,y} \frac{(f(x,y) - \bar{f})(t(x,y) - \bar{t})}{\sigma_f \sigma_t} \quad (3.3)$$

In functional analysis this is a scalar product of two vectors. NCC is the cosine of two angles between the normal vectors. This method is widely used in the template matching techniques [7]. However, it is not part of our research due to the fact we are searching for the most lightweight solution and this algorithm does not perform better than SAD.

4 Automatic user interface navigation system

4.1 Brute-force pattern matching

For solving the basic task we have used the most common way of iteration through an array – two-level for loop iterating the x and y axes of the screenshot. At each step we crop an image with the size of the template. We compare it with the template, forming the sum of differences of all the odd pixels. If the match is good enough we go for the even pixels. If the match is still good enough, which is given by experimentally deducted coefficients, the coordinates of the starting pixel of the cropped image are returned as a result of the program. For clarity the algorithm is described with the following UML diagram in Figure 4.1:

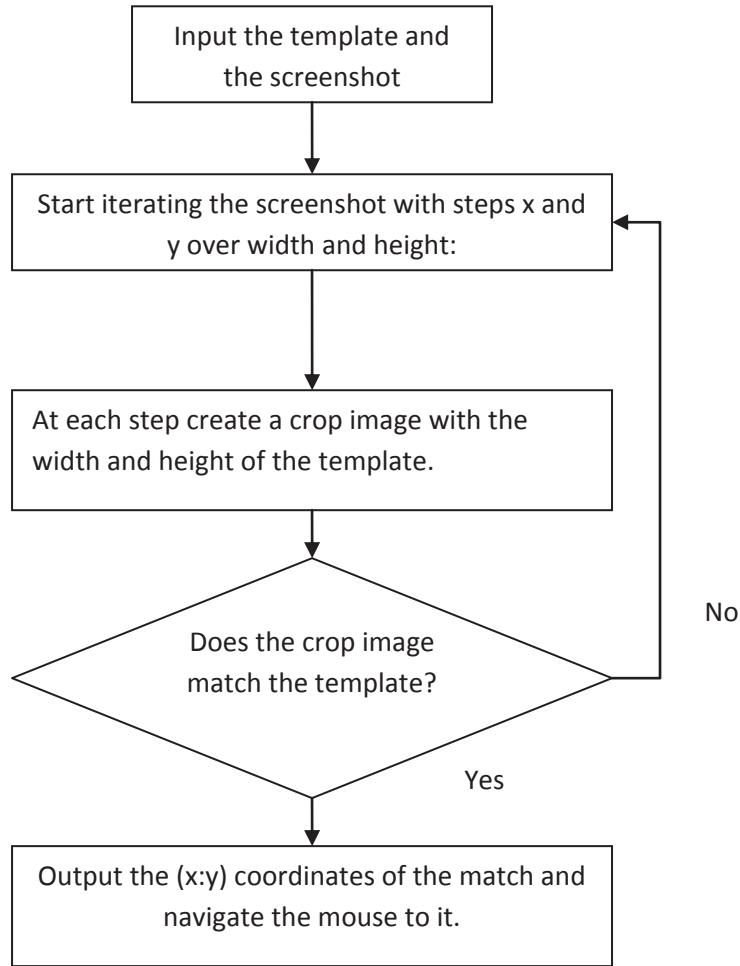


Figure 4.1 UML diagram describing the template matching algorithm

The method which is performing the template matching is Point findMatch(). This is the method present in OpenCV and which we had to rewrite from scratch. The implementation is given in the following code listing:

```

private Point findMatch(BufferedImage template, BufferedImage screenshot)
throws IOException {

    int[] templateArray = template.getRGB(0, 0, template.getWidth(),
                                         template.getHeight(), null, 0, template.getWidth());
    this.templateHeight = template.getHeight();
    this.templateWidth = template.getWidth();

    // slide through screenshot array with windows as big as templateArray
    for (int j = 0; j < screenshot.getHeight() - template.getHeight(); j++) {
        for (int i = 0; i < screenshot.getWidth() - template.getWidth(); i++) {
            int[] screenshotArray = new int[this.templateWidth*this.templateHeight];
            screenshot.getRGB(i, j, this.templateWidth, this.templateHeight, screenshotArray, 0,this.templateWidth);
            int[] difference = new int[this.templateWidth*this.templateHeight];
            int allDifferences, matchedPixels = 0;
            for (int k = 0; k < templateArray.length; k += 2) {
                difference[k] = getPixelARGBDifference(screenshotArray[k], templateArray[k]);
                allDifferences += Math.abs(difference[k]);
                if (screenshotArray[k] == templateArray[k] || difference[k] <= DIFFERENCES) {
                    matchedPixels++;
                }
            }
        }
    }
}

```

```

        }
    }

int coeff = DIFFERENCES * this.templateHeight * this.templateWidth;

if (Math.abs(allDifferences) < coeff / 2 && matchedPixels > 50) {
    System.out.println("Half matched");
for (int k = 1; k < templateArray.length; k += 2) {
difference[k] = getPixelARGBDifference( screenshotArray[k], templateArray[k]);
allDifferences += Math.abs(difference[k]);
if (screenshotArray[k] == templateArray[k] // difference[k] <= DIFFERENCES) {
    matchedPixels++;
}
}
}

if (Math.abs(allDifferences) < coeff && matchedPixels > 100) {
    System.out.println("Matched!");
    return new Point(i, j);
}
}
return null;
}

```

The results show that the time for matching depends very much on the position of the template in the screenshot – the further it is from the top left corner the more iterations it takes to find it and thus more time. The next factor to determine time cost is the size of the template. Results for time cost are presented in table 4.1 in 4. Experimental results section.

We have implemented also the SSD algorithm. For that purpose we have created the getPixelARGBDifferenceSquared(int pixel1, int pixel2) method. Comparative results are presented in table 4.2 in section 4. Experimental results .

```

public int getPixelARGBDifferenceSquared(int pixel1, int pixel2) {
double redDiff = Math.pow(((pixel1 >> 16) & 0xff) - ((pixel2 >> 16) & 0xff), 2.0);
double greenDiff = Math.pow(((pixel1 >> 8) & 0xff) - ((pixel2 >> 8) & 0xff), 2.0);
double blueDiff = Math.pow((pixel1 & 0xff) - (pixel2 & 0xff), 2.0);
return (int) (Math.sqrt(redDiff) + Math.sqrt(greenDiff) + Math.sqrt(blueDiff));
}

```

Experiments show that SSD algorithm is performing slower than SAD algorithm without gaining more precise results. Thus the research will continue with the usage of SAD algorithms.

Performance of this basic implementation is not good enough – we can only use it to find templates which are in the top of the screen captured. The basic solution needs optimisation which will be performed in the following directions:

1. Instead of checking the odd pixels and then the even, we will lower the quantity of pixels checked for a preliminary decision. Our experiments have shown that 1/10 of all the pixels are enough.
2. Extend the step from one pixel we will iterate with bigger steps and the steps will differ by x and y.

The implementation of these optimisations is explained in the next section.

4.2 Optimization with more precise delimiters

In order to achieve optimization we have only one loop, where the iteration step is changing based on the number of the iteration – starting with 10, then 5, 3, 2, 1. Results from these changes are shown in table 4.3 in section 4 Experimental results.

Next optimization – changing of the iteration step, needed also a change in the delimiter values which determine whether the template matches, because otherwise with increasing the step and not making the delimiters tolerant enough the algorithm stops working. The value of the step is very important. Our

experiments show that the y step could be bigger than the x step. In order to achieve this two new constants have been added. These could be later externalized as parameters of the function.

When the step is bigger than one pixel it's very often so that the crop we found was not aligned with the template. We could never expect a full match – thus we had to implement partial match functionality. This means adding an additional set of constants to determine partial match. After partial match is found we continue searching in the suspected area with a step of one pixel to reach a full match. False partial matches could lead to more time wasted on searching with lower incrementing step, so setting the right constant delimiters is very important and this is the focus of this research.

Here is an example of the constant delimiter depending on the iteration value. It takes in consideration the worst case - that this could be partial match so we expect that there are at least that count of pixels that match:

```
int stepWiseMatchesCount = templateHeight * this.templateWidth - stepX * this.templateHeight -
stepY * this.templateWidth;
```

Based on the pixels that we expect to match we determine the minimum sum of differences these pixels could have in order to consider the result a match. This is given with *differenceTolerance***stepWiseMatchesCount*:

```
int coeffStepWise = differenceTolerance* stepWiseMatchesCount+ 255*3*(stepX * this.templateWidth + stepY*
this.templateHeight);
```

In the second part of the expression we considered the differences that are expected to be present if the crop and the template are not aligned. In the worst case that is: (255 values for each colour) * 3 (for R G B) * (number of pixels that will differ based on the size of the step). Experiments with that many differences were leading to wrong results. More precise equation is proposed. The result is the following:

```
(allDifferences/templateArray.length)*(2*templateArray.length/( stepX * this.templateWidth + stepY*
this.templateHeight)) * (stepX * this.templateWidth + stepY* this.templateHeight);
```

Instead of using 255 we get the average of all differences considering the size of the template. In the next part instead of 3 we are using the ratio between the number of pixels in the template and the number of pixels that will be different in the worst case. The optimisation is presented with the results from table 4.4 in section 4 Experimental results.

4.3 Scaling and colour invariance problem solving

In order to solve the task with scaled images the template is scaled with the needed coefficient and then the iterative check is performed with each scaled image. This means that the already created functions are reused but we call them with the scaled template as an argument instead. Performance will significantly drop if we simply do these executions one after another. We used several threads instead – each one searching for a different scaling coefficient. The first thread that found a match is stopping the execution of the algorithm. We have conducted experiments with a finite number of scaling coefficients. A 2013' research shows that the most popular screen resolutions are 1920/1080, 1366/768, 1280/800, 1024/768. If needed one could add a resolution and then a new thread will execute searching with it.

To fulfil the requirement for color invariance we also started checking for intensity. The intensity check is given by the formula (4.1).

$$I=(R+G+B)/3 \quad (4.1)$$

We could get too much difference with scaled images due to interpolation. In our experiments we have used the Java API to create the scaled images using a *RenderingHints.VALUE_INTERPOLATION_BICUBIC* which corresponds to maximum quality. Nevertheless the scaling is digital, which guarantees loss of information and this is affecting the correctness of our results. Our experiments showed better results when using the intensity check, so we adjusted the delimiters with it. Results from the execution are presented in figure 5.5 in section 5. Experimental results.

5 Experimental Results

The following figures – Figure 4.1 and Figure 4.2 have been used to check the performance of the applications. They differ with the position of the template – in the first one it is further away from the top left corner.



Figure 5.1 First test screenshot image with enlarged template

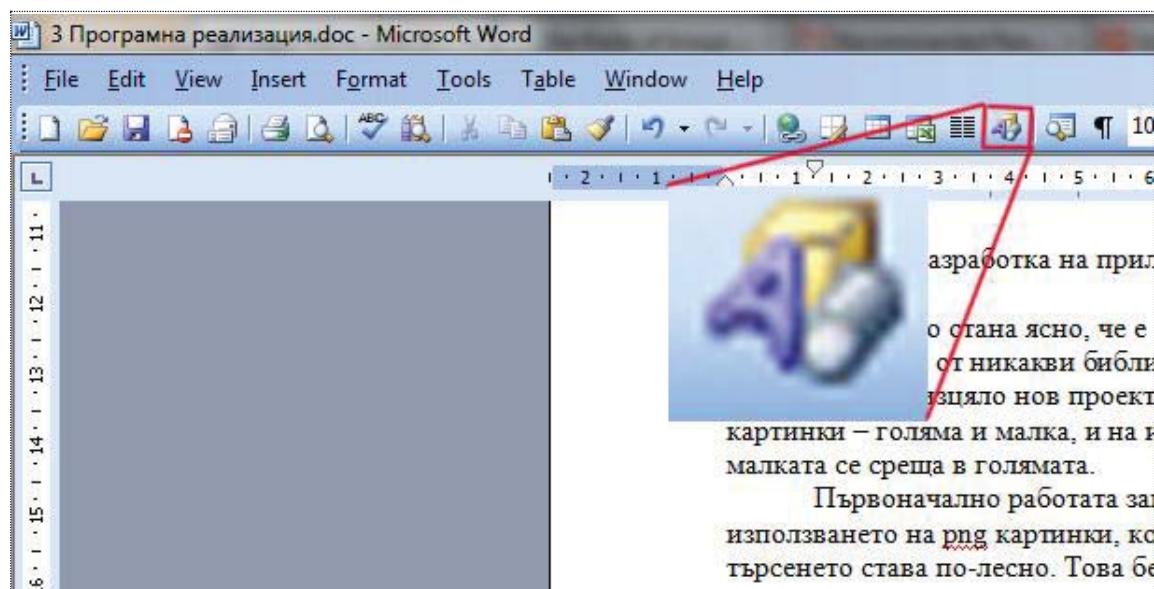


Figure 5.2 Second test screenshot with enlarged template.

Experiments are conducted the following way – the application is called 10 times with one and the same template and screenshot. The first series are with the images from Figure 5.1, second series are with the image from Figure 5.2. Table 5.1 shows the experiments with the basic solution with step 1 pixel and check for 50% of the pixels at the first iteration. Execution times are not satisfactory.

Test image number	Execution time [s]										AVE
Test image from Figure 5.1	7 6 6 5 6 6 6 6 5 5										5.8

Test image from Figure 5.2	23	41	44	35	29	41	45	32	45	37	37.2
----------------------------	----	----	----	----	----	----	----	----	----	----	------

Table 5.1 Experimental results with the usage of SAD algorithm

Experimental data from table 5.2 shows that the SSD algorithm performs slower than the SAD algorithm. We have 17% slower execution for Figure 5.1 and 12% slower execution for Figure 5.2

Test image number	Execution time [s]										AVE
Test image from Figure 5.1	7	7	7	7	8	6	7	8	6	7	7
Test image from Figure 5.2	39	48	56	50	50	44	33	30	38	38	42.6

Table 5.2 Experimental results with the usage of SSD algorithm

Experimental data in table 5.3 which refers to section 3.2 for optimization with changing the number of pixels checked at first iteration is showing better performance. We have 19% decrease in time for Figure 5.2 and 26% decrease in time for Figure 5.1.

Test image number	Execution time [s]											AVE
Test image from Figure 5.1	4	5	4	4	4	4	4	4	4	5	5	4.3
Test image from Figure 5.2	32	20	34	35	30	28	29	33	29	32	30.2	

Table 5.3 Experiments with optimization of the number of compared pixels on first iteration

Experimental data from table 5.4 which refers to section 4.2 optimisation with enlarging the iterations step is showing satisfactory results. Further reduce of time could be achieved with better mathematical expressions for the constant delimiters. These experiments are executed with steps $x = 4$ pixels and $y = 7$ pixels.

Test image number	Execution time [s]											AVE
Test image from figure 4.1	3	3	3	3	3	3	3	3	3	3	3	3
Test image from figure 4.2	3	4	4	4	3	3	4	3	3	3	3	3.4

Table 5.4 – Experiments with enlarging the iteration step

Figure 5.5 shows the advantage of parallel executions – the first thread finds the desired template and there is no need to search further with the other threads and loose performance.

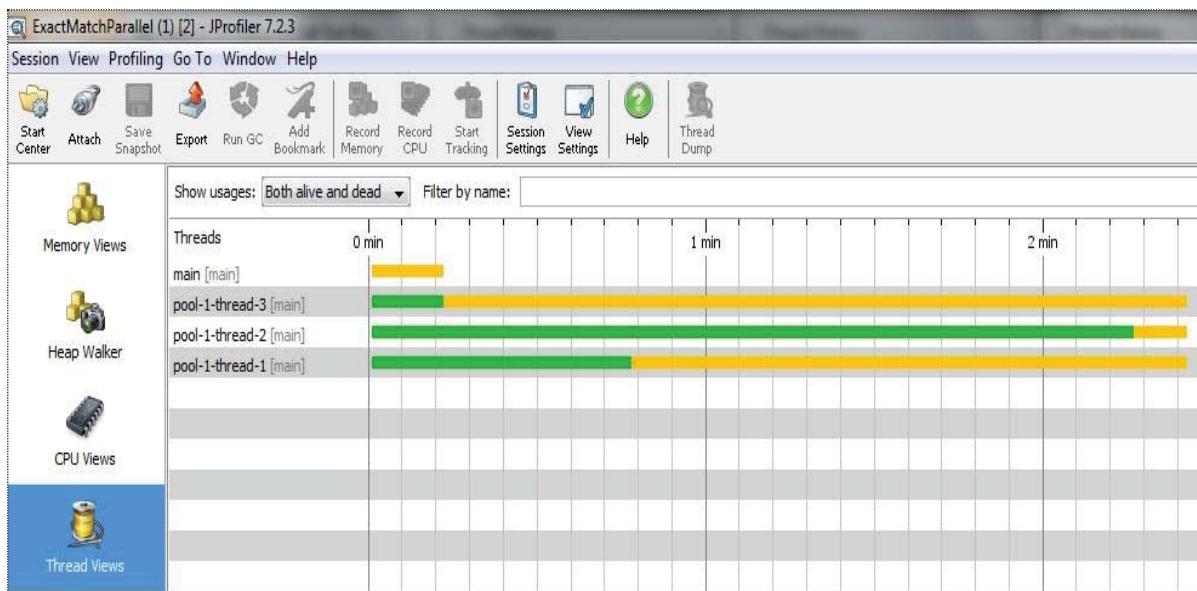


Figure 5.5. Performance look up for parallel execution with several threads with JProfiler

Figure 5.6 shows the console output of the execution with three threads. While steps are big the algorithm was in the partial match section, later when a partial match was found the steps are decreased to 1 pixel to iterate the region and find a precise match:

```

Execution with diffToleranceColor: 6 diffToleranceIntensity: 4 and stepX: 2 and stepY: 7
Execution with diffToleranceColor: 6 diffToleranceIntensity: 4 and stepX: 2 and stepY: 7
Execution with diffToleranceColor: 6 diffToleranceIntensity: 4 and stepX: 2 and stepY: 7
I, J : 1242 70
Execution with diffToleranceColor: 6 diffToleranceIntensity: 4 and stepX: 1 and stepY: 1
I, J : 1244 77
Execution with diffToleranceColor: 6 diffToleranceIntensity: 4 and stepX: 1 and stepY: 1
I, J : 295 7
Precise match false Color match false IntensityMatch true
java.awt.Point[x=1539,y=84]
Thread 3: java.awt.Point[x=1539,y=84]
Match found for: 0 min, 12 sec

```

Figure 5.6 Console output of parallel searching with three threads

6 Conclusions

The problem for fast auto-navigation with a subtask for finding a template in a screenshot has been solved in a platform invariant manner. The research suggested an implementation using no external libraries, optimised with constraints based on the size of the template image for better performance and intensity check for colour invariance. Parallel searching was introduced to check for different resolutions by scaling the template with proper scale factors. The proposed solution could be further developed to support enhanced color invariance reliability by implementing better algorithms for template matching based on edge detection. Further optimization can also be achieved by more detailed research on the constant delimiters – it would require deeper understanding in the mathematical correlation between sizes, probability for a match and step-wise iterations.

References

- [1] M. Bennamoun, G.J. Mamic Shin-His Chang, *Object Recognition: Fundamentals and Case Studies: , Fast Algorithm For Point Pattern Matching Invariant to Translations, Rotations and Scale Changes*, Springer 2002.
- [2] S. Ahuja, *Correlation based similarity*, <http://siddhantahuja.wordpress.com/tag/sum-of-squared-differences/>
- [3] Sikuli Java API Google project, <http://code.google.com/p/sikuli-api/>
- [4] Java OpenCV MatchTemplate,
http://www.larmor.com/portal/index.php?option=com_content&task=view&id=27&Itemid=60
- [5] AutoIT, <http://www.autoitscript.com/site/autoit/>
- [6] Sum of absolute differences, http://en.wikipedia.org/wiki/Sum_of_absolute_differences
- [7] http://en.wikipedia.org/wiki/Cross-correlation#Normalized_cross-correlation
- [8] Java programming language, http://en.wikipedia.org/wiki/Java_%28programming_language%29

Kristina PETKOVA
Technical University Sofia
Faculty of Industrial Technology
bul.Kliment Ohridski 10
Bulgaria
E-mail: kristinapetkova.tu@gmail.com

Asoc. Prof PhD Milena LAZAROVA
Technical University Sofia
Faculty of Computer Science
bul. Kliment Ohridski 10
Bulgaria
E-mail: milaz@tu-sofia.bg

Assoc. Prof Phd Hristo ALADJOV
Bulgarian Academy of Sciences
IBPhBME
Akad. Georgi Bonchev 105
Bulgaria
E-mail: aladjov@biomed.bas.bg

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Web Content Management System using ASP .NET, C#, SQL Server and Windows Azure

**Petrescu Octavian Florin
Teacher Coordinator: Maria Miroiu**

Abstract

This project is a Web Content Management System (WCMS) developed on ASP .Net Platform using C# programming language. The project was realized in a desire to implement a platform for creating web pages that is easy to use even for users without knowledge in the domain. Projects that belong to the same range are Umbraco, Sitefinity, Kentico etc. The project consists of several parts: Business, DomainModel, DAL, enums and WCMS, and to store the content of the pages is used a Microsoft SQL database. This platform differs from other WCMS sites by using user controls and the control DynamicControlsPlaceHolder. This creates a hierarchy of persistent assets contained in a page, that don't need to be recreated after postback. Another difference is the fact that it is hosted using the Windows Azure.

1 Introduction

A WCMS is a software tool used both by technical people and people who do not have technical knowledge to create a structure of web pages. It manages small units of information (web pages) that are interconnected by a navigation structure or path. Each web page is defined by its location in the site, all pages are combined in a hierarchy created based on the positioning of pages in a tree. Attention is directed primarily on creating pages and managing their content. It provides the ability to publish only certain pages, starting with a specific date.

2 Project Structure

The application consists of 5 parts: DAL, Business, DomainModel, WCMS and Enums, each part fulfills a different role in the platform. It is created using ASP .NET Web Forms. The administration part is created using a Master Page that displays a navigation bar with sections and subsections. They offer the ability to manage sliders, news, events etc. The pages are displayed in a TreeView control, a page may have subpages.

2.1 Data Access Layer

Accessing the database is done using Entity Framework and the EDMX is included in DAL (Data Access Layer). There are implemented the Repository and Unit of Work patterns to perform CRUD operations (create, read, update, delete). These patterns are designed to create an abstraction between the data access and business side of the application. The unit of work class coordinates the work of multiple repositories by creating a single database context class shared by all of them.

2.2 Domain Model

In Domain Model there are a number of filters that are sent as parameters in a method which aims to manipulate database. Filters contain parameters that are applied to queries. A filter should extends the SingleIdFilter abstract class . If a service doesn't have a specific filter, it uses BasicFilter class.

2.3 Business

The business part contains services that interact with database. Each service extends the abstract class SingleIdService specifying the entity that is used by the service, ID's data type and a filter. Service must implement the abstract methods: GetUnitOfWork, GetIdComparer and GetByFilter. The SingleIdService class has already implemented methods for adding, updating or deleting data from a table. To extract data from tables after certain criteria the filters created in DomainModel are used.

2.4 Enums

In enums are included several enumerable used in order to better organize the code, for example, the types of assets. Some of the entities, for example the Asset, has one of the fields converted to enum.

```
public enum AssetTypes
{
    [Description("Text Asset")]
    TextAsset = 1,
    [Description("Slider")]
    Slider = 2,
    [Description("News")]
    Slider = 3,
    [Description("Events")]
    Slider = 4,
    [Description("GoogleMap")]
    Slider = 5,
}
```

2.5 WCMS

In WCMS is located the mechanism that is managing all pages and their content. Every page should have a template. Template is created, in turn based on a physical template, which is a user control that holds html/css content and one or more dynamicdivs. A dynamicdiv is, in turn, a user control that has Pageid, ZoneID and AdminMod and contains a DynamicControlsPlaceHolder. DynamicControlsPlaceHolder is a control created by Denis Bauer and has the ability to recreate its children after postback [4]. An asset is a control that can be added, edited and deleted in the admin. The application provides a list of assets like: TextAsset, Slider, News, Events, Google Map, Facebook feeds, Twitter etc.

Workflow to create a page is:

- ❖ First time a template is created, if one does not already exist or if it hasn't the structure that we want. It receives a name and a physical template whose structure it wears. In a template we can insert meta tags or any other html code in the head or body, which is inherited by all subpages created on its basis. You can set a page to be added to the cache and a date when this cache is about to expire.
- ❖ It creates a page based on a template. To a page and we can set a name, a URL, a name under which to appear in the menu, if it is visible or not, a URL to be redirected at every request, a date and a title. In addition there are available SEO type meta tags as: Noindex,nofollow and Description.
- ❖ It injects the assets. Some of the assets can be customized, for example, the slider asset can take a specific size (height and length) and a specific slider to display, created in admin page.

To determine the page that is requested, the routing mechanism of ASP. Net platform is used. The name of the page is unique in the site and depending on it the assets are extracted from database and are injected to editable areas from the template that the page is based on.

The platform also benefits from a number of grids in admin, which display the records of forms from the site.

The News asset type can be displayed in several forms: full, summary, pics list and list. In each form, it will display the latest news and their number will be set in the asset's wizard.

Event asset type has the same properties as the news, but it will also be displayed on a map of events (Event Map).

Facebook and Twitter Feeds display latest posts on a particular profile, the number of posts and profile being inserted by the administrator.

Google Map is an asset that displays the map using the API of the same name. It can adjust the height and length.

TextAsset is a control that renders html code. This code is entered using the html editor from Ajax Control Toolkit [3].

The administrator has at his disposal a number of pages that can create sliders, news, events etc, which will later be displayed in pages.

The platform is hosted on a virtual machine using the Windows Azure. The virtual machine uses Windows Server 2012 as the operating system. The IDE that was used to create the project was Visual Studio 2013, and it is configured to migrate the site to the virtual machine.

3 Conclusions

The project is different from other works because the assets are persistent after postback due to DynamicControlsPlaceHolderelor, the data base will be accessed only once to create the page. Another difference is the fact that the user can set if the page is cached and when it is invalidated. In contrast to other platforms of the same gendre that uniquely identifies the pages through an id, this WCMS uses the page name as a unique identifier .

4 Further Work

As future improvements several new asset types will be implemented: Image Gallery, QR Code, SiteMap, Menu etc. It will also be created a shopping cart and it is aimed to change the platform to provide the ability to create not only pages, but entire sites and the assets to have a different design at each site. Another improvement will be a mechanism to edit the email templates from admin. For a better design, Bootstrap 3.0 will be used to create templates and controls. Also, will be implemented a system to upload files in platform from admin to be used in pages.

5 References

- [1] Bill Evjen, Scott Hanselman, Farhan Muhammad, Srinivasa Sivakumar, Devin Rader, *Professional ASP .NET 2.0*
- [2] Bill Evjen, Scott Hanselman, Devin Rader, Professional ASP .NET 4 in C# and VB
- [3] AJAX Control Toolkit <http://www.ajaxcontroltoolkit.com/>
- [4] Denis Bauer , DynamicControlsPlaceHolder - A place holder that saves its child controls <http://www.denisbauer.com/Home/DynamicControlsPlaceholder>
- [5] The Official Microsoft ASP .Net Site <http://www.asp.net/>

Octavian Florin PETRESCU
University of Pitesti
Department of Mathematics and Informatics
Str. Targul din Vale nr. 1, Pitesti, Arges
ROMANIA
E-mail: petrescu.octav@gmail.com

Real-time shape recognition using contours and Hough transform

Radmila Sekulic, Nebojsa T. Peric, Milos Jordanski
Teacher Coordinator: Milan Tuba

Abstract

In this paper an approach to detecting two-dimensional shapes is proposed. Firstly, different types of edge operator detectors which minimize both noise errors between the input and the filtered output are derived. Secondly, an investigation into how an error propagates in relation to shape geometry. This enables to predict localization and detection of contour tracking in video. In many cases, boundary shape information, such as rectangular shapes of vehicles, seems to play crucial roll. This paper proposes shape recognition method where circle, rectangle and triangle object in the image will be recognizable.

Keywords: edge detection, geometrical shapes, image processing, real-time, shape recognition

1 Introduction

The term shape is commonly used to refer to the form of an object or its external boundary, as opposed to other properties such as colour, texture, material composition. Shape in an image is one of the key information that the human eye receives when it recognizes an object. In human's perception, recognition and understanding the shape of an object is important factor. Shape of an image does not change when colour of image is changed. Identifying 2-D shapes has been a classical problem in computer vision. The "shapes" here refer to 2-D image features of an object which are invariant to scene factors. In a computer system shape of an object can be interpreted as a region encircled by an outline of the object[3]. The important job in shape detection is to find and represent shape information. Shape recognition finds its applications in robotics, fingerprint analysis, face recognition, etc.

Many methods for 2D shape representation have been reported: Curvature scale space (CSS), dynamic programming, shape context, Fourier descriptor and wavelet descriptor. There are two methods for shape recognition: area based and boundary based technique[9]. In area based technique, all pixels within the region of image are taken into consideration to get shape representation.

Common area based technique uses moment descriptors to depict the shape. Whereas boundary based technique focuses mainly on object boundary[10].

Boundary based technique represents shape feature of object more clearly as compared to area based technique. It is fast in processing and needs less computation than area based technique[11]. Due to fast processing and easy computation it is widely used in real-time and practical applications. In this paper we consider just the boundary based technique.

2 Detection

For many years, computer vision researches have been trying to design good edge detectors, since edge detection is an essential step in many computer vision applications. This process can be complicated when the relevant object is complicated or blurred. Edge detection is often used as preprocessing to shape recognition. Shape recognition groups a set of objects in several clusters based on their similarity. A successful shape recognition approach requires a robust shape descriptor, representing the measure that captures shape features and in a concise manner and is invariant to all geometric transformations.

2.1 Edge detection

The purpose of edge detection is to localize variations such as discontinuities, local extrema and 2D features where at least two edges meet. There are many ways to perform edge detection. However, the most may be grouped into two categories: gradient and Laplacian[4].

The gradient method detects edges by looking for the maximum and minimum in the first derivative of the image. The Laplacian method searches for zero crossings in the second derivative of an image to find the edges.

However, differentiation of an image is an ill-posed problem; image derivatives are sensitive to various sources of noise and quantification effects. An edge detector accepts discrete, digitalized images as input and produces an edge map as output. In addition, we will propose several types of edge detectors[8].

2.1.1 Laplacian edge detector

Laplacian operator is second derivative operator used in edge detection[5]. The Laplacian of an image highlights regions of rapid intensity change.

The Laplacian $L(x,y)$ of an image $I(x,y)$ is given by:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2}(x,y) + \frac{\partial^2 I}{\partial y^2}(x,y)$$

Two commonly used kernels are :

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

2.1.3 Roberts Cross operator

The Roberts Cross operator is one of first edge detectors [5]. In its most common usage, the input to the operator is a grey scale image, as is the output. As differential operator, the idea behind the Roberts cross operator is to approximate the gradient of an image through discrete differentiation which is achieved by computing sum of the squares of differences between diagonally adjacent pixels.

Roberts proposed following equations:

$$y_{i,j} = \sqrt{x_{i,j}}$$

$$z_{i,j} = \sqrt{(y_{i,j} - y_{i+1,j+1})^2 + (y_{i+1,j} - y_{i,j+1})^2}$$

where x_{ij} is intensity of an image. The result of an operation will highlight changes in intensity in diagonal direction. One of most appealing aspects is its simplicity; the kernel is small and contains only integers, but this filter suffers greatly from sensitivity noise.

2.1.5 Sobel operator

The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial gradient that correspond to edges[5]. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grey scale image.

The operator consists of pair 3x3 convolution kernels as shown. One kernel is simply the other rotated by 90°.

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation.

2.1.4 Prewitt operator

Prewitt Operator is another operator that can be used for edge detection[5]. However Prewitt Operator is more inclined towards finding the horizontal and vertical edges and then using both of these intermediate result to generate a final image. The Masks used for this operator are as below.

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	-1

2.1.6 Canny filter

The Canny edge detection algorithm is known as optimal edge detector. There is a list of criteria to improve current methods of edge detection. The first is low error rate, it is important that edges occurred in image should not be missed. Second is that edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum[5]. A third criterion is to have only one response to a single edge. This was implemented because the first two were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smooths the image to eliminate noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (non-maximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the two thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above second threshold.

2.2 Shape recognition

Shape recognition represents a procedure employed to extract information from acquired images[12]. In this paper we will propose shape recognition using Canny detector, as optimal detector of edges. Before applying Canny filter we must do several steps:

1. Smooth image with an appropriate Gaussian filter to reduce desired image details
2. Determine gradient magnitude and gradient direction at each pixel

3. If the gradient magnitude at a pixel is larger than those at its two neighbours in the gradient direction, mark the pixel as an edge. Otherwise, mark the pixel as the background.
4. Remove the weak edges by hysteresis threshold.

Here we will focus on object detection by grouping the edge fragments in image according to contour model. From the extracted edging image, we use the contour algorithm to detect closed boundaries of objects. These contour lines are transformed to Hough coordinate, and then we recognize objects such as circles, triangles and rectangles.

2.2.1 The Hough Transform

The Hough Transform was first devised in 1962 as a means of detecting the paths of high energy particles[1]. It has since evolved and has been applied to many different image processing applications. The Hough Transform technique works by transforming complex patterns of pixels in the image domain into compact features in a chosen parameter space. The transformation operates such that many points in the image space map to single points in the parameter space. This means that searching for complex patterns of pixels is simplified by working in the parameter space and that the technique is robust to some loss of data due to occlusion and other sources of noise.

2.2.2 The straight line Hough transform

Any line in an image can be described by the slope-intercept equation and appears as a single point in $(m; c)$ space:

$$y = m \cdot x + c$$

By transforming from image space to $(m; c)$ space the problem of line detection is simplified to a problem of point detection[2]. In practise the transformation has to be applied to each individual point in the image which transform into lines in the parameter space, however, collinear points in the image map to intersecting lines in the parameter space points. If votes are accumulated for points plotted in the parameter space then the intersection will be seen as a point of local maxima. It follows that lines in the image are found by searching for points of local maxima in the parameter space.

A modification has to be made to this technique to cope with vertical or near vertical lines for which the size of the parameter space becomes infinite. This is achieved by using two parameter spaces, one in $(m; c)$ space and the other in $(m'; c')$ space, where:

$$m' = \frac{1}{m}$$

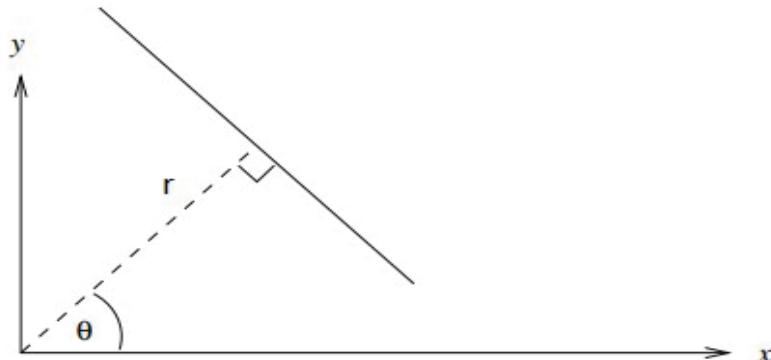
$$x = m' \cdot y + c$$

Peaks in the first space represent lines with gradients less than one. Peaks in the second space represent lines with gradients greater than or equal to one. The main problem with this technique is the large amount of computation required. This arises because for every point in the image a whole line has to be plotted in two parameter spaces and then these spaces need to be searched for points of local maxima.

The technique was improved by Duda et al in 1972 by replacing the slope-intercept equation by the normal equation:

$$r = x \cdot \cos(\Theta) + y \cdot \sin(\Theta)$$

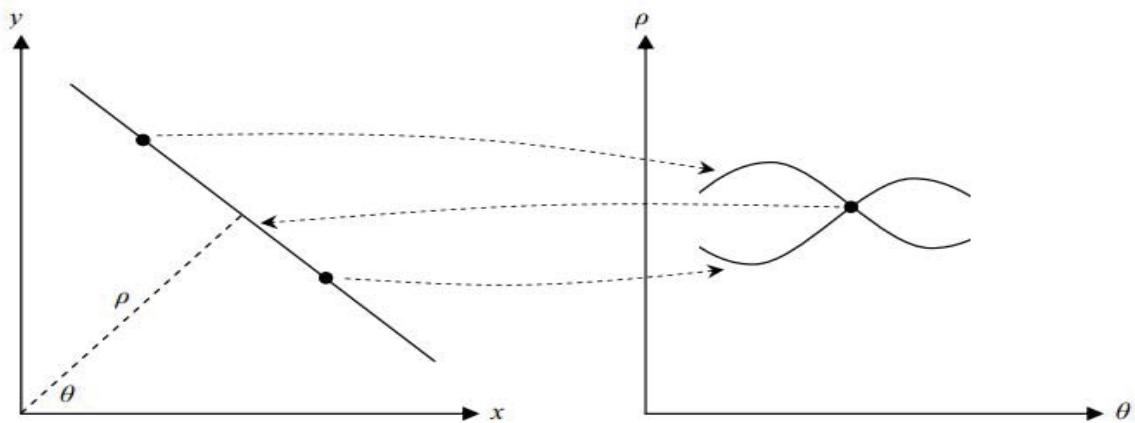
Where r is the length of the normal from the line being detected to the origin and Θ is the angle between the normal and the positive x-axis.



Line in image space defined via r and θ

A point in image space now appears as a sinusoid in $(r; \theta)$ space and collinear points in the image appear as intersecting sinusoids. The advantage of this method is that only one parameter space is required so the storage and computational requirement is reduced.

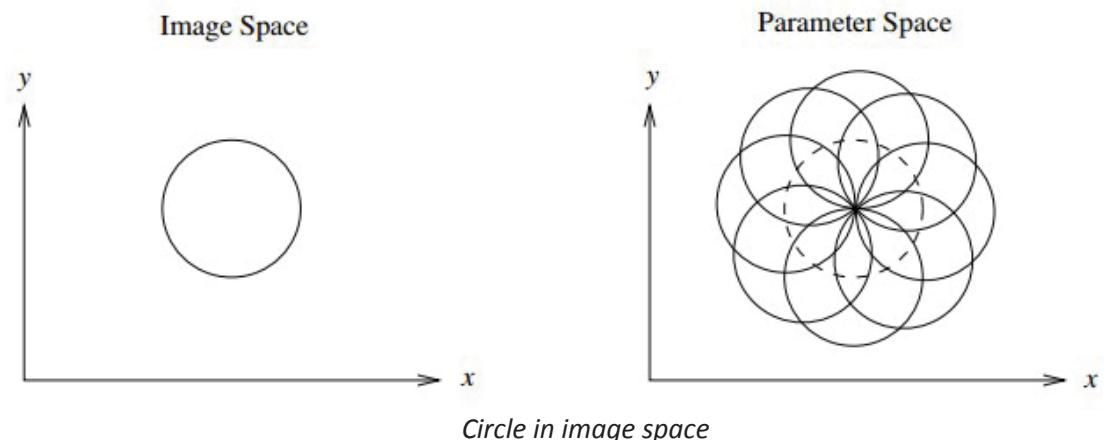
A line in the (x, y) space is represented by the point of intersection of sine curves, in the characteristic “butterfly” pattern and the line detection consists of finding accumulation points of curves.



Further improvements have been made to reduce the computational load. Rather than transforming every point in image space into a sinusoid in parameter space, the image can be first searched for short line segments and these used instead. Because the orientation of the segments can be determined only a single point for each has to be plotted in parameter space. Initially a histogram of line segment orientations is built and significant peaks identified. The line segments associated with these peaks are then plotted in r space and peaks in this space correspond to lines in the images.

2.2.3 Circle detection using Hough transform

The Hough Transform can be used to detect circles within an image[6]. In this scheme the parameter space is congruent with the image space, that is, each point in the image maps to a point in the same position in the parameter space. To detect a circle of radius R circles of this radius are plotted in parameter space centred on edge segments found in the image. Peaks in the parameter space indicate the centre of detected circles.

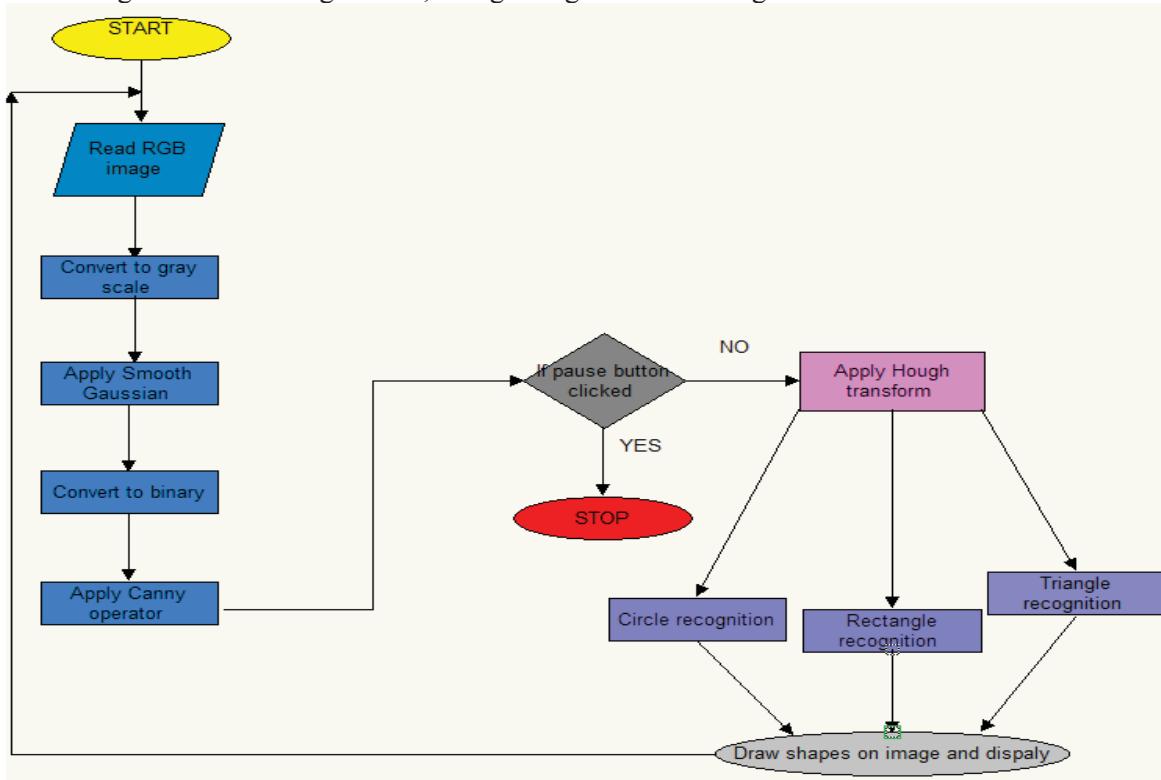


This technique is particularly computationally intensive as a whole circle has to be plotted in parameter space for every edge found in image space. A significant saving is made by reducing parameter space for circle detection. It consists of line-transforming each point on curved, using normal-form parametrization. This transforms from a family of curves in parameter space and the envelope of family represent all tangents to original shape in image[7]. Second transform detects flat stretches on the enveloping curve; from these parameters of original shape can be derived.

3 The scope of application

Our application reads RGB image in from camera and converts coloured image to a gray scale image. Afterwards smooth Gaussian filter is applied to blur the image in order to reduce image details. Clearly, we have a fundamental trade off here between the loss of information and noise reduction.

The ultimate goal is to find optimal detectors that ensure a favourable compromise between noise reduction and edge conservation. We opted for Gaussian because it is invariant to rotation. Then this output is converted to binary image. Also, the binary image is inverted in order to speed up time of image processing. In the next step, the Canny operator is applied to extract edges. We use this filter, because it shows the best results in terms of performance, velocity and extracting. Once the edges have been extracted we are able to perform shape recognition. Circles, triangles and rectangles will be recognizable, using Hough transform algorithm.



Flowchart of the application

3.1 Technology used

The application was developed using C# programming language 4.0 on .NET platform. C# is an OOP language which appeared in 2000, used by software developers for building .NET applications.

4 Conclusion

This paper introduced the concept of Hough Transform and edge operators, and how they are used in object detection and shape recognition in real-time application. The Hough Transform in general has several limitations making it challenging to detect anything other than lines and circles. This is especially the case when more parameters are needed to describe shapes, this add more complexity thus decreasing the votes within the accumulator.

References

- [1] Duda, R.O., Hart, P.E., "Use of the Hough Transform to detect lines and curves in pictures," Comm. ACM 15, pp. 11-15, 1972.
- [2] V.F. Leavers, "Shape Detection in Computer Vision Using the Hough Transform," Springer 1992.
- [3] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Third Edition, 2008
- [4] M. Heath, S. Sarkar, T. Sanocki, and K.W. Bowyer. "Comparison of Edge Detectors: A Methodology and Initial Study ". Computer Vision and Image Understanding, vol. 69, no. 1, pp. 38-54 Jan. 1998
- [5] T. Peli and D. Malah. "A Study of Edge Detection Algorithms". Computer Graphics and Image Processing, vol. 20, pp. 1-21, 1982.
- [6] Illingworth, J., Kittler, J., "A survey of the Hough transform," CVGIP 44, pp.87-16, 1988
- [7] Illingworth, J., Kittler, J., "The adaptive Hough transform," IEEE Trans. PAMI-9 (5), pp.690-698, 1987.
- [8] E. Argyle, "Techniques for edge detection," Proc. IEEE, vol. 59, pp.285–287, 1971.
- [9] Wilhelm Burger, Mark J. Burge, Principles of Digital Image Processing, Springer-Verlag London, pp. 261, 2009.
- [10] S. Belongie, J. Malik, J. Puzicha. Shape matching and object recognition using shape contexts, PAMI, 24(4):509–522, 2002
- [11] Xiang Baia, Xingwei Yangb, Longin Jan Latecki. "Detection and recognition of contour parts based on shape similarity", ScienceDirect 2008.
- [12] Hankyu Moon, Rama Chellappa, and Azriel Rosenfeld, Life Fellow, IEEE "Optimal Edge-Based Shape Detection", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 11, NO. 11, NOVEMBER 2002

Radmila Sekulic
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail:
radmilasekulic90@gmail.com

Nebojsa T. Peric
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail:
nebojscha.perich@gmail.com

Milos Jordanski
University of Belgrade
Faculty of Mathematics
Studentski trg 16,Belgrade
SERBIA
E-mail:
jordanski90@hotmail.com

Prof. Milan Tuba
University Megatrend Belgrade
Faculty of Computer Science
Bulevar umetnosti 29N.Belgrade
SERBIA
E-mail: tuba@ieee.org

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

Tursib4iPhone Public transport in Sibiu made easy

**Emanuel Silivăsan
Teacher Coordinator: Delilah Florea, Adrian Florea**

Abstract

The fast paced evolution of technology and the need of the public to use devices that ease and enhance the way of life was the reason for developing this application. Thanks to the continuous size reduction of mobile devices and the increase in their performance, applications are becoming an asset in the life of each individual. Designing applications which make living in a city more convenient, which lighten traffic conditions, reduce pollution and that improve tourism are very important for the hustle and bustle of large cities. In the city of Sibiu, public transport is outdated but it is still used by all ages. Nowadays the younger citizens and tourists have a smartphone and an application that monitors the bus timetable offline and information of passing buses nearby would be very useful for planning and maybe for avoiding crowded transport and congested traffic.

1 Introduction

The article focuses on an application called Tursib4iPhone created to improve the current outdated public information of public transport in the city of Sibiu implemented on the iPhone, using the iOS Development Kit. The concept of this application, its features and behind-the-scenes coding will be thoroughly explained in this article. It will start by explaining the concept, the implementation, its functionality and ideas for further development. A few applications have been developed in this sense, but I have not stumbled upon an application that does exactly what Tursib4iPhone does yet.

In order to develop the code we use certain sources of information. These served not as a source of inspiration for the composition of the paper, but as learning materials for certain aspects related to the programming part. The work on several code-based problems, such as managing text files on the Apple operating system, development of the layout, etc. was supported by information obtained from the following books and websites [1], [2], [3] and [4]. Other research similar with our work may be found in [5], their tool, Moovit, having as disadvantage that it is not free to use.

2 The Tursib4iPhone concept

The concept will be explained in two possible scenarios. First scenario: You have just finished your work day and need to get home by bus, but you are hungry and you would like to eat something. You are not sure when the next bus arrives at the station but you need to get home quickly because you have another event planned, and going to the fast-food restaurant will take a few minutes. Running to the bus station to check the timetable will also be a waste of time because the restaurant is in the other direction and you probably will not have time to go back after you know when the next bus arrives. A second solution would be taking your smartphone and start browsing the Internet to find the timetable, which is both unpractical and annoying. A third option can only exist if that smartphone has an application installed that can automatically detect the nearest station around you and offer you information about the next buses. This fixes all issues, you can instantly check and see if your unexpected stop can be made.

Tursib4iPhone is a mobile application designed to improve the way people travel by public transport in Sibiu and can be used later even for other cities. It is specifically designed for the iPhone using its advanced features to make it as practical as possible. It was developed using Apple's native Objective-C language and the iOS Development Kit.

The reason why this application is very useful is because it uses real time GPS localization to pinpoint the user's position in relationship from a bus station. After the location was determined, the user gets a feed of information regarding the buses that stop at that station and the immediate timetable. The next bus that stops will be highlighted and it will indicate its end station and ETA (estimated time of arrival) displayed in minutes. The user can use this application in real time by touching the highlighted bus when he boards it. The new view will display the current boarded bus, the next station it will stop, ETA to the next bus stop (an average between the established time in the timetable and GPS arrival time, based on traffic conditions), and a list of all the stations till the end of the line with their respective ETA.

The application is practical because it does not really need Internet connection since it is using an internal database where all the station and bus data is saved. All station coordinates and all bus lines information are stored in an offline database using the *sqlite3* framework, a native database to iOS [6].

3 Implementation details

To make the concept become reality, I have used the public transport system of my hometown, Sibiu, as a mould on which to implement this application. The data stored inside the database has been manually inserted from the company's website and the coordinates of stations for demonstrative purposes were taken from Google Maps.

The tools I have used in making this application are: XCode 5.1 (the IDE), the iOS 7.1 SDK, SQLiteManager (for creating and managing the database) and an image editing software for designing the user interface. Going deeper into the genesis of the code the following have been used: the UIKit framework for the user interface, the CoreLocation framework to implement the GPS functions and the MapKit for coordinate calculations.

Apple's advanced IDE provides great means of designing the user interface using Storyboard. The user interface of the application can be designed using this feature. The storyboard of this application can be seen in Figure 1. Presently Tursib4iPhone has one parent view (the main menu) and 5 child views (or sub-views). Four of the sub-views are referred to by the main menu using buttons and one sub-view is referred by the most important view of the application, the "Station View". Before getting to this important view, I will briefly explain the other ones. Among other views, one of them contains a map of all bus stations and lines. In this view you can zoom in and out using the "Pinch" gesture, and move around to get a more detailed view of the stations. There is also a "Useful Information" view which is currently empty but will contain important information such as the company's phone number, the help desk phone number and address, ticket fees and other miscellaneous information. Going back to the "Station View", this view (see Figure 2) is made out of several sections. The concept and heavy duty is done in this view. The upper part of the view is the title and back button, followed by a "Bus Stop" sign, which is either red or blue. Below it there are either two or just one arrow pointing in opposite directions with a dot at the end. The importance of the colours and arrows is explained in the 4th section of this article. On the right there is a frame in which the closest station next to the device's position is displayed. This frame is also accompanied by a button, which the user can use to change the station in case it is not the preferred one. The second section is the black rectangle with the yellow writing. This is where the information regarding the next bus that arrives at this station is displayed, including its ETA (estimated time of arrival). I have used a custom font and background image to create the impression of the led display, which is usually located on the front or on the side of the bus. Once the ETA is less than one minute, the rectangle begins to blink yellow (created using animation functions) to get the users attention. The last section of the view shows a list of buses that follow after the highlighted bus, including their line number, direction and established time of arrival.

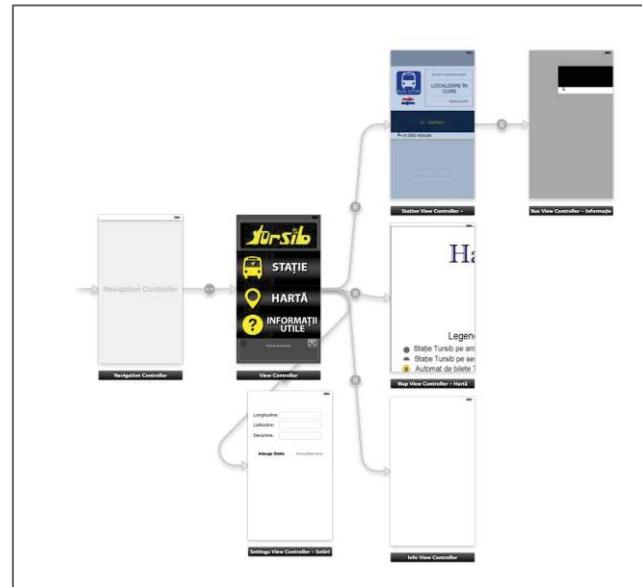


Figure 1



Figure 2

Looking behind the scenes of this attractive user interface, the algorithm starts the moment the user enters this view. The application, using the CoreLocation framework starts getting the user's location. If location services are disabled (if the user has not allowed the application to get its location yet or if the device does not have the required hardware) it will display a warning or an error message. Once the location is determined, the search ends to save battery power.

```
- (void)locationManager:(CLLocationManager *)manager didUpdateLocations:(NSArray *)locations
{
    location = [locations lastObject];
    [locMgr stopUpdatingLocation];
    [self loadStations];
    //NSLog(@"Location found!");
}
```

The program then calculates the closest station and queries the database for bus line information and displays it. Each station has a unique identifier and all bus tracks are checked to see if they stop at the selected station. If they do, the exact times when the buses are leaving the first station are loaded (in the HH:MM format and converted to fit algorithmic needs) together with the delays of each station (in minutes). This way, the exact ETA can be determined.

```
temporary = [NSString stringWithFormat:@"%@",sqlite3_column_text(compiledStatement, 2)]; //preluam detaliile
            traseului (orar, minplus)
NSArray* orar_statie = [temporary componentsSeparatedByString:@","];
temporary = [NSString stringWithFormat:@"%@",sqlite3_column_text(compiledStatement, 6)];
NSArray* minplus_statie = [temporary componentsSeparatedByString:@","];
```

After the data has been retrieved and conveniently converted, the lines are sorted by their time of arrival. The ETA is recalculated every few seconds and updated using a timer. Below you can see a sample of the bus tracks table, followed by a sample of the station table and a query.

```
if([currentDate compare: date] == NSOrderedAscending)
{
    t_interval = [date timeIntervalSinceDate:currentDate];
    traseu_temp.inter=t_interval;
    busTrasee* buf_traseu = [[busTrasee alloc] init];
    buf_traseu.numar=traseu_temp.numar;
    buf_traseu.directia=traseu_temp.directia;
    buf_traseu.inter=traseu_temp.inter;
    [cTrasee addObject:buf_traseu];
```

num	hour1	hour2	station1	station1_min	station2	station2_min
1	05:19,05:30,05:4...	05:34,05:45,06:0...	1,2,3,4,5,6,7,8,9,1...	0,1,2,3,4,5,7,9,10,...	20,21,22,23,24,2...	0,1,3,4,5,7,8,9,11,...
id	name					
1	CIMITIR 1		45.770457		24.138379	
2	TURSIB 1		45.77295		24.140467	
3	DIMA 2		45.77629		24.143615	
4	CALEA CISNADIEI		45.774621		24.148657	
5	INTEREX		45.774827		24.153448	
6	PIATA RAHOVA 2		45.777573		24.15827	

```
NSString*ultima_s = [NSString stringWithFormat:@"SELECT name FROM stations WHERE id=%i",[[t_statii objectAtIndex:t_statii.count-1] intValue]];
```

Sqlite3 offers improvements compared to other database systems because it is lightweight, it uses fast algorithms to search for data, and the data you need, can be selected without loading the entire database into memory, which in case of mobile devices is very important.

When the ETA of the highlighted bus is blinking, meaning the bus is arriving, the user will be able to tap on the rectangle and be transferred into a new view. The user will be prompted to do that if he boards that bus. In this view, he will be able to see at which stations the bus will stop next and the ETA. He will also be able to see connecting bus lines at stations of interest, as well as the speed and the street where the bus is (a optional fun feature, if the user has internet connection and is willing to use it for this).



4 Conclusions and further work

The last paragraph in the implementation section is actually about further work to be done, as it has not been implemented yet.

The only problem that I have currently encountered is separating stations, which have two directions. By this I mean stations with the same name (on narrow streets for example), having the same GPS coordinate (since the practical GPS accuracy of an iPhone is at 5 meters) where buses go both ways. That is the reason for implementing the red and blue colored bus sign and arrows. If a station meets those conditions, it will display both arrows and will act as a button, see Figure 3. The user can press it and toggle the 'direction' of the station. This way, he can see bus information of buses going in the opposite direction.



Figure 3

Other future developments would be improving the user interface to make it easy to understand and the displayed feed of information reliable (including sound notification on bus arrival for example, ETA in timetable with real time traffic conditions comparison).

Important steps would also be to improve overall performance on getting the information (since larger databases would require more efficient data handling). The "Useful information" view will be using 'static' data (hard-coded into the application) to display as information, if the user has no internet access, and a 'dynamic' feed of information coming from a server administered by the transportation company where information like downtime, delays and unforeseen events can be posted.

All in all, Tursib4iPhone would be a great project to complete because it will offer the citizens of Sibiu a relief by digitizing the old and annoying paper-based timetable, as well as making it dynamic and interactive for the user. Even more than that, this application can be modified to fit any public transportation system nationally or even internationally. Many people avoid public transport for fear of losing the bus, which has negative consequences at the work place. Having the time schedule and interactive information at a push of a 'screen' will make travelling by bus a more pleasant means of transport.

5 References

- [1] Apple iOS Developer Library (<https://developer.apple.com/library/ios/navigation/>)
- [2] David Mark, Jack Nutting & Jeff LaMarche, *Beginning iOS6 Development: Exploring the iOS SDK*, Apress, ISBN (electronic): 978-1-4302-4513-1.
- [3] Neuburg Matt, *Programming iOS 6*, O'Reilly, ISBN: 978-1-449-36576-9.
- [4] Stackoverflow (<http://stackoverflow.com>) (accessed 18 April 2014)
- [5] TranzMate, Moovit: Live Transit App for Bus Train Metro Subway with Map Schedules and Next Bus, (<https://itunes.apple.com/us/app/moovit-live-transit-app-for/id498477945?mt=8>) (accessed 21 April 2014)
- [6]SQLite, (<https://sqlite.org>) (accessed 21 April 2014)

Emanuel SILIVĂŞAN
„Samuel von Brukenthal“ National College
Piața Huet, 550182 Sibiu
ROMANIA
E-mail: emanuel.silivasan@hotmail.com

The logistics management a transportation system «point of departure – point of destination» in real time regime

Maria Sokolova

Teacher Coordinator: Prof.dr.Vladimir Chernov

Abstract

The paper illustrates the using of Neural Fuzzy Inference Systems for solving a particular problem and describes possible solutions of the problem in the program MATLAB. The problem is defined and the steps to solve the problem are detailed. The paper shows the impact of experimental data on the quality of the results.

The fuzzy linguistic variables and the some images of the program MATLAB are shown and explained. The obtained results of using ANFIS to solve the problem are analyzed.

1 Introduction

In recent years much attention is paid to a problem of planning, organization and management of work of the logistic delivery systems of goods in the conditions of risk and uncertainty of environment.

In this paper the transport system management «point of departure - point of destination» in real time regime was analyzed, based upon application of methods of theory of fuzzy sets and artificial neuron networks. The problem of coordinated management of links of goods delivery process by forming of rational route of delivery is solved. The approach for solving the problem is based on the fuzzy neural network ANFIS in the program MATLAB.

The work consists from general description of a neural network, description of process of goods delivery, sequence of development of a neural network, results of testing and plan of further researches.

2 Description of a neural network

The efficiency of the logistics cargo management is achieved through coordinated interaction of separate parts and elements of the logistics chain of cargo delivery in the system "place of departure - place of destination" by rapid response to the incoming current information about changing the conditions at all stages of the transport process.

It's necessary to process a large flow of complex information about the current state of the transport system's objects, managing cargo in real time. So the use of artificial neural networks is

proposed.

In order to coordinated interaction of different transport's types effectively on different stages of the transport cargo delivery process to the destination in the real-time mode, it is offered to submit a fuzzy inference system in the form of neuro-fuzzy network [4].

Generally, mathematical model of the system "point of departure - point of destination" is a functional dependence between the variables of the system states, control actions, the observed parameters of the system and the external environment [2]:

$$\bar{Y}(t) = F(\bar{X}(t), \bar{U}(t), \bar{V}(t)), \quad (1)$$

$\bar{X}(t)$ – the vector of the current state of the system model; $\bar{V}(t)$ – the control actions vector; $\bar{U}(t)$ – vector of external influences; $\bar{Y}(t)$ – the vector of output model signals.

Neural network is a directed graph which vertices are neurons, external input or output variables, the arc is the direction of distributing or synapses. The number of layers of neurons and connections between them depends on the specific task.

Neuron has a certain internal structure, shown in Figure 1, and the rules for reformation of signals [1]. A neuron consists of multipliers (synapses), accumulator and nonlinear converter. Synapses are intended for communication between neurons and multiply the input to the weight of the synapse.

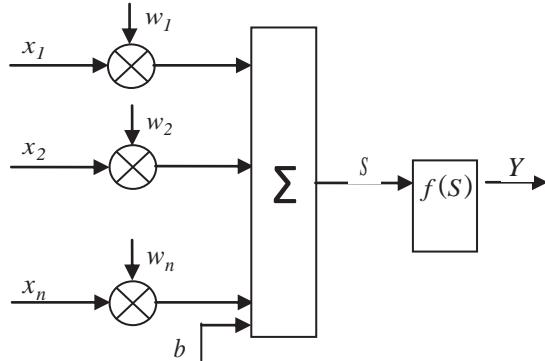


Figure 1 – The Structure of the neuron

The rules for converting signals are defined as the mathematical model of a neuron:

$$s = \sum_{i=1}^n w_i \cdot x_i + b ; \\ y = f(s) , \quad (2)$$

w_i – the weight of the synapse ($i = \overline{1, n}$); b – the offset value; s – the result of summation; x_i – component of input signal vector ($i = \overline{1, n}$); y – the output of a neuron; n – the number of input neurons; f – activation function (transfer function) of a neuron, which represents a non-linear transformation.

3 Description of process of goods delivery

The process of cargo delivery in the system "PDep - PDes" is presented by the following stages:

1. shipment at the shipper enterprise;
2. the transportation of goods to the destination with selecting the type of transport and route;
3. unloading at the destination point.

Each stage is characterized by a set of parameters that reflect the level of transport service, which at the design stage are fuzzy linguistic variables, so it is reasonable to use fuzzy sets [3].

Input fuzzy linguistic variables:

- 1 - type of transport;
- 2 - the speed of cargo delivery;
- 3 - safety of cargo;
- 4 - dependence on climatic conditions;
- 5 - bandwidth of station;
- 6 - the costs of cargo shipping.

Now we are setting term-sets of variables following verbal - numeric scale (Table 1):

Table 1
Verbal - numeric scale of variables

Designation	Linguistic evaluation	Numeric value
type of transport	Automotive	[0 1 2]
	Train	[1 2 3]
	River	[2 3 4]
	Marine	[3 4 5]
	Air	[4 5 6]
the speed of cargo delivery	Low	[-500 0 500]
	Average	[0 500 1000]
	High	[500 1000 1500]
safety of cargo	Bad	[-5 0 5]
	Average	[0 5 10]
	Good	[5 10 15]
dependence on climatic conditions	Low	[-5 0 5]
	Average	[0 5 10]
	High	[5 10 15]
bandwidth of station	Low	[-50 0 50]
	Average	[0 50 100]
	High	[50 100 150]
the costs of cargo shipping	Small	[-5e+004 0 5e+004]
	Average	[0 5e+004 1e+005]
	Large	[5.132e+004 1.013e+005 1.513e+005]

It's required to choose the most efficient route of cargo delivery with the required level of the transport service adequacy. For solving the problem the method based on fuzzy neural network ANFIS in MATLAB has been proposed.

ANFIS is an abbreviation for Adaptive Neuro-Fuzzy Inference System (ANFIS system). ANFIS editor allows you to synthesize neuro-fuzzy network from experimental data automatically. Neuro-fuzzy network can be considered as one of the varieties of fuzzy logic conclusion systems (type Sugeno). The facilities function of synthesized systems configured (trained) to minimize the deviations between the results of fuzzy modeling and experimental data. The command "anfisedit" is used to open ANFIS editor.

4 Sequence of development of a neural network

The General sequence of the process of developing hybrid network models can be represented by the following steps:

1. Preparing the file with the training data (Fig.2). It is reasonable to use the MS Excel. Training data must be stored in an external file with the *.dat extension. (In the example file 1.dat.)

	1	2	3	4	5	6	7	8
1	1	0	0	10	0	10000	0	
2	1	10	1	9	10	9000	0.1	
3	1	20	2	8	20	8000	0.2	
4	1	30	3	7	30	7000	0.3	
5	1	40	4	6	40	6000	0.4	
6	1	50	5	5	50	5000	0.5	
7	1	60	6	4	60	4000	0.6	
8	1	70	7	3	70	3000	0.7	
9	1	80	8	2	80	2000	0.8	
10	1	90	9	1	90	1000	0.9	
11	1	100	10	0	100	0	1	
12	2	0	0	10	0	10000	0	
13	2	1	1	9	10	9000	0.1	
14	2	2	2	8	20	8000	0.2	
15	2	3	3	7	30	7000	0.3	
16	2	4	4	6	40	6000	0.4	
17	2	5	5	5	50	5000	0.5	
18	2	6	6	4	60	4000	0.6	
19	2	7	7	3	70	3000	0.7	
20	2	8	8	2	80	2000	0.8	
21	2	9	9	1	90	1000	0.9	
22	2	10	10	0	100	0	1	
23	3	0	0	10	0	10000	0	
24	3	15	1	9	10	9000	0.1	
25	3	30	2	8	20	8000	0.2	

Figure 2 - Piece of data for the study of neuro-fuzzy network

2. You should open the ANFIS editor (comand “anfisedit”) and upload a file with the training data. Press button “Load Data”, then the dialog box of file selection (Fig. 3) will appear.

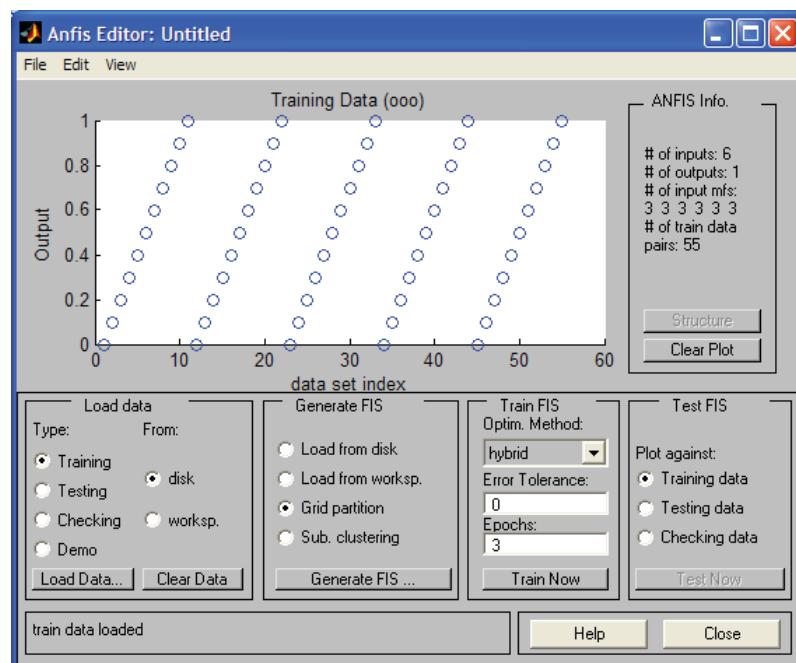


Figure 3 - GUI of ANFIS editor after loading the training data

3. After preparation and uploading the training data you can generate the structure of the FIS system (Sugeno type) that is a model of a hybrid network in Matlab. For this purpose you should use the button “Generate FIS” in the lower part of the working window of the editor.

Then you should point the number of terms (Grid partition method) for each input variable (Number of Mfs: 5 3 3 3 3 3) and the type of membership functions (Fig. 4) for input and output variables (MF Type: constant).

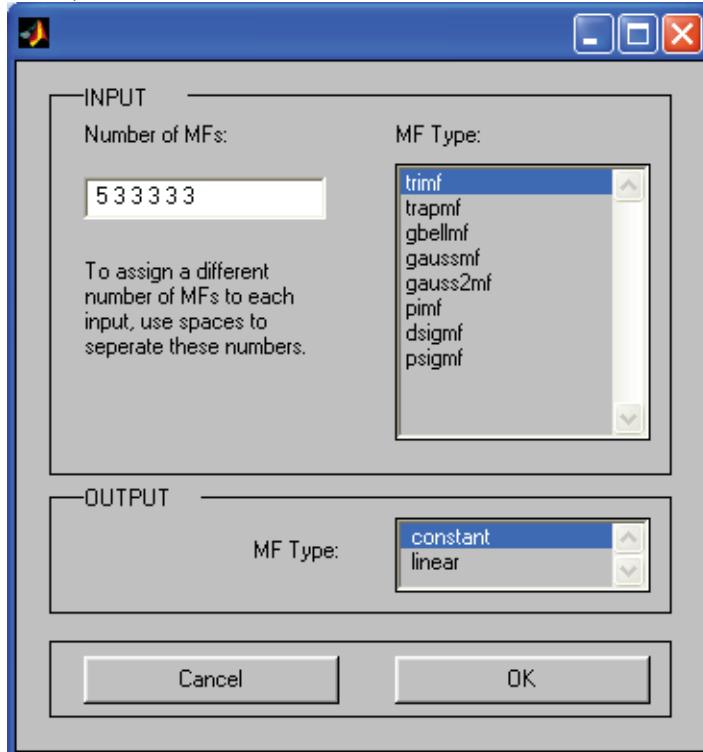


Figure 4 -Dialog window for specifying the number and type of membership functions

In the main MATLAB window the characteristics of the network (Fig. 5) appear.

```
ANFIS info:  
Number of nodes: 2479  
Number of linear parameters: 1215  
Number of nonlinear parameters: 60  
Total number of parameters: 1275  
Number of training data pairs: 55  
Number of checking data pairs: 0  
Number of fuzzy rules: 1215
```

Figure 5 - Characteristics of the created network

4. After generation the hybrid network, you can visualize its structure, pressing the button “Structure” in the right part of the graphics window (Fig. 6).

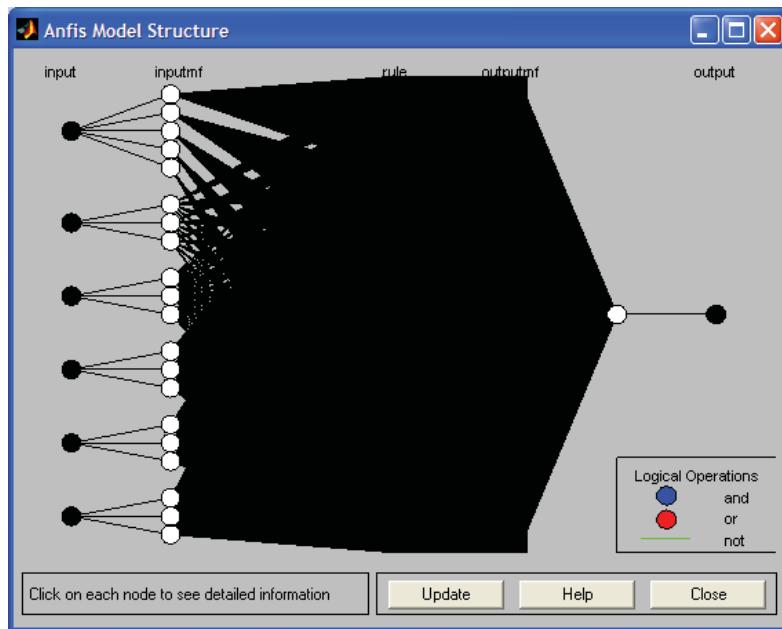


Figure 6 - Structure of the generated fuzzy inference system

5. Before the hybrid network training, you should set the training parameters by using the following group of options in the right bottom of a working window:

1) Choose the method of hybrid network training - back-propagation (backpropo) or hybrid, which is a combination of the least squares method and the method of decreasing the reverse gradient (select hybrid).

2) Set the level of training error (Error Tolerance) - the default value is 0 (not recommended to change).

3) Set the number of cycles (Epochs) - the default value is 3.

For the network training, you should click “Train now”. The progress of the training process is illustrated in the render window in the form of the graph of the error dependence on the number of cycles (Fig. 7).

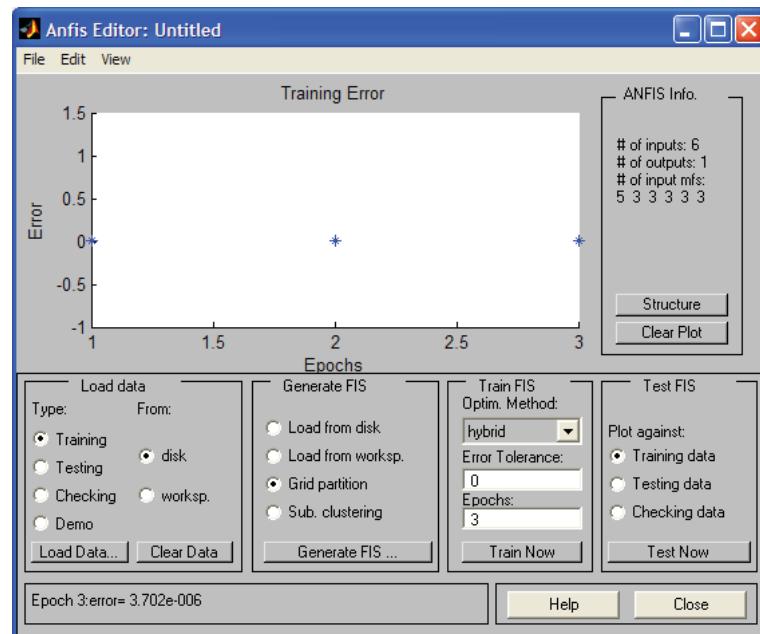


Figure 7 - The graph of errors training dependence on the number of cycles

Training completed at epoch 2 (Fig. 8).

```

Start training ANFIS ...

1      3.70203e-006
2      3.71846e-006

Designated epoch number reached --> ANFIS training completed at epoch 2.

```

Figure 8 – Training completed at epoch 2

6. Further configuration options of built and trained hybrid network can be done using standard graphical tools of the package Fuzzy Logic Toolbox. For this purpose it is recommended to save the created system of fuzzy inference in an external file with the extension *.fis. After that you should upload this file in the editor of fuzzy inference systems FIS (let's call a saved network set1.fis).

Using the graphical interface FIS Editor you should rename linguistic variables (Fig. 9) and change the names of their terms (Fig. 10). In FIS Editor we can also see the base of fuzzy rules (Fig. 11) and the surface of fuzzy inference (Fig. 12).

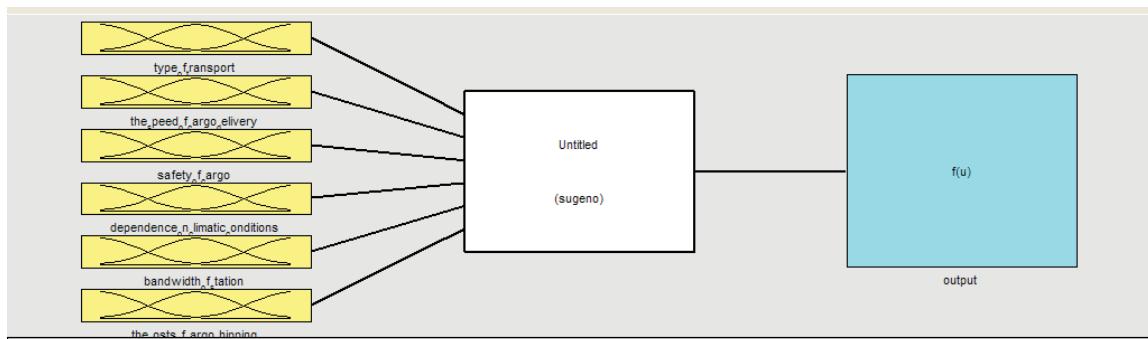


Figure 9 - GUI FIS editor for generated fuzzy inference system

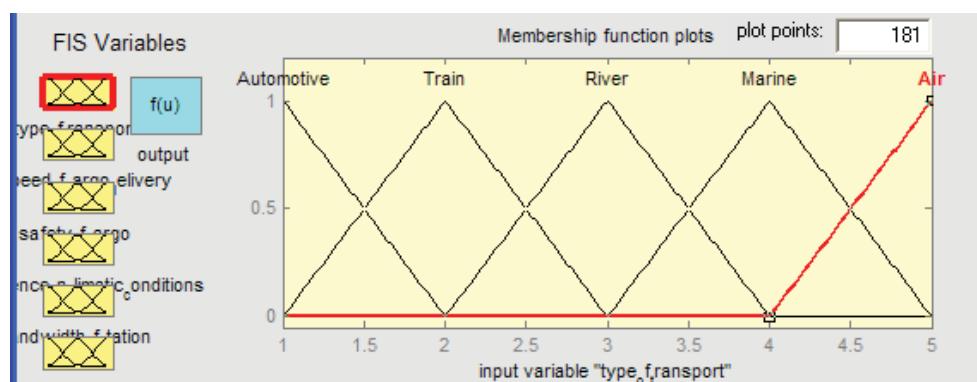


Figure 10 - GUI editor of membership functions of fuzzy inference system

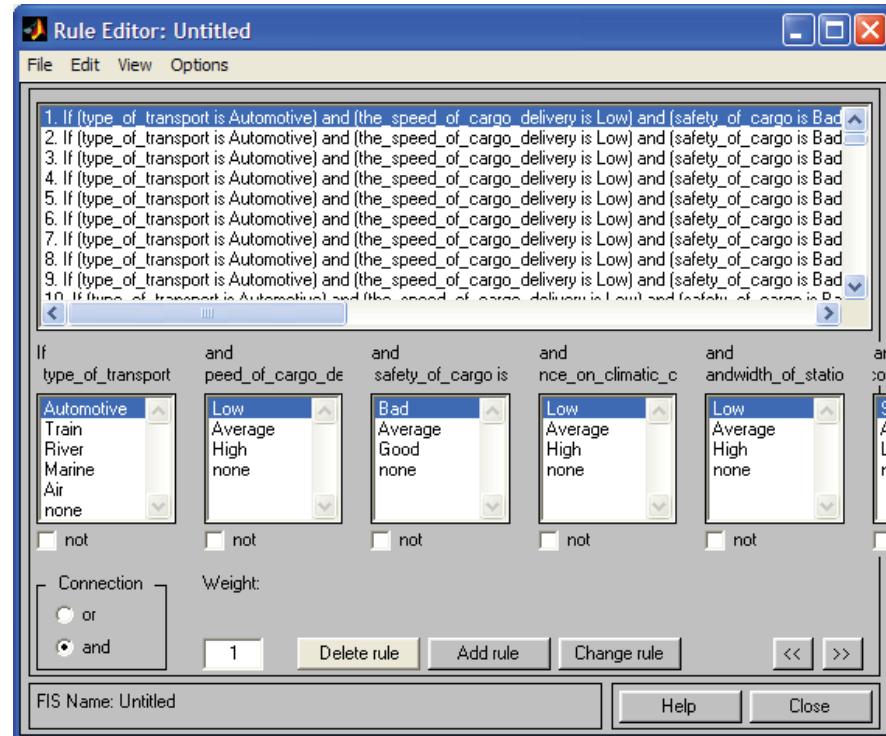


Figure 11 - The fragment of the fuzzy rules base

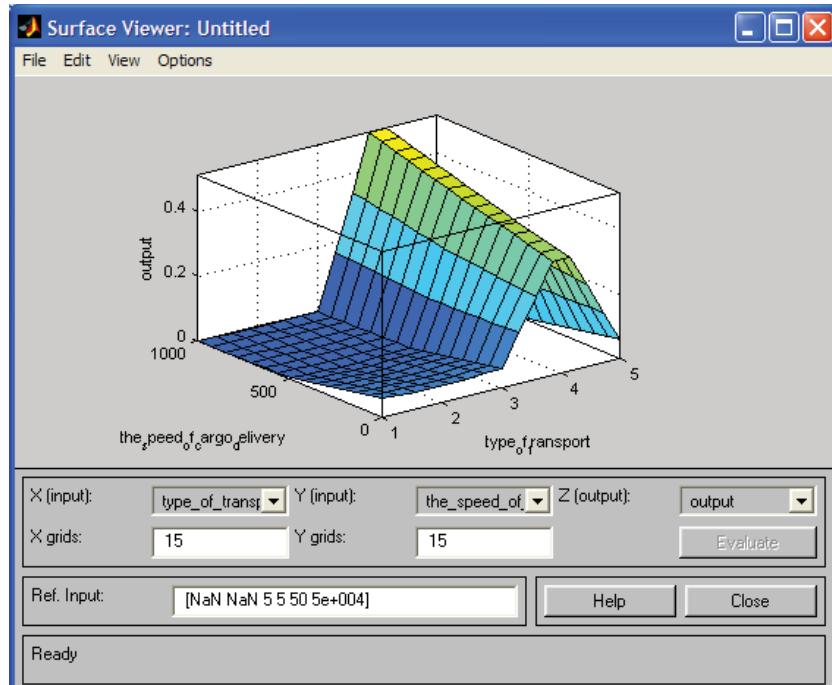


Figure 12 - Surface of fuzzy inference

7. The next step is testing of fuzzy system and getting the output of results in the visualization. Upload information for testing ("Load data"), select the option Testing (a file with the sample is "1.dat"). In the testing field ("Test Fis"), select the Testing data and run the test by "Test Now" (Fig. 13).

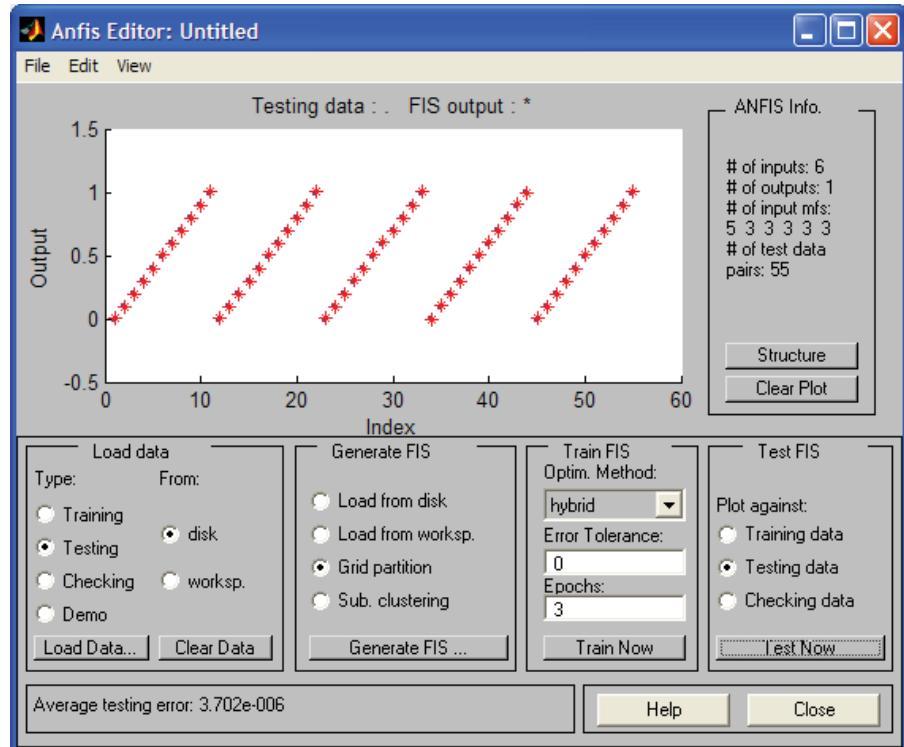


Figure 13 - Network Testing

8. Now we should verify the adequacy of the constructed fuzzy model of hybrid network. To solve this task, you should use the function “`evalfis`”.

The results of the forecast for the specific input is obtained by typing the following code in the command area (Fig. 14).

```
>> x=[1 70 7 3 70 3000] %the input parameters
x =
1      70      7      3      70      3000
>> fis = readfis('set1.fis'); %upload file set1.fis
>> y = evalfis(x, fis) %conclusion of the forecast
y =
0.7000
```

Figure 14 - Review of model adequacy of a hybrid network

The obtained value of the output of the linguistic variable “adequacy of transport service” for membership function $y=0,7$ is the result of solving the problem of fuzzy inference for the proposed linguistic values of input variables at a certain stage of rational delivery route selection. It means the less time of cargo delivery to the destination.

5 Conclusions

The obtained results give grounds to speak about the possibility of practical application of the neuro-fuzzy network to improve the efficiency of logistic management of the cargo delivery transport system.

Further researches consist in developing fuzzy situation networks for operational decision-making in the management of all participants interaction at the delivery system in real time.

We can draw the following conclusions:

1. To coordinate the interaction of the logistics chains in real-time, operating in conditions of uncertainties, the usage of neuro-fuzzy network is proposed.
2. The management of cargo traffic in real time is solved by using the method of fuzzy logical conclusions (the theory of fuzzy sets), implemented by artificial neural network in the MATLAB (package Fuzzy Logic Toolbox).
3. The proposed approach of coordinated interaction of the transport system elements and parts "PDep - PDes" allows to reduce the time of cargo delivery.
4. On the basis of this experiment we can make the conclusion that the use of ANFIS for solving the problem of coordination of logistics chains interaction is very promising.

The disadvantage is that the quality of the results depends on the quality of experimental data or training data. Therefore, the selection of training data is an important process in using ANFIS.

References

- [1] Gubenko V.K., *Efficiency of the logistics system of cargo delivery*, Visnyk Priazovskogo sovereign technical University, 2007.
- [2] Nefedov Y.A., *Logistic management of the transport system maintenance "metallurgical enterprise - port" in real time*, 2010.
- [3] Leonenko A.V. *Fuzzy modeling with MATLAB and fuzzyTECH*, BHV-Petersburg, 2003.
- [4] Stovba S. D., *Designing fuzzy systems by means of MATLAB*, Moscow, 2007.

Maria SOKOLOVA
Vladimir State University
Faculty of Information Technology
Department of Management and Informatics in
technical and economic systems
Str. Gorky, 87, Vladimir
RUSSIA
E-mail: sok.masha2011@yandex.ru

Highlight Image Filter Produces Outstanding Results in Optical Character Recognition

**Iulia Știrb
Teacher Coordinator: Horia Ciocârlie**

Abstract

Image filtering is changing the appearance of the image by altering the shades and colors of the pixels. Increasing the contrast as well as adding a variety of texture, tones and special effects to images are some of the results of applying filters. Still the main purpose is reducing the noise of text images in order to improve the OCR (Optical Character Recognition). Within Silicon & Software Systems Limited (S3Group) Company, image filters improve the process of testing set-top boxes by using optical recognition of the current settings (i.e. text) that are displayed on the output video of the set-top box under test. The important role played by image filters in improving a subsequent OCR processing on images was the reason I created two new efficient filters that will be presented in this paper.

The first filter, named Smart Contrast, increases the contrast of the image in a way depending on the value of each component (Red, Green and Blue) of each pixel in the image.

The second filter, called Highlight, produces outstanding results when it comes to perform an OCR on the filtered image. As it is carried out, the implementation differs from all other known filters, while the visual effect on the filtered image can be described as a combination between contrasting in a manner similar to Smart Contrast and other two visual effects: sharpening and highlighting the edges. Precisely, this way of combining the resulting visual effect makes the filter so powerful in improving OCR on images which contain text.

1 Introduction

Researches reveal that specific filters are used for certain images. For instance if the original image is blurred and the expected result would have to be an image with a higher clarity than Sharpen filter can be used and on the other hand if a less level of details in the image are desired, than Blur filter would be the right choice. The choice of the filter is, thus, done by humans after observing the characteristics of the image (color, shape, thickness of text). As in [2], the process of selecting the scale of the filter in order to perform edge detection over the image can be automated. Overall, as in [1], researches have been carried out regarding the automated selection of the filters' parameters. In other words, once the proper filter to apply to the image has been chosen by humans, the filter parameter is selected by the computer depending on the desired output.

However, an automated analysis of the image in order to select the proper filters to be applied to it would be a complicated, expensive and time consuming process since the analysis depends on many factors (e.g. noise, clarity, contrast). Highlight image filter, which will be presented in this paper, is meant to avoid this analysis, since it is a universal filter designed to improve OCR on a large variety of text images.

Beside highlighting and sharpening the text in an image, Highlight filter contrasts the image in a similar way as Smart Contrast filter does. The filter which I entitled it Smart Contrast performs a selection between two types of transformation and chooses the proper one to be applied. Instead of simply applying the same transformation to all components of each pixel like Contrast filter would do, the selection is done for each component (e.g. Red) of each pixel separately. The improvement made by Smart Contrast in various cases is that it contrasts even more the text in the image when the color of the text is close to the color of the background making the text more visible than Contrast filter does.

This paper contains two major sections in the second chapter in which are described the two new filters I have created: the first section describes how Smart Contrast filter works and the second present some already existing filters and Highlight new filter. Snippets of code from Highlight filter implementation are shown in Chapter 3. Conclusions Chapter presents the major benefit Highlight filter brings that is improving the rate of OCR success in comparison to other filters and describes the visual effect on images of both Smart Contrast and Highlight filters.

2 View on existing image filters

Contrast filter increases the contrast in an image by modifying all components of the image (i.e. Red, Green and Blue) except Alpha, according to different mathematical models such as the one in (1). In fact, the Alpha component remains unchanged for the majority of image filters.

Color filter has a variety of forms. Some of them keep unchanged the value of one component (e.g. Red) or two components (e.g. Green and Blue) and assign the other component(s) to zero. In addition, others may produce grayscale images.

Invert filter applies an old look on the images similar to first TVs with black and white screens, except for the fact that the images are coloured (if the original image isn't black and white only).

Sharpen filter accentuates edges in an image, by applying a fixed $N \times N$ dimension mask (also named kernel) to all pixels. The new value of the current pixels depends on all neighboring pixels covered by the mask which is centered in the current pixel.

Blur filters reduces the details in an image by blurring it. This is done using a mask as well but with different values from the ones used for Sharpen filter mask.

3 Smart Contrast and Highlight image filters description

3.1 Smart Contrast image filter

Smart Contrast filter compares the value of each component (e.g. Red) of each pixel with 127 ($255/2$, note that the range in which the components Red, Green and Blue vary is 0-255) and if the value is less if less than 127 perform a certain transformation to that specific component, if greater perform a different transformation. As a case study, if it is considered an image which contains some text and the color of the text would be $(R_t, G_t, B_t) = (126, 126, 125)$ and the color of the background would be $(R_b, G_b, B_b) = (130, 137, 136)$, then the colors would be pretty similar, so the text would be hard to recognize even for the human eye. In this case, Smart Contrast filter decreases more the color of the text and increases more the color of the background than Contrast filter would do, making the text more visible and more easily to be detected. Thus, Smart Contrast keeps the good work Contrast filter does and, in addition, produces good results for edge cases.

3.1.1 Important step in creating Smart Contrast filter: knowing how Contrast filter works

Contrast filter is based on the transformation in (1) where *contrast* is the contrast scale (the degree to which the contrast is increased on image) and *red*, *green* and *blue* are the values of the components of a pixel:

$$((red / 255.0 - 0.5) * contrast + 0.5) * 255.0 \quad (1a)$$

$$((green / 255.0 - 0.5) * contrast + 0.5) * 255.0 \quad (1b)$$

$$((blue / 255.0 - 0.5) * contrast + 0.5) * 255.0 \quad (1c)$$

The graphic representation for this transformation (blue plot) is represented in Fig. 1. In the same figure, the identity function it is also being drawn to spot how the values of pixel components increase or decrease according to the transformation. If the value of the pixel component is less than zero it is set to zero and if it is greater than 255 it is set to 255. So far, the same transformation is being applied to all components of each pixel.

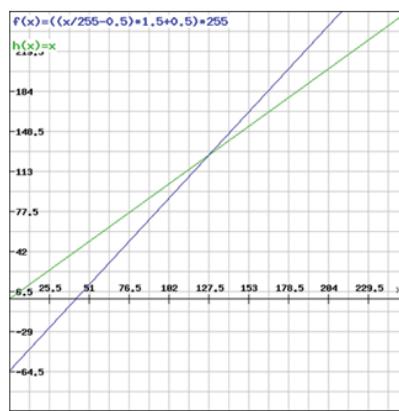


Fig. 1: Contrast filter transformation (bleu plot)

3.1.2 New image filter: Smart Contrast

Smart Contrast performs two transformations depending on the value of the pixel component. For less than 127 values the formula is illustrated by (2).

$$((value/255.0-0.6) * contrast + 0.6) * 255.0 \quad (2)$$

Equation (3) illustrates the transformation applied to values of pixel components greater than 127.

$$((value/255.0-0.4) * contrast + 0.4) * 255.0 \quad (3)$$

Thus, the same filtered pixel could be the result of applying two types of transformations to its components (e.g. (1) applied to Red component and (2) applied to Green component of the pixel), but, no more than one transformation will be applied to a single component of a pixel (e.g. Red component could not possibly be the result of applying (2) and (3), it will have to be either (2) or (3), but not both). Two different pixels could be the result of applying different transformations to the same component of the two pixels (e.g. the filtered Red component of the first pixel that has the coordinates (x_1, y_1) could be the result of applying (2) and the filtered Red component of the second pixel that has the coordinates (x_2, y_2) could be the result of applying (3)). The graphic representation of the two transformations is shown in Fig. 2A, together with the identity function that helps in spotting the way pixel components are increased or decreased. Fig. 2B highlights the difference between Contrast and Smart Contrast algorithms. In both Fig. 2A and 2B, if the value of the pixel component will be less than 127 the blue plot represents the transformation that is applied to a certain component, else the red plot represents the computations that will be made on the component.

Best OCR results for the filtered image using Smart Contrast filter are produced when contrast scale is set to 1.5.

3.1.3 Visual results of applying Smart Contrast image filter on images

Smart Contrast filter produces the results shown in Fig. 3. The results produced by Contrast filter are also shown in the same figure in order to spot the improvements made by Smart Contrast. The effect of applying Smart Contrast filter would be that, in most of the cases, contrast is increased

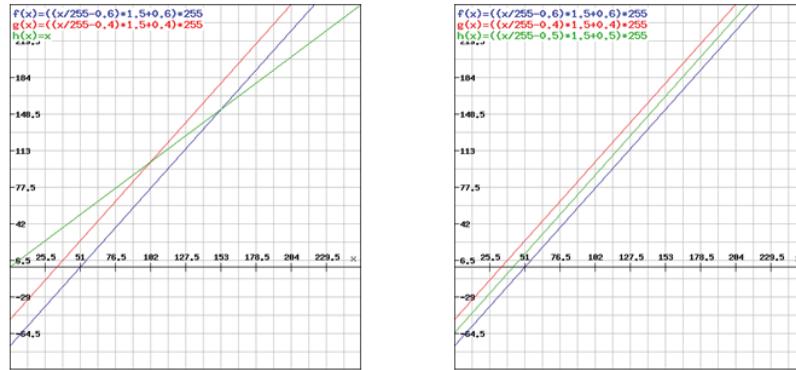


Fig. 2: A. Smart Contrast transformations (blue and red plot) and the identity function (green plot)

B. Smart Contrast (blue and red plots) and Contrast transformations (green plot)

the areas of the image where characters appear. Exceptions occur when the color of the characters is close to the color of the background. This drawback is solved in Highlight filter.

ALL CHANNELS	No filter	Golf: Women's British Open
ALL CHANNELS	Contrast	Golf: Women's British Open
ALL CHANNELS	Smart Contrast	Golf: Women's British Open

Fig. 3: The visual effect of Smart Contrast filter

3.2 Highlight image filter

The way Highlight filter increases the contrast of the images is similar with the way Smart Contrast does, but is slightly improved. Furthermore, Highlight filter detects the areas of rapid intensity change like Laplacian or Gaussian filter would do, but still, the two would have totally different implementations. In addition, it creates shadows behind characters.

Once the edges are detected, they are being sharpened which would produce an effect that is similar to what Sharpen filter would do to an image, but again, the two would have no similarities with respect to the their implementation. All this combined properties contribute to a better rate of OCR success.

With respect to the implementation, what Highlight image filter brings new is that the filtered value of the current pixels depends only on two neighboring pixels, which reduces the amount of noise taken into consideration when computing the filtered pixels.

3.2.1 Important step in creating Highlight filter: understanding how Sharpen and Laplacian of Gaussian filters work

Smart Contrast produces an effect similar to what Contrast does, that is increasing the contrast of the image. In addition, Smart Contrast decreases more the values of pixel components that are less than 127 and increases more the values greater than 127 than Contrast would do. Highlight filter contrasts the image in a manner similar to Smart Contrast.

Sharpen filter accentuates edges, but it does as well with the noise, as in [3], which is undesired and could make the OCR produce worse results than with the unfiltered image. Highlight filter takes the concept of spotting the edges from Sharpen filter, but does not accentuate the noise. Laplacian of Gaussian combines the effects of Laplacian filter and Gaussian filter (which blurs the images in order to reduce the sensitivity to noise). While Laplacian detects the regions of rapid intensity change therefore being used in edge detection, Laplacian of Gaussian sharpens edges between two regions of uniform color but different intensities, as in [4].

3.2.2 New image filter: Highlight

Highlight filter gathers together the concepts from Smart Contrast, Sharpen and Laplacian of Gaussian in an original implementation using no convolution kernels (masks) like the last two mentioned filters do. Highlight filter performs a different contrast increase for each component of each pixel in the image. In other words, a component (e.g. Red) of the current pixel is increased or decreased depending on the value of the same component of the pixel in discussion and two other pixels. The contrast scale of the e.g. Red component of the current pixel (pixel with coordinates (x,y)) depends on the values of the Red components of the pixels filled with Red color in Fig. 4 (pixels with coordinates (x,y) , $(x-1,y-1)$, $(x-2,y-2)$, $x>1$, $y>1$). While the current pixel is spotted in this figure with a high color intensity, the other two pixels are filled with lower color intensity in order to be differentiated from the rest of the surrounding pixels, not filled with any color, that have nothing to do with the current pixel when computing the contrast scale for it.

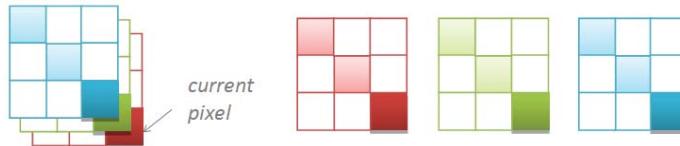


Fig. 4: Pixels (filled with color) which count in computing the value of the contrast scale of each individual component (i.e. Red, Green and Blue) of the current pixel

The way contrast scale of each individual component of the current pixel is computed is emphasize by (4), where r , g and b indicate the Red, Green and Blue components and xy , x_1y_1 , x_2y_2 indicate the pixels with coordinates (x,y) , $(x-1,y-1)$ and $(x-2,y-2)$, so, for instance, $r_{-x_2y_2}$ will be the value of the Red component of the pixel with coordinates $(x-2,y-2)$.

$$(100 + |r_{-x_1y_1} - r_{xy}| + |r_{-x_2y_2} - r_{-x_1y_1}|) / 100 \quad (4a)$$

$$(100 + |g_{-x_1y_1} - g_{xy}| + |g_{-x_2y_2} - g_{-x_1y_1}|) / 100 \quad (4b)$$

$$(100 + |b_{-x_1y_1} - b_{xy}| + |b_{-x_2y_2} - b_{-x_1y_1}|) / 100 \quad (4c)$$

Equation (4) is not applied to the left and top edges of the image, a simplified one is being applied in that case, but this is not part of the core of the algorithm and it shall be omitted to be presented for this reason. Equation (4) produces values in the range from 1.0 to 6.1.

Once the contrast for each component of each individual pixel in the image has been recorded, the algorithm is ready to be applied. The starting point of Highlight filter algorithm is based on the fact that human eye is sensitive to a difference of at least 30 between the values of at least one of the same component of two adjacent pixel when it comes to perceive and recognize characters. To be more specific, if we would have to write some characters on a uniform color background with all the pixels having the same color e.g. (0,10,20), the color of the characters would have to be (0+40,10,20) or (0,10+31,20) or any other combination that would meet the above request, in order for the human eye to recognize what is written. Fig. 5 proves what is being said before. The background of the picture is uniformly colored with black i.e. (0,0,0) and the color of the characters is (0,31,0). The condition is fulfilled because the difference between the Green component of the background color and the Green component of the text color is exactly 31 (greater than 30) - it could be more, but is enough for the text to be perceived by human eye. A

reasonable assumption that is made from the start is that, characters that would not be perceived by the human eye, are not expected to be recognized by a machine using OCR, but every character perceived by the human eye must be also recognized by a machine (assuming there is no noise), or at least expect to be recognized. Thus, for the text that could not be perceived by the human eye, the performances of OCR are not improved by first filtering text images using Highlight filter.

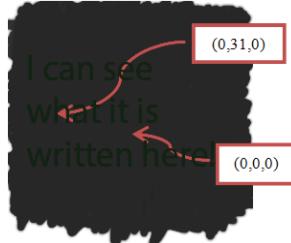


Fig. 5: The condition to be met for characters to be recognized by human eye

The algorithm is described in the following. Since an example makes the general case more explicit, it would be better to bring into discussion, for instance, the Blue component of the pixel having the coordinates (x,y) relative to the upper-left corner of the image. If the absolute difference between the value of the Blue component of the pixel with coordinates $(x-1,y-1)$ and the value of the Blue component of the pixel having the coordinates (x,y) is greater than 15 and the absolute difference between the value of the Blue component of the pixel with coordinates $(x-2,y-2)$ and the value of the Blue component of the pixel with coordinates $(x - 1, y - 1)$ is also greater than 15 than a certain transformation will be applied to the Blue component of the pixel having the coordinates $(x-2,y-2)$, assuming x and y are greater than 2. Leaving aside, for the moment, exactly which transformation will be applied to the Blue component of the pixel with coordinates $(x-2,y-2)$, the required thing right now will be to clarify that this condition will be also applied separately to other two components (i.e. Red and Green) of each pixel in the image except the ones in the right and bottom edges of the image (those will not be filtered). The condition described is shown in (5).

$$|r_{x_1y_1} - r_{xy}| > 15 \text{ &&} |r_{x_2y_2} - r_{x_1y_1}| > 15 \quad (5a)$$

$$|g_{x_1y_1} - g_{xy}| > 15 \text{ &&} |g_{x_2y_2} - g_{x_1y_1}| > 15 \quad (5b)$$

$$|b_{x_1y_1} - b_{xy}| > 15 \text{ &&} |b_{x_2y_2} - b_{x_1y_1}| > 15 \quad (5c)$$

Why 15? In many real situation the minimum difference 30 between the color component of the background and the color component of the text could be spread among three adjacent pixels (i.e. located around the edges of the text in the image) having three different colors, instead of two adjacent pixels like it is in the case of using just two colors i.e. text color and background color as in Fig. 5. These three different colors must fulfil the condition in (5). For instance, if the color of the background would be $(0,0,0)$, the intermediate color (i.e. noise color) would have to be $(0,0,16)$ and the text color would have to be $(0,0,32)$, for the condition to be fulfilled.

Sure, any increase of the Blue component of the intermediate color and/or of the text color will do nothing else than produce more clarity, but still, (5) is satisfied in the state the three pixels are exemplified before.

A clarification must be added with regards to what is being understood by saying „three adjacent pixels”. These three pixels will be the current pixel and the other two fill with color in Fig. 4. Please note that the three squares colored with Red, Green and Blue in the same figure are the three components of the same pixels reason why the same squares are first superposed in this figure. The squares are then represented separately to emphasize that a different transformation could be applied to each individual component i.e. Red, Green and Blue.

Fig. 6 shows all possible cases for the condition in (5c) to be met. First of all, a convention must be made regarding the three colors with which the squares, representing the pixels, have been

filled. Assuming that the less intense color (white) of the squares will be associated with the color (0,0,32), the light blue will correspond to the color (0,0,16) and the dark blue will be associated with (0,0,0), this leads to a better understanding of the condition in (5) which is fulfilled for this example. The example said before, states, in other words, that the value of the Blue component of current pixel is 32, the value of the Blue component of the pixel with the coordinates (x-1,y-1) is 16 and the value of the same component of the pixel with coordinates (x-2,y-2) is 0. All possible cases that fulfill the condition in (5c) are visually represented in Fig. 6.



Fig. 6: Possible cases for the condition in formula (5c) to be fulfilled

Before getting to the point where a specific transformation is applied, a new condition described by (6) must be first met. If the conditions in (5c) and (6c) are fulfilled, meaning that there is a diagonal gradient, then the same transformation as with Smart Contrast filter is being applied.

$$/ r_{x_2y_2} - r_{xy} / > 30 \quad (6a)$$

$$/ g_{x_2y_2} - g_{xy} / > 30 \quad (6b)$$

$$/ b_{x_2y_2} - b_{xy} / > 30 \quad (6c)$$

Please note that, just like in (6a-c), the same condition is being described for all the components i.e. Red, Green and Blue of the pixels in discussion in order to emphasize that for each of these components a different transformation could be applied depending on whether the conditions are met or not. By making the same assumptions as with Fig. 6, Fig. 7 spots the two of the six possible cases shown in Fig. 6 which meet the condition in (6c).

If the condition in (6c) is not fulfilled than the intensity of the Blue component will be increased or decreased depending on whether the condition in (7c) described below is met or not. To be more specific, if the condition is fulfilled, the intensity of the Blue component is decreased using the transformation in (2) and in the opposite case, the intensity is increased using (3), both regardless of the value of the Blue component. As for the example mentioned before in which the values of the Blue components of the pixels in discussion were 0, 16 and 32, the condition in (6c) was already fulfilled and as a result, the condition in (7c) is not tested anymore.

$$/ r_{x_2y_2} - r_{x_1y_1} / > 0 \quad (7a)$$

$$/ g_{x_2y_2} - g_{x_1y_1} / > 0 \quad (7b)$$

$$/ b_{x_2y_2} - b_{x_1y_1} / > 0 \quad (7c)$$

Making the same assumptions regarding color association as with Fig. 6, Fig. 8 spots the two of the 4 left possible cases shown in Fig. 6 (initially were 6 but two of them were already discussed and matched with the condition in (6c)). In other words, the two cases, shown Fig. 8, meet the condition in formula (7c) and are different from the ones shown in Fig. 7.

In case the condition in formula (7c) is not fulfilled, the two left cases that were not discussed before out of six spotted in Fig. 6, are shown in Fig. 9.

It was never said before what happens if the condition in (5c) is not met and this will be the appropriate time to be speaking about this. Well, if the condition is not fulfilled another condition is being tested and shown in (8c).

$$/ r_{x_2y_2} - r_{x_1y_1} / > 15 \quad (8a)$$

$$/ g_{x_2y_2} - g_{x_1y_1} / > 15 \quad (8b)$$

$$/ b_{x_2y_2} - b_{x_1y_1} / > 15 \quad (8c)$$

If neither this is fulfilled no transformation will be applied to the Blue Component of the pixel with coordinates $(x - 2, y - 2)$. If the condition in (8c) is met, the condition in (7c) explained before is tested again.

The visual representation of the possible cases that would meet the condition in formula (8c) is shown in Fig. 10.

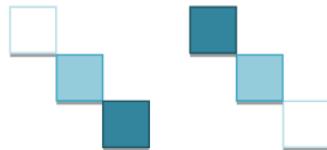


Fig. 7: Possible cases for the condition in (6c) to be fulfilled



Fig. 9: Possible cases while condition in (7c) is not fulfilled



Fig. 8: Possible cases for the condition in (7c) to be fulfilled

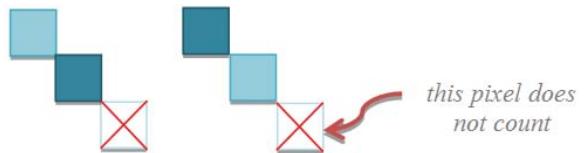


Fig. 10: Possible cases for the condition in (8c) to be fulfilled

As a final clarification before showing the code, it is true the Blue component was chose to explain better the algorithm, but it could have been either Red or Green. Also, the pixel with coordinates (x, y) was a randomly chose, the algorithm being applied for each component of each pixel in the image.

3.2.3 Visual results of applying Highlight image filter on images

Highlight filter produces the results in Fig. 11. It can be obviously seen how the filter sharpens the characters (the best example would be “ALL CHANNELS” image) and how it creates shadows behind the characters (e.g. the black shadows can be best seen on “Golf: Women’s British Open” image). In the filtered images the characters appear highlighted, which is the main visual effect of Highlight filter, beside sharpening the edges, increasing the contrast of the image and creating shadows behind characters.



Fig. 11: The visual effect of Highlight filter

4 Highlight image filter program code

Part of the C# code that corresponds to Highlight image filter is being listed in the following:

Code 1. A part of the Highlight filter algorithm

```
public void EdgeIntensityChange(byte[] buffer, double[] contrastBuffer, int Stride, int k)
{
    int diff01 = buffer[k - Stride * 2 - 8] - buffer[k - Stride - 4];
    int diff12 = buffer[k - Stride - 4] - buffer[k];
    int diff02 = buffer[k - Stride * 2 - 8] - buffer[k];
    // might be a diagonal gradient
    if (Math.Abs(diff01) > 15 && Math.Abs(diff12) > 15)
    {
        if (Math.Abs(diff02) > 30) // if there is a gradient
        {
            // increase or decrease the component depending on its value, if less than 127
```

The visual representation of the algorithm is shown in Fig. 12. The six cases next to the first `if` are the visual representation of the corresponding condition. In other words, if the condition in (5c) is met we will find ourselves in one of the six possible cases. So far for the rest of the figure, the possible cases that meet the conditions are shown next to the `if`-s and `else`-s, like it is also in the case of the next `if`, for which its condition is represented visually by two out of six possible cases.

Conclusions

I recommend the usage of Highlight filter rather than other filters in situations when the image contains areas of narrow text (but sure you can successfully use it also when the text in the image is wide). After applying this filter in the situation mentioned above, the rate of OCR success is considerably increased. Also an important thing to mention and probably the most important thing regarding Highlight filter would be that it determines a very good OCR rate of success when there is noise present in the image and more exactly, all around the characters in the image. The image could be also a bit blurred (not too much) and still it produces good results when OCR is performed on the filtered image.

In few words, Highlight filter determines outstanding OCR results on images with the following properties:

- Text (present in the image) is narrow
 - Noise is present (could be around characters)
 - Any other situation (e.g. lack of contrast, too much blurring)

The effect of Highlight filter is detecting the edges and once detected it sharpens them and increases their contrast. As a result, it highlights the edges. The visual effect on characters which are present in the image would be sharpening them and increasing their contrast, creating shadows (behind them) that contrast with character's color and obviously highlighting them.

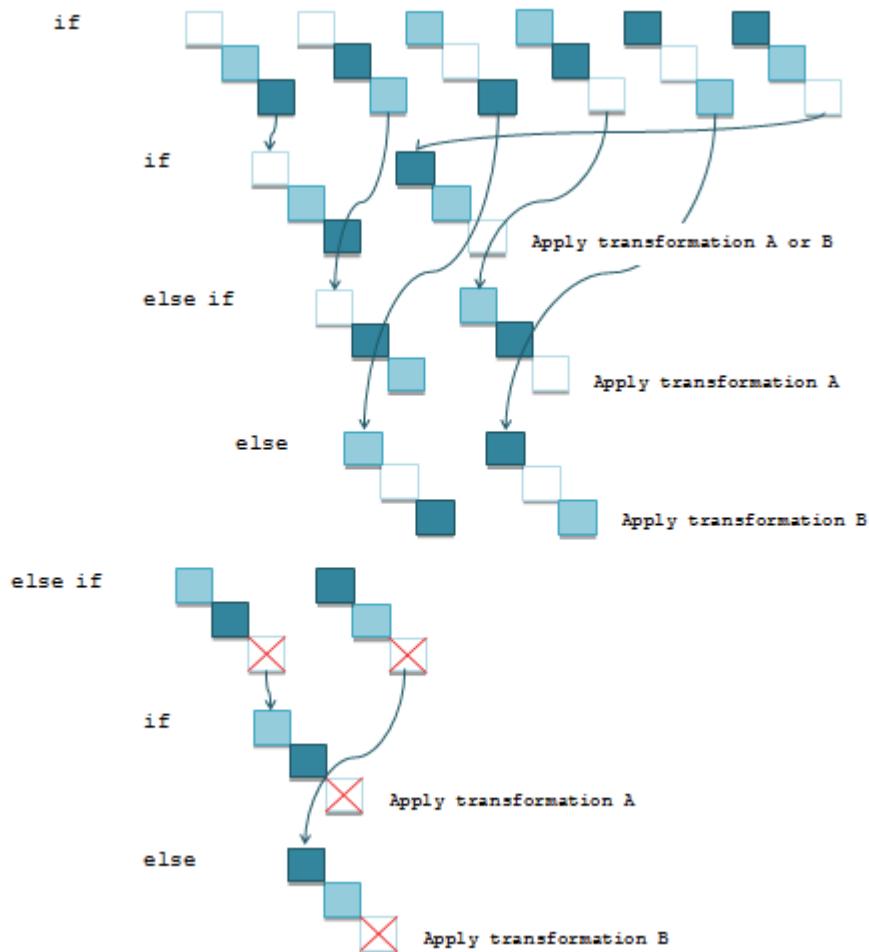


Fig. 12: The visual representation of the Highlight filter algorithm

In conclusion, the visual result of Highlight filter on characters is:

- *Sharpen*
 - *Contrast*
 - *Highlight*

Many techniques, such as adaptive restoring of text image, have been tried, as in [5]. Also, image filtering has also made an improvement in important areas, such as medicine, as described in [6], but lately improving OCR performance using filtering has gathered the focus and has become and will be a great challenge. Smart Contrast filter had as a starting point Contrast filter that was already implemented within Silicon & Software Systems Limited (S3Group) Company. Beside Smart Contrast's incontestable performances in improving OCR, this filter was actually just the triggering point for creating Highlight filter which overcomes the challenge of improving the OCR rate of success on text images.

Many of the existing image filters perform linear transformation as described in [7]. Lately, image filters are designed to perform non-linear transformations. Highlight image filter perform a non-linear transformation and even more, it is less time consuming (around 0.3 seconds for: 1920 x 1080 resolution images which are considered HD – High Dimension images, 32 bits per pixel).

References

- [1] A. Rajwade, A. Rangarajan, A. Benerjee, *Automated Filter Parameter Selection Using Measures of Noiseness*, University of Florida, Gainesville, United States, 2010
- [2] Tony Lindeberg, *Edge Detection and Ridge Detection with Automatic Scale Selection*, International Journal of Computer Vision, XXX, 117-156, 1998
- [3] *Enhanced Filters. Sharpen*, The GIMP Help Team (Graphics by Jakub Steiner), 2001 – 2002, [Online: <http://docs.gimp.org/en/plug-in-sharpen.html>]
- [4] Robert Fisher, Simon Perkins, Ashley Walker, Erik Wolfart, *Laplacian / Laplacian of Gauussion*, 2003, [Online: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm>]
- [5] P. Stubberud¹, J. Kanai, V. Kalluri, *Adaptive image restoration of text images that contain touching or broken characters*, ¹Nevada University, Las Vegas, United States, 1995
- [6] J. C. Barber¹, C. Daft², *Adaptive filtering for reduction of speckle in ultrasonic pulse-echo images*, ¹Institute of Cancer Research, Surrey, United Kingdom, ²University of Oxford, United Kingdom, 1996
- [7] I. Pittas, *Digital image processing algorithms and applications*, John Wiley & Sons Publishing, United States, 2000

Iulia STIRB
“Politehnica” University of Timișoara
Computer and Software Engineering Department
2, Piața Victoriei Street, 300006 Timișoara, România
E-mail: iulia_s24@yahoo.com

Comparison between the Optimization Techniques of Image Filters Used to Improve Optical Character Recognition

Iulia Știrb
Teacher Coordinator: Horia Ciocârlie

Abstract

Improving Optical Character Recognition (OCR) rate of success is one of the aspects in which image filtering is used. Altered shades and colors, increased contrast or texture, tones and special effects are some of the visual results of applying filters on images. Still the main purpose of image filtering is reducing the noise of the text images, which is the most important factor that influences the OCR rate of success.

The image filters described in this paper are as well used to improve a subsequent OCR within Silicon and Software System Limited (S3Group) Company from Dublin, Ireland. The OCR is part of the process of testing the menu options displayed on the output video of the set-top box under test.

This paper presents Contrast, Sharpen, Blur, Invert, Color and Highlight, the new image filter that I have developed and gathers the particularities of each of this filters. Next, the focus is on the optimization techniques and on the analysis of which better suits to be applied to each of this filters in order to make them perform faster. The optimization that produces the best improvement in terms of execution time for all filters above is done using a byte array to store the components i.e. Alpha, Red, Green, Blue of each pixel. Two other optimizations will be described, one using C# predefined `ColorMatrix` class for improving Contrast filter even more and the other by computing just once the filter weight for all pixels in the interior of the image in the case of Sharpen and Blur filters.

1 Introduction

The OCR of videos which involves first an automatic extraction of many frames (captures) in a second, as in [1] is proved to be very useful in digital television domain. Lately, in digital television, the need for a high number of OCR on images in a short time interval is increasingly common. The need for filtering the images before passing them to OCR is becoming more and more frequent because of the noise which appears in it. Since the OCR speedup must increase, the filtering speed must do as well. Thus, the optimization techniques in terms of execution time of Contrast, Sharpen, Blur, Invert, Color and Highlight filters will be presented the third section of this paper. The description of these filters will be included in the second section and the Conclusions section compares the results of the study of the optimization techniques' improvements in terms of execution time discussed in the paper.

What this paper brings new is a case study of which optimization technique suits well to each filter separately. I have carried out many tries of applying the optimization techniques on all filters. For instance, Contrast filter's most optimized version is the one in which the implementation is done using C# `ColorMatrix` class.

1.1 Image filters

Contrast filter increases the contrast in an image by modifying all components of the image (e.g. Red, Green and Blue) except Alpha. In fact, the Alpha component remains unchanged for all presented filters.

Sharpen filter accentuates edges in an image, by applying a fixed $N \times N$ dimension mask (also named kernel) to all pixels.

Blur filters reduces the details in an image by blurring it. This is done using a mask as well but with different values from the ones used for Sharpen filter mask.

Invert filter will apply that old look on the image similar to first TVs with black and white screens, except for the fact that the images will be coloured (assuming the original image isn't black and white only).

Color filter has a variety of forms. While some of them keep only the intensity of one component (e.g. Red) or two components (e.g. Green and Blue), others may also produce grayscale images.

Highlight is the new filter I have created and designed to be an universal filter since it improves OCR on a large variety of images. So, this filter is meant to avoid automatic selection of the filter described in [2], which is an expensive analysis.

1.2 Optimization techniques of image filters

All the optimizations proposed by this paper are measured in terms of execution time.

The first optimization is based on the usage of `Copy` overloaded methods from C# `Marshal` class, in order to copy the bytes of the image representation to a buffer on which the calculations are done afterwards. This avoids accessing the image each time a pixel needs to be set with the new filtered colour.

The second optimization is characterized by the usage of C# `ColorMatrix` class which allows software developers to apply different transformations to each pixel component separately (e.g. Red component of the image could be increased more than Green component).

The third optimization is more related to programs logic. Based on Sharpen and Blur filter implementation, I observed that the filter weight, which depends on the chosen mask, needs to be computed only once for the pixels in the interior of the image, avoiding its calculation for each pixel separately.

2 Analysis of image filters

2.1 Contrast image filter

Contrast filter increases the contrast according to a (1) which is applied to each component (e.g. Red, Green and Blue) of each pixel. Since the value of the components varies in a range from 0 to 255, the contrast scale, representing the degree to which the image is contrasted, is multiplied by a number that varies in an interval that is symmetric to zero ($[-0.5; 0.5]$ – full intensity of the component to no intensity at all). Afterwards, the value is scaled up again to the range from 0 to 255, by assigning, when necessary, the value 0 to less than 0 results and the value 255 to greater than 255 results of the computation in (1).

$$((\text{component} / 255.0 - 0.5) * \text{scale} + 0.5) * 255.0 \quad (1)$$

2.2 Sharpen image filter

Sharpen filter accentuates edges, but it does as well with the noise, as in [3], which is undesired and could make the OCR produce worse results than with the unfiltered image.

In order to compute the filtered pixel in the position (x, y) relative to the upper-left corner of the image, a mask of $N \times N$ dimension is applied to the original pixel. Since N is agreed to be an even number, the center of mask overlaps the original pixel (x, y) , as in Fig. 1 (for $N=3$). If we consider the mask to be represented as a matrix m of $N \times N$ dimension and the filter weight w which represents the sum of all elements of matrix m , then the value $p_{x,y}^f$ of the filtered pixel with coordinates (x, y) is computed as in

$$p_{x,y}^f = \left(\sum_{i=x-N/2}^{x+N/2} \sum_{j=y-N/2}^{y+N/2} m[i - x + N/2][j - y + N/2] * p_{i,j} \right) / w. \quad (2)$$

As described in (2), $p_{x,y}^f$ depends on the values of the original pixel (x, y) and on the neighboring original pixels that overlap with the mask. In other words, we sum the products between the value of one element in the mask and the value of the overlapping pixel. Equation (2) is used separately to compute the filtered value for each of the components of the pixel namely Red, Green and Blue, except Alpha. As a result, $p_{x,y}^f$ will actually have the form $(r_p_{x,y}^f, g_p_{x,y}^f, b_p_{x,y}^f, a_p_{x,y})$, in which $r_p_{x,y}^f$, $g_p_{x,y}^f$ and $b_p_{x,y}^f$ are the values of the components of the filtered pixel, while the Alpha component $a_p_{x,y}$ is the same as with the original pixel.

Equation (2) is applied to all pixels in the image, except for pixels located in the edges. Note that because the value of the filtered pixels depend also on the values of the neighboring original pixels, an extra structure is needed to keep the filtered pixels, in order to avoid altering the original pixels that are needed for further computations. If any of the original pixels would be altered, the neighboring pixels will be calculated according to the altered original pixels, which would be undesired for this filter since a mask has been already used to delimitate the dependencies of the filtered pixels.

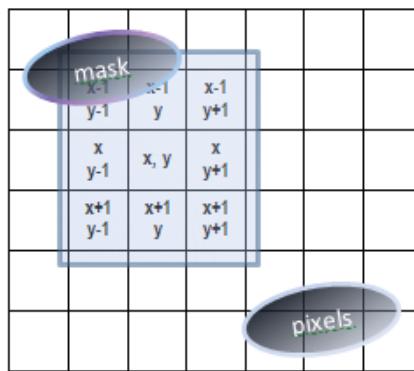


Fig. 1: A small area of the image and a 3×3 dimensions mask overlapping the current pixel (x, y) and the neighboring pixels

Two examples of masks are shown in Fig. 2. A common property of the two masks is that both are symmetrical in relation to the center, meaning that the neighboring pixels that are equally spaced from the center of the mask, influence to the same extend the value of the current filtered pixel.

2.3 Blur image filter

Also named Gaussian Blur, this filter reduces the noise in the image by performing a blur action on it. By blurring the image, the details are reduced. While OCR benefits from noise reduction, not the same thing can be affirmed when it comes to detail reduction. By blurring an image which contains cramped characters that are coloured almost the same as the background colour, OCR yields the worst results.

-1	-1	-1
-1	9	-1
-1	-1	-1

-1	-1	-1	-1	-1
-1	2	2	2	-1
-1	2	8	2	-1
-1	2	2	2	-1
-1	-1	-1	-1	-1

Fig. 2: Two examples of Sharpen filter masks

Regarding the implementation of Blur filter, it can be performed in the same manner as with Sharpen filter, except for the fact that the mask contains different values from the ones of Sharpen filter. Fig. 3 shows two examples of such masks.

1	2	1
2	4	2
1	2	1

1	1	2	1	1
1	2	4	2	1
2	4	8	4	2
1	2	4	2	1
1	1	2	1	1

Fig. 3: Two examples of Blur filter masks

2.4 Invert image filter

Invert filter transforms the original image, named positive image into a filtered image, called negative image, as in [1]. Each component, namely $r_p^f_{x,y}$, $g_p^f_{x,y}$ and $b_p^f_{x,y}$ of the filtered pixel $p^f_{x,y}$ in the negative image, is obtained by subtracting the value of each component, i.e. $r_p_{x,y}$, $g_p_{x,y}$ and $b_p_{x,y}$, of the original pixel $p_{x,y}$ from 255 as in

$$r_p^f_{x,y} = 255 - r_p_{x,y}, \quad (3a)$$

$$g_p^f_{x,y} = 255 - g_p_{x,y}, \quad (3b)$$

$$b_p^f_{x,y} = 255 - b_p_{x,y}. \quad (3b)$$

2.5 Color image filter

The generic Color filter includes a wide variety of filters that can be classified under this name. Table 1 contains the Color filters and a brief description of the action of each one on images. For instance, when Red filter is applied to the image, everything looks red and black.

Many of these filters lead to improved OCR rate of success that still, depends on many factors related to the colour content of the image such as the colour of the characters, the colour of the background around characters, the noise, the font of the characters and many others. Thus, Color filters do not have a high percentage of reliability on improving OCR on all kind of images.

As described in [4], Color filter with all its range of sub filters did find its applicability in television.

2.6 Highlight image filter

The effect of Highlight image filter is detecting the edges and once detected it sharpens them and increases their contrast. As a result, it highlights the edges. The effect on characters would be

sharpening them and increasing their contrast, creating shadows that contrast with their colour and obviously highlighting them.

Highlight filter is the best choice when the image contains area of text which is narrow (but sure you can successfully use it also when the text in the image is wide). After applying this filter in the situation mentioned above, the rate of OCR success is considerably increased. Probably the most important thing to mention is that Highlight filter improves OCR when there is noise present all around the characters in the image.

The image could be also a bit blurred (not too much) and still the OCR is improved.

Many techniques, such as adaptive restoring of text images, have been tried, as in [5] in order to improve a subsequent OCR. Highlight filter is suitable in fact for any kind of images, since it is a universal filter designed to improve OCR rate of success.

TABLE 1. View on the implementation guidelines of some of the Color filters

Curr. No.	Color Filter	Description
1	Red	Red component is left unchanged, the other two assigned to zero
2	Green	Green component is left unchanged, the other two assigned to zero
3	RedBlue	Red and Blue components are left unchanged, Green is assigned to zero
4	GreenGray	Same action as Green filter, in addition grayscale the result
5	RedBlueGray	Same action as RedBlue filter, in addition grayscale the result
6	Gray	Grayscale the image

3 Analysis of optimization techniques

3.1 Usage of a byte buffer

Initially, the byte buffer stores the byte representation of the original image. Blue, Green, Red and Alpha components are stored in this order in the byte buffer for each pixel of the image as in Fig. 4. The buffer elements are then used when computing each filtered pixel. Afterwards, the corresponding buffer elements are reassigned with the filtered values as in (4). Function f is the filter's transformation and k varies in a range from 0 to $4*w*h$, where w and h are the width and height of the original image. Since Alpha component remains unmodified, (4) is not applied for values of k for which $k \bmod 4$ equals 3.

$$buffer[k] = f(buffer[k]), k = \overline{0, N}, N = 4 * w * h \quad (4)$$

An additional cloned buffer is required for Sharpen and Blur filter to avoid altering the original pixels.

Thus, in the end, the byte buffer will contain the byte representation of the filtered image.

Regarding the storage of the image itself, both the original and the filtered images are encapsulated in C# Bitmap objects. The full description of the Bitmap class and many useful others can be found in [5].

First step in the algorithm is to obtain an object of type `BitmapData`. C# `LockBits` method of `Bitmap` class returns the desired object.

In next step `Scan0` property is accessed on the obtained object in order to get the address of the first pixel in the image. Once we have this address, all the image data is copied to the byte buffer using the overloaded `Copy` method of C# `Marshal` class.

The third step is specific to each filter separately and represents the calculations done formally described in (4). By using this byte buffer, we avoid calling `SetPixel` method on a `Bitmap` object each time the filtered pixels needs to be set, which would be time consuming since the

Bitmap object is accessed as many times as the number of pixels in the image. Thus, a major execution speedup is measured by performing the calculations on the byte buffer structure. The next step is to copy the filtered buffer back to the address of the first pixel of the image using the overloaded Copy method.

The last step is to unlock the Bitmap object, using `UnlockBits` method.

This optimization technique produces the best results for filters such Invert, Color, or even Sharpen and Blur, which need an extra buffer. All of these have in common the fact that the calculations on the byte buffer are less than the ones necessary for Contrast filter described in (1). The next optimization technique performs even faster than this one in case of Contrast filter since the number of calculations is less. Details follow.

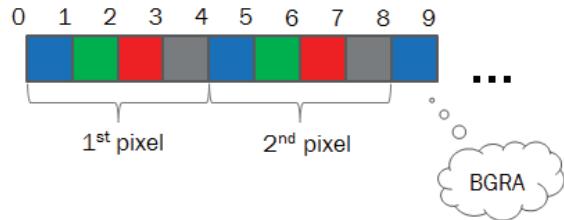


Fig. 4: Storage in byte buffer of image representation i.e. Blue, Green, Red and Alpha components in this order for each pixel

3.2 Usage of ColorMatrix class

The first step when using this technique is to declare a matrix of 5×5 dimension as exemplified in Fig. 5.

As in [6], the input vector represents the original pixel with its components Red, Green, Blue and Alpha (e.g. in Fig. 5 the Red component has the value 0.2). The values of the components vary in a range from 0 (0 / 255) to 1 (255 / 255). While zero is the equivalent of 0, one is the equivalent of 255. The order in which the components are stored in the vector is Red, Green, Blue and Alpha. The value in the last position in the vector represents a factor used for additional computations which may be needed (if not, it can be left zero). The output array represents the multiplication result.

Note that for each original pixel an input vector is created and the values of the pixel's components are scaled to fit into the interval [0; 1]. Then, again, after the calculation of the output array, the values are scaled to fit into the interval [0; 255].

The matrix for Contrast filter is calculated based on (1) and exemplified in Fig. 6. *diff* is computed as in (5) and required in the additional computations.

$$diff = -0.5 * (scale - 1.0) - 0.5 / 255.0 \quad (5)$$

The reason for the occurrence the term $-0.5/255.0$ is to eliminate in many cases as possible the undesired rounding which appears on double to float conversions, since the color matrix contains float values and *diff* and *scale* are double values.

In order to compute the filtered value of one component of a pixel, “byte buffer” technique performs two multiplies, one division, one addition and one subtraction according to (1), while `ColorMatrix` technique performs just one multiply, one addition, a division and a multiply required for scaling operations described above. The speedup when using `ColorMatrix` compared to “byte buffer” technique is recorded only for Contrast filter and it comes from the lack of an additional subtraction every time the new value of a component of a pixel is computed. Thus, Contrast filter performs *x* seconds faster than optimized with “byte buffer” technique, where *x* is the product between the average subtraction time *subtractTime*, the number of components *no_comp* (equal to 3 i.e. Red, Green and Blue, except Alpha) and the number of pixels *no_pixels*.

$$x = subtractTime * noComp * noPixels \quad (6)$$

$$\begin{bmatrix} 0.2 & 0.7 & 0.5 & 1.0 & 1.0 \end{bmatrix} \times \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0.5 & 0.1 & 0.2 & 0 & 1 \end{pmatrix} = \\ = \begin{bmatrix} 0.9 & 0.8 & 0.7 & 1.0 & 1.0 \end{bmatrix}$$

Fig. 5: Color matrix example of multiplication for one pixel

scale	0	0	0	0
0	scale	0	0	0
0	0	scale	0	0
0	0	0	scale	0
diff	diff	diff	diff	1

Fig. 6: Color matrix for Contrast image filter

3.3 Computing filter weight once for all pixels in the interior of the image

The third optimization is described by the calculation of the filter weight, which is, again, the sum of all elements in the mask, just once at the beginning of the filtering algorithm. Since this optimization implies the usage of a mask, it is applied only to filters which use it as Sharpen and Blur filter do.

Fig. 7 shows an example of a small image and a 3 x 3 dimension mask which overlaps different areas of the image, one at a time. Pixels 1 and 2 are considered to be in the interior of the image since the mask doesn't get out of the bounds of the image. Pixel 3 is considered to be in the edges of the image because a part of the mask gets out of the edges of the image.

In the case of pixel 3, the elements of the mask which are out of the bound of the image are not summed when computing the filter weight.

The filter weight for pixel 2 and 3 is the same, so is for the other pixels in the interior of the image. For this reason, the filter weight needs to be computed only once for this pixels. For the rest of the pixel, which are located in the edges of the image, filter weight is calculated each time a filtered pixel is set.

Sure, we can be more meticulous and expect to have the same filter weight for the pixels located, for example, in the first column of the image except for the upper-left and bottom-left corner. For these pixels, the filter weight would need to be computed only once, but this does nothing else than to induces extra computation (i.e. additional `if`'s) and does not produce any speedup, so I advice limiting the granularity of filter weight calculation to the two main ways of calculating it described before: the same filter weight for the pixels located in the interior and separately computing the filter weight for each pixel in the edges of the image.

This optimization produces good results on small images, such as 30 x 20 pixels. Since usually OCR is performed on this kind of images, this technique proves to be very useful when it comes to speed up the execution time of Sharpen and Blur filters.

3.4 Comparison between the three optimization techniques

While on some of the filters, two of the optimization techniques presented in this paper can be applied, on other filters, just one of it can be implemented. Furthermore, the optimization

technique using a byte buffer and the one using `ColorMatrix` cannot be applied together on the same filter. Since using `ColorMatrix` implies having to perform the operation on the `Bitmap` object that encapsulates the image, using also a byte buffer would make no sense.

Table II gathers the improvements in terms of execution time made on the image filters using the optimization techniques presented in this paper.

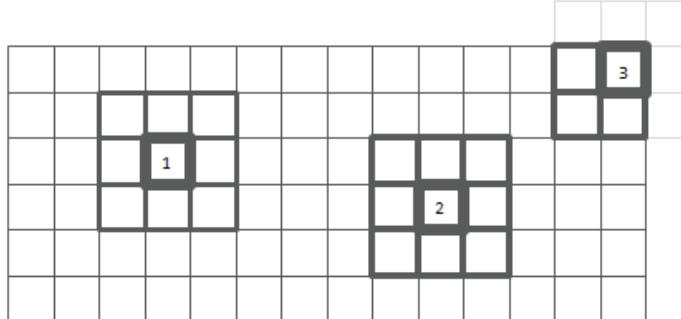


Fig. 7: Possible cases in which the mask overlaps pixels exemplified on a small image

TABLE 2. Comparison between the optimization techniques improvements in terms execution time of the image filters

Curr. No.	Filter	Optimization technique used			Speedup
		Byte Buffer	Color Matrix	Filter Weight	
1	Contrast	-	✓	-	8.7 times
2	Sharpen	✓	-	✓	4.7 times
3	Blur	✓	-	✓	4.7 times
4	Invert	✓	-	-	53 times
5	Color	✓	-	-	42 times

3 Filters program code optimized using byte buffer technique compared to the initial implementation

Code 1. Image filters implementation before being optimized

```
Bitmap clonedBitmap = new Bitmap(originallImage.Width, originallImage.Height);
UnsafeBitmap bitmap = new UnsafeBitmap(originallImage);
bitmap.LockBitmap();
...
// for each pixel with coordinates (x, y) set the new filtered value
clonedBitmap.SetPixel(x,y,Color.FromArgb(r,g,b));
...
bitmap.UnlockBitmap();
return clonedBitmap;
```

Code 2. Image filters optimized implementation using byte buffer technique

```
FilteredBitmap filteredBitmap = new FilteredBitmap(originallImage);
byte[] buffer = filteredBitmap.LockBitmap();
...
// here is the code which processes the buffer elements obtaining the filtered values which are then
// stored in the buffer, this part is specific to each filter separately
...
// set the new value of all pixels, by accessing the image only once
filteredBitmap.SetPixels(buffer);

filteredBitmap.UnlockBitmap();
return filteredBitmap.Bitmap;
```

In Code 1, the method which is used to set the colour of the each pixel i.e. `SetPixel` is called on the object which encapsulates the image representation i.e. `clonedBitmap` of predefined C# type `Bitmap`. The call happens for each pixel in the image and it is time consuming even for small size images. A reason for the high execution time could be the assumption that the image representation is entirely loaded in memory each time a pixel needs to be set with the new colour (r, g, b).

The code which is optimized using byte buffer technique is presented in Code 2. `FilteredBitmap` is the class I have implemented which hides all the details regarding:

- The way pixels are get from the `originalImage` object of type `Bitmap` and stored in the byte buffer as shown in Fig. 4 and described in Code 3;
- How pixels in the resulting (filtered) image are set to the new values that are get from the byte buffer which contains them. Program code for this can be found in Code 4.

By using this technique, the image will be accessed only once for setting all pixels with the filtered values, considerably improving the execution time (e.g. average improvement for all filters discussed in this paper is 22 times).

Code 3. Storing the original image representation in the byte buffer (methods from `FilteredBitmap` class)

```
public byte[] LockBitmap()
{
    // get the bounds of the image which will be further locked
    Rectangle bounds = GetBounds();
    // lock the filtered bitmap in order to obtain the pointer to the first pixel in the filtered image
    bitmapData = bitmap.LockBits(bounds, ImageLockMode.WriteOnly,
                                 PixelFormat.Format32bppArgb);
    // set the pointer to the first pixel in the filtered image which will be further used when setting the
    // new values of the pixels of the filtered image
    ptr = bitmapData.Scan0;
    // store the pixels in byte buffer attribute
    GetPixels(bounds);

    return byteBuffer;
}
public byte[] GetPixels(Rectangle bounds)
{
    BitmapData sourceBitmapData = sourceBitmap.LockBits(bounds, ImageLockMode.ReadOnly,
                                                       PixelFormat.Format32bppArgb);
    // set the pointer to the first pixel in the original image
    IntPtr ptrSource = sourceBitmapData.Scan0;
    byteBuffer = new byte[sourceBitmapData.Stride * sourceBitmap.Height];
    // copy the entire image representation in the byte buffer
    Marshal.Copy(ptrSource, byteBuffer, 0, byteBuffer.Length);
    // unlock the original image
    sourceBitmap.UnlockBits(sourceBitmapData);
    sourceBitmapData = null;

    return byteBuffer;
}
```

Code 4. Setting the byte representation of the filtered image (method from `FilteredBitmap` class)

```
public void SetPixels(byte[] buffer)
{
    // copy the entire content of the byte buffer to the address of the first pixel of the filtered image
    Marshal.Copy(buffer, 0, ptr, buffer.Length);
    byteBuffer = null;
}
```

Conclusions

Filters can be combined in many ways to improve a subsequent OCR on the image. The combined action of filters can be done by applying them one after another. Thus, the filtered image could be the input of another filter and the result could be more powerful with respect to the rate of success of the OCR. As in [7], this rate of success also depends on whether the characters are broken or they touch each other. If there is a lot of noise in the image, a Blur filter is the proper solution, afterwards followed by Color filter.

Figuring out and implementing the proper optimization technique for each filter depends on the particularities of the filters:

- The usage of a byte buffer is the technique with the largest applicability over image filters, since most of the filters perform a reduced number of computations per pixel, which is desired when using this optimization technique. This technique could also be useful when large number of computations are carried out by the filter but the optimization would not be as higher as for the filters with perform a small number of computations
- *ColorMatrix* is the most desired solution when the filter performs a lots of calculations, more than the fixed number of calculation done using this technique i.e. Contrast filter , but cannot be applied for the filters for which the filtered pixels depend on the neighboring original pixels
- Single calculation of filter weight is applicable to all filter that use a mask, thus, in situations when the filtered pixel depends on the neighboring pixels

As the underlying mathematical transformation of each filter is less complex than other, the improvement will be higher (see TABLE II in which Contrast image filter is improved only 8.7 times compared to Invert image filter which is improved 42 times from the original implementation that used C# SetPixel method).

References

- [1] T. Sato, T. Kanade, E.K. Hughes, M.A. Smith, *Video OCR for digital news archive*, Carnegie Mellon University, Pittsburg, United State, 1998
- [2] A. Rajwade, A. Rangarajan, A. Benerjee, *Automated Filter Parameter Selection Using Measures of Noiseness*, University of Florida, Gainesville, United States, 2010
- [3] “The Invert Photo Negative Filter”, Microsoft, 2014, [Online: <http://msdn.microsoft.com/en-us/library/cc294672.aspx>]
- [4] Edmund Jack Ambrose, Derek John Pacey, Joseph Gerard Salmon, *Color Filter*, Electric & Musical Industries Limited, Hayes, Great Britain, 1946
- [5] “Bitmap Class”, Microsoft, 2014, [Online: [http://msdn.microsoft.com/en-us/library/system.drawing.bitmap\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.drawing.bitmap(v=vs.110).aspx)]
- [6] “ColorMatrix Class”, Microsoft, 2014, [Online: [http://msdn.microsoft.com/en-us/library/system.drawing.imaging.colormatrix\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.drawing.imaging.colormatrix(v=vs.110).aspx)]
- [7] P. Stubberud¹, J. Kanai, V. Kalluri, *Adaptive image restoration of text images that contain touching or broken characters*, ¹Nevada University, Las Vegas, United States of America, 1995

Iulia STIRB
“Politehnica” University of Timișoara
Computer and Software Engineering Department
2, Piața Victoriei Street, 300006 Timișoara, România
E-mail: iulia_s24@yahoo.com

3D Modelling of a Restaurant

**Marius-Alin Tăpordei
Teacher Coordinator: Dana Dânciulescu**

Abstract

This paper is about to describe the 3D Modelling Design technology of a restaurant. You will see how can be created a 3D Model starting from an AutoCAD plan, using 3Ds Max. The final changes are made with Photoshop.

1. Introduction

Nowadays, 3D is a very important and necessary part of the Technology. Everything we see and create is 3D.

Games, publicity, industry, every domain uses 3D Modelling. For example, if we want to see how a neighbourhood will look like after the construction, we will need to make a 3D presentation of it. Companies like Apple, Samsung and other devices brands also use 3D models to present their new concept.

Creating a game also include a very large part of 3D. Characters, background and also other objects are created in 3D. Programs such as 3Ds Max, V-ray, Maya, Unity 3D, try hard to keep up to date the technology and satisfy the user's needs.

As working as a freelancer, there are a lot of projects from different domains that involve collaborating also with architects and programmers.

As a start point, a 3D model can have sketches, Auto-Cad or CAD files, blueprints, or a simple idea. The most interesting and pleasing part is that creating a 3D model you can see how the ideas come to reality. You can imagine everything you want and by modelling it you can finish and finally have an object that other people can see. You can make movies, animations or games that others can see and enjoy it. It's a pleasure to see how you can make people smile and having fun with your creations. The industry of gaming is very developed in our century and companies like Ubisoft, EA Games and Gameloft are very known worldwide for their advanced 3D Graphics technologies. More recent are the applications for Android and IOS that also use 3D technology.

Beside the gaming and entertainment part is also important the Architectural domain. Many countries have a severe legislation for buildings. To build something, you have to respect the architectural surroundings. For this, and to demonstrate it, also to see how a building will look greatest, many people want to make a 3D model. 3D also is useful for interior design, to see how a house, a restaurant or an office building will look like and what changes can be made.

In the sections below you can see how a plan or sketch become reality and the process of creating a 3D model. In the first section is presented the start point, what is important when you start a project and what do you need to start.

The second section will show the technologies that can be used for a 3D project and the programs such as 3Ds Max and V-ray. In the third section you can see what is the first step for starting the modelling of the 3D restaurant and the last section will show the final project and how it looks like, and also some screenshots.

2. 3D Software

In 3D computer graphics, 3D modeling is the process of developing a mathematical representation of any three-dimensional surface of object (either inanimate or living) via specialized software. The product is called a 3D Model. It can be displayed as a two-dimensional image through a process called 3D Rendering or used in a computer simulation of physical phenomena. The model can also be physically created using 3D printing devices. (1)

Most commonly used programs for 3D Modelling are:

- Anim8or
- Blender
- CityEngine
- LightWave3D
- Rhinoceros 3D
- Sketchup
- Autodesk 3ds Max
- Autodesk Maya
- Blender
- ZBrush

We are using 3Ds Max, because is very complex.

3ds Max® 3D modeling software provides a comprehensive modeling, animation, simulation, and rendering solution for games, film, and motion graphics artists. 3ds Max delivers efficient new tools, accelerated performance, and streamlined workflows to help increase overall productivity for working with complex, high-resolution assets. (2)



Fig 1. 3Ds Max interface

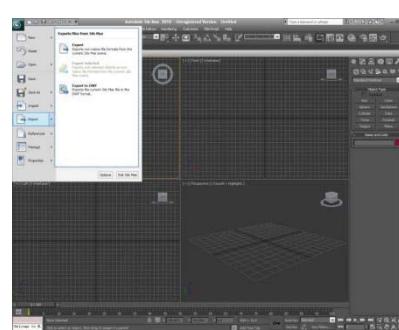


Fig. 2. 3Ds Max main menu

After the 3D modelling which is the most important and massive part, the final retouch can be made with Photoshop. Luminosity, Colours, Brightness can be changed by Photoshop. Also, you can delete parts that you don't like or you can merge multiple photos to make a final scene. Photoshop is also used for Texturing.

Texture mapping is a method for adding details, surface texture, such as bitmap or other image, or color to a character or object. This process is about applying a patterned paper to a shape. V-ray is also an additional software for 3Ds Max, used for a high-quality rendering and a fast rendering.

V-Ray for 3ds Max is the core development of Chaos Group, which allows users to quickly and easily create realistic images while giving them full control over the 3D production process. V-Ray 3.0 for 3ds Max is the most complete lighting, shading and rendering toolkit on the market, making speed and simplicity accessible to all artists. The latest version delivers powerful feature set, technical advances and support for open source technologies.(3)

3. Starting the 3D project of the restaurant

When starting a project, it is very important to set up the way of working. Different types of domains have different ways of working. For example, if you want to develop a game for mobile devices, such as phones or tablets, and the device have low resources and is not that powerful in its technology, you have to work more structured and to have a low number of polygons and the final project must be low – sized and optimized. If you develop an application for a high performance device, the game can be high – dimensioned and have a high – poly modelling type. You can start from sketches, CAD or AutoCAD files.

Also it is important if you want a realistic model, a photo-realistic, cartoonish or gaming model. When you make a realistic model, the final result can be easily confused with a real photograph. This type can be used for architectural presentations, movies, or the most commonly used is for devices. Before launching on the market, brands make a 3D prototype to make public how the device will look like.

Cartoonish style is very used in the film industry. The best known companies are Pixar (that made the first computer animated film and are the creators of Cars, Toy Story, Finding Nemo, Wall-E), Cartoon Network Studios, DreamWorks Animation (creators of Kung Fu Panda, Shrek, Madagascar) and Walt Disney Animation Studios, with a lot of creations, most recently being Frozen.

Gaming models are used, as the name says, for games. A game is made with programming and 3D modelling. While the programming uses a code and is the non-see part of a game, it is used as a brain for the game, the 3D is the seen part. With the code, you can learn the game what to make in different situations and structure it by levels. By modelling, you create the characters, their objects (such as weapons, munitions), the objects the main character interact with (such as animals, cars, airplanes) and also the background object (such as grass, trees, maps or buildings). Companies that succeeded in gaming are Ubisoft (the creators of Assassin's Creed, Bomberman, Brothers in Arms, and other), Gameloft (the creators of Prince of Persia, Rayman, and other), EA Games (creators of Need for Speed, Battlefield, The Sims, Fifa, and other).

After setting the type of model you need, you have to upload in the software you work, the plans, in case you have it. For example, for an architectural project you will need the building plan in AutoCad.

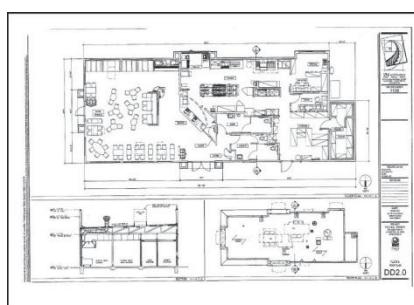


Fig. 3. AutoCad plan of the restaurant

Because the restaurant is going to be a real one, we will use the realistic modelling. In this case, we are talking about a fast-food restaurant with a Moroccan specific. After loading the AutoCAD plan in 3Ds Max, we can start modelling the walls, then put the windows and finally add the texture and polish up with Photoshop. This will be presented in the next Section.

First, for loading the AutoCAD plan in 3Ds Max, we have to import it to have a perfect scale modelling. The restaurant must have the same dimensions and for this we will work in centimetres and we will use its real dimensions. For example, the walls are having 3.5 meters tall.

4. Final scene of the restaurant

Starting with the AutoCAD plan presented in Fig. 3, we've modelled the fast-food restaurant named Moroccan Dish. In the figures below it can be seen how the process of creation evolved. First stage is to build walls and have the main body of the building and add some shadows and light. You can see that in the initial phase, the restaurant isn't looking very good.



Fig.4.The basic shape of the restaurant



Fig. 5. Restaurant after adding some lights

In Fig. 4 you can see that we don't have a texture, and we only formed a shape of an object. Basically, the 3D model is only formed from points called vertices that define the shape.

Fig.5 represents the restaurant after adding some lights. To see more detailed the shape, it is important to add some daylight or any kind of light we want. For this, we use V-ray, which is an extension of 3Ds Max. It is very useful for rendering realistic architecture. The software is very complex and we can give a large example of lights that can be used in V-ray: single-sided, double-sided lights, real-world lights with falloff, skylight, self-illuminated panels, bulb lights and the list can go on.

For the restaurant, we used realistic Skylight to give the object a good illumination and shadows. After building the basic shape, next step is adding the texture. This process also involves Photoshop because the texture can be created using it. For the restaurant, we used a brick texture for the roof, and for walls we used plaster. For windows, the texture used is glass.



Fig. 6. A scene that shows the restaurant at night

Depending on the demand and what is the final purpose of the modeling, we can use different lights. In our example, we wanted also to see how the restaurant will look like at night. For this, we changed the illumination also with V-ray. For exterior, we used moonlight, changing the intensity and color, and for interior, to give a realistic aspect, we used bulb light.

The final step is to add parts like background using Photoshop and also change the brightness and colors to have a perfect picture of the final scene. You can also add flowers, trees and sky using Photoshop and change the daylight. This part is called Rendering, which involve placing the objects in a scene. Rendering means converting a model into a image.



Fig.7. The restaurant after adding some bushes and trees in Photoshop



Fig.8. Night see of the restaurant



Fig. 9. The final scene of the restaurant on daylight

5. Conclusions

As a conclusion, we can see that programs like Photoshop and 3Ds Max are completing each other and can contribute to a good and high-quality final scene. That's why for the restaurant we used 3Ds Max for modelling and make the final changes with Photoshop.

We choose 3Ds Max because it is the most complex Software and has a good and pleasant interface. To have a complete and perfect final, Photoshop is irreplaceable. With brushes, lightness or by adding objects, you can make a perfect project down to the smallest details.

After modelling and making the final changes with Photoshop, we can see that the final scene is totally changed and is better.

References

- [1] http://en.wikipedia.org/wiki/3D_modeling
- [2] <http://www.autodesk.com/products/autodesk-3ds-max/overview>
- [3] <http://www.chaosgroup.com/en/2/vray.html>

Accessing date: 05.05.2014

Marius-Alin TĂPORDEI
University of Craiova
Department Informatics
Craiova, St. A. I. Cuza No.13
Romania
E-mail: alin.tapordei@yahoo.com

Real-time palm detection and animation

**Vasil Tsolov
Teacher Coordinator: Milena Lazarova**

Abstract

The paper is aimed at real-time hand palm detection and tracking. The process of palm detection is described and implemented as an algorithm based on background subtraction, skin detection, finding the biggest skin contour and points of interest on it. The palm detected is then used for real-time gesture driven animation of the human hand. The developed solution can be utilized in video games as well as other applications that can take advantage of such augmented reality gesture-based human computer interaction.

1 Introduction

Nowadays computer graphics and computer vision are developing rapidly and are being used in many different areas for a lot of purposes. The entertainment sector is one of the main reasons for the constant development and improvement of computer graphics algorithms, platforms and engines. The video games are typical example of the intensive usage of computer graphics for modelling the game characters, creating the game content and the game world. In order to animate and move the characters in more realistic and natural way "the human factor" is also used in the games in the last few years. The realistic animation of the character models is achieved using motion capture which is a process of recording human movement. The motion capture method requires sensors and markers to be put on real human actor's body, and then to track the movement of the sensors and markers when the actor moves and performs specific actions. The data gathered from the markers are later processed and are used to map and guide the animations in the game [1].

Another approach for human guided animation of digital characters models is to use computer vision methods for recognition and tracking of human body parts. In this case, simple webcam can replace the usage of specific sensors and markers. The later approach allows for real-time movement tracking and animation compared to the offline processing of data gathered with the motion capture approach.

The paper describes the process of real-time hand palm detection, tracking and animation. Based on the video grabbed by a web camera the human hand is recognized and animation of a selected model is applied on it. The processing stage involves background subtraction, skin detection and palm detection. The palm movement is then tracked so that the animation is

following that movement and is also adapted to some specific hand gestures. The purpose of the palm recognition and tracking is to achieve results similar to the motion capture based animation that is used in video games and movies. However the disadvantage of the latter approach is the offline processing that requires data for the human body movement to be first recorded and processed and used for the animation control. The suggested approach doesn't involve extra markers on the human hand and allows for real-time animation adoption based on the hand movement. In order to implement the described augmented reality perception a software application is developed in C++ using OpenCV library [2].

The paper is divided into two main parts. The first part describes the recognition processing stage. It includes the background subtraction, skin detection, finding the biggest skin colored contour, finding points of interest such as convex hull and convexity defects and limiting the region of interest only to the palm in case the whole arm is visible. The second processing stage is the gesture-based augmented reality. It involves loading and animation of some predefined model, tracking the palm movement and enhancing the model animation according to the palm gestures.

2 Gesture driven animation of the human hand

The hand gesture-based texturing and animation is divided into two main stages: palm detection and palm animation. The two stages and the processing steps they involve are shown on Fig. 1.

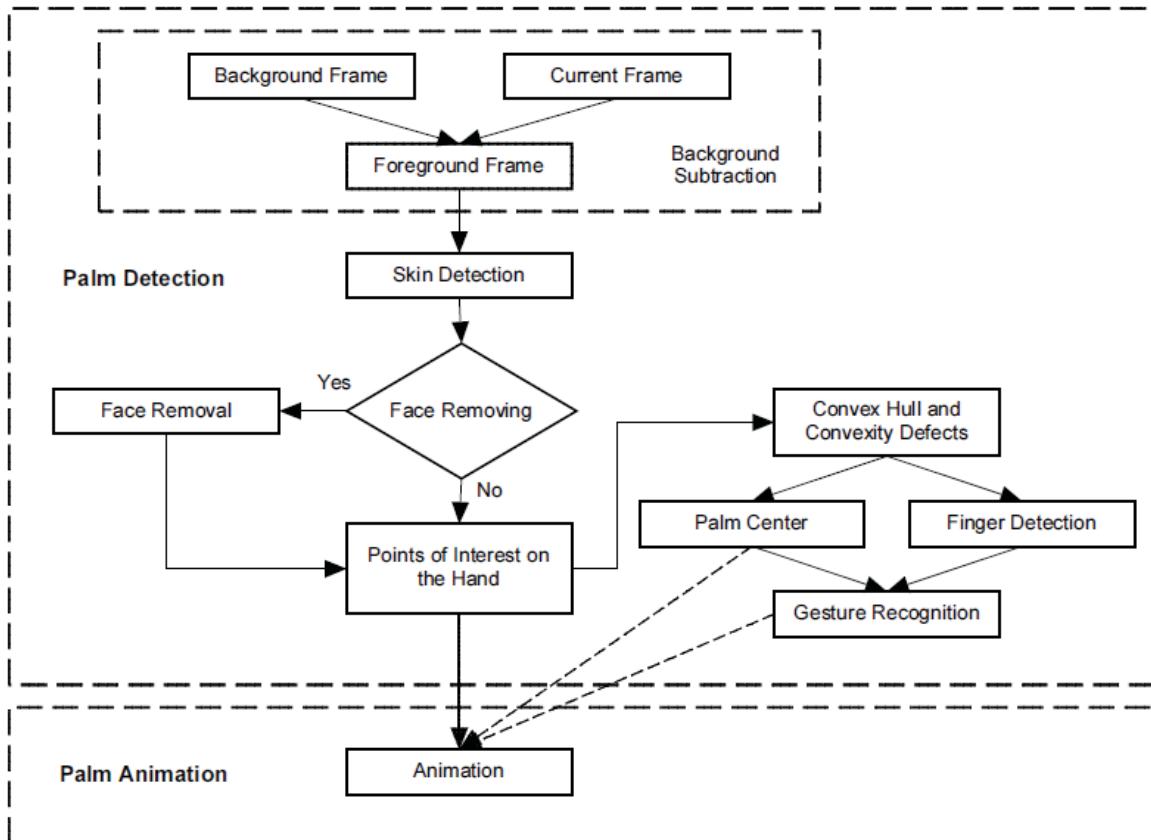


Figure1. The algorithm stages and processing steps.

2.1 Palm detection

The first processing stage is the palm detection. It requires the palm position in the current video frame to be discovered. For that stage the processing steps include background subtraction, skin detection and palm detection.

2.1.1 Background subtraction

In order to achieve real-time detection, recognition and analyses, steps are only performed on the human hand in the frame. Thus, the first step is background removal so that only the foreground that contains the palm to be processed at the next step. In the described approach color image is perceived by the camera and the only requirement for the background is to be static (Fig.2).



Figure 2. Background image.

There are several possible approaches for background detection and removal. The described solution uses a variant of background subtraction that is based on subtraction of the background frame and the current frame (Fig. 3). The two frames are decomposed into the three channels sensed by the camera and then the corresponding channels of the two frames are compared [3]. Thus, the effect of shadows is minimized and the errors that can be introduced in case the human hand is close to the surrounding objects, or the wall of the room, are removed.

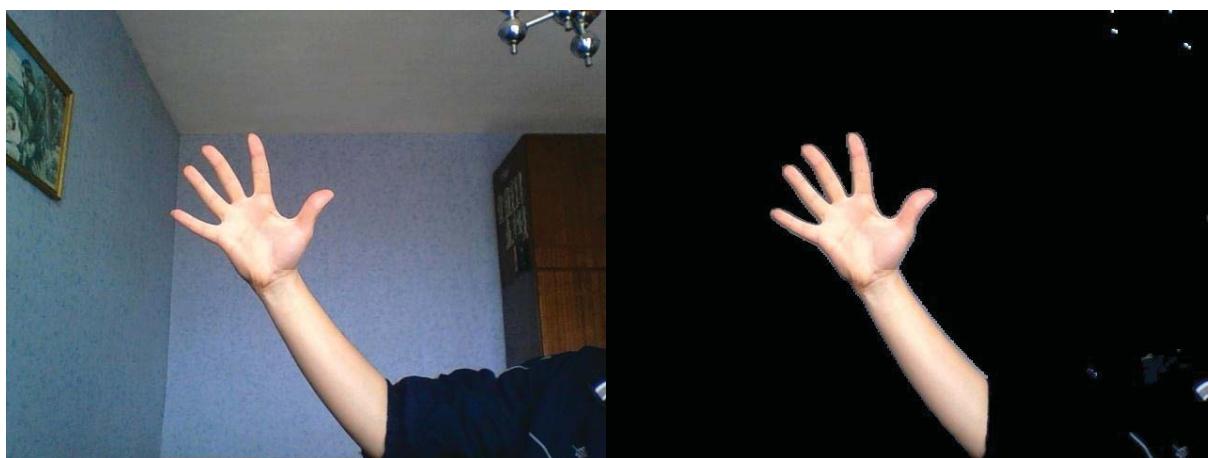


Figure 3. The current frame (left) and the extracted foreground (right).

2.1.2 Skin Detection

The next processing step requires detection of skin colored blobs in the foreground frame. For the segmentation of the skin blobs a color filter is used, that only retains the pixel in the foreground frame, which values fall in the range of predefined skin color values.

In the described solution the skin filter uses YCbCr color space. As a result only the regions that contain skin colors are passed to the next processing and analyses steps (Fig. 4).



Figure 4. Skin detection and subtraction: before (left) and after (right).

2.1.3 Biggest skin contour

After the skin colored parts are separated, there is one more processing step which results are then used in the following analysis. This step is aimed at the removal of erroneous skin pixels. Such "noise" pixels found at the skin detection step are pixels which have colors in the defined skin range, but are not really part of skin blobs. Furthermore, there might be some skin colored regions that occur in the new frame unintentionally. An example of the latter is a part of the user's face that might be seen in the frame when the user moves the hand palm. In order to locate and remove that type of errors, it is recommended to extract and focus only on the biggest region in the frame, which has the required skin colors. Using the inbuilt OpenCV function for finding contours in the frame, all contours are detected and the biggest contour is found.

However, sometimes the biggest skin contour might not be the one corresponding to the hand. For example, if both the user's face and hand are present in the current frame, the biggest skin contour can be assigned to the face instead of hand. That is why a face detection step is added to the processing pipeline as an option. The face detection is based on AdaBoost algorithms using Haar wavelet features as it is implemented in OpenCV library [4]. In case a face is detected in the frame, it is blackened by its bounding rectangle and is removed from further processing.

2.1.4 Analyzing the biggest skin contour

Once the biggest skin colored region in the frame is separated, the analysis of the points of interest is the next processing step. First, the bounding rectangle of the biggest skin region is found, as it defines the global position of the whole hand. After that, the convex hull and convexity defects of the hand are determined. The latter are particularly important for finding the palm and its center. In order to avoid some random convexity defects, only those with depth bigger than 20 are accepted. The bounding rectangle, the convex hull, the hand contour and the convexity defects are shown on Fig.5 as red rectangle, purple lines, green contour and yellow points respectively.

The next step requires detection of the hand palm itself. In order to distinct the palm from the arm the features found at the previous step are used. The goal is to determine the palm position and its size in the frame, so that to be able to use these parameters for the next stage of palm animation. The palm position and size will be represented as a circle with center and radius determined by the palm points of interest. In order to estimate these parameters the positions of the minimum enclosing and the maximum inscribed circles are used.

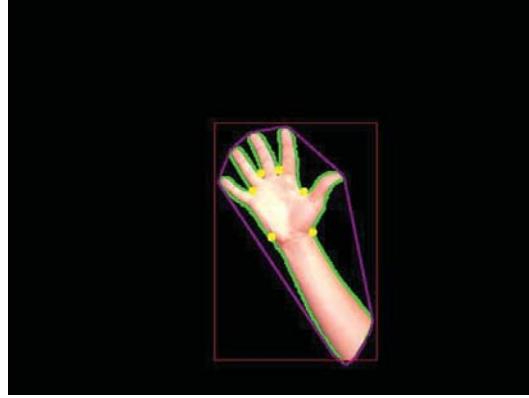


Figure 5. Points of interest on the hand.

2.1.4.1 Outer circle

First a circle is defined using some points of interest on the hand. The minimum enclosing circle of the convexity defects is calculated and is called an outer circle as suggested in [5], because it is not entirely in the hand contour (Fig. 6). However, if one of the convexity defects corresponding to the palm contour is not determined correctly and slips down the hand, the outer circle will enlarge and will not give correct palm position and size. Therefore, in order to avoid such kind of errors and ensure that the outer circle is stable and does not change abruptly in the adjacent frames, another circle parameters are estimated based on an average values of the detected palm feature points. The new circle's center is calculated as an average of all convexity defects and its radius is the average distance between the center and those convexity points. Then, the center and the radius of the outer circle are calculated as an average of these two circles – the minimum enclosing and the average circle. Furthermore, additional temporal mean filtering is applied for the values of the palm's center and the radius based on the values calculated for these parameters in three subsequent frames of the video sequence. Thus, rapid changes of the size or the position of the outer circle caused by movement of the convexity defects positions are prevented.

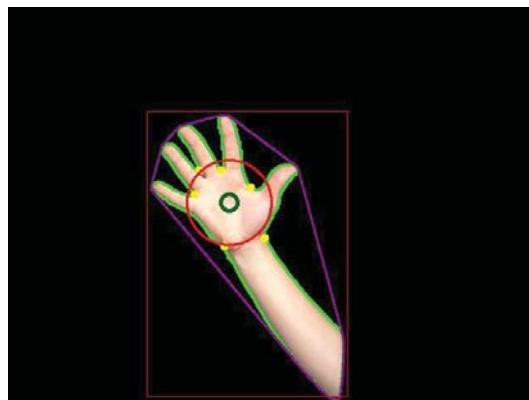


Figure 6. Outer palm circle (the red circle).

2.1.4.2 Inner circle

Additional parameters that are used for the correct palm position detection are the center and the size of the maximum inscribed circle. This one, is the biggest possible circle that is contained within the hand contour and is called inner circle (Fig. 7).

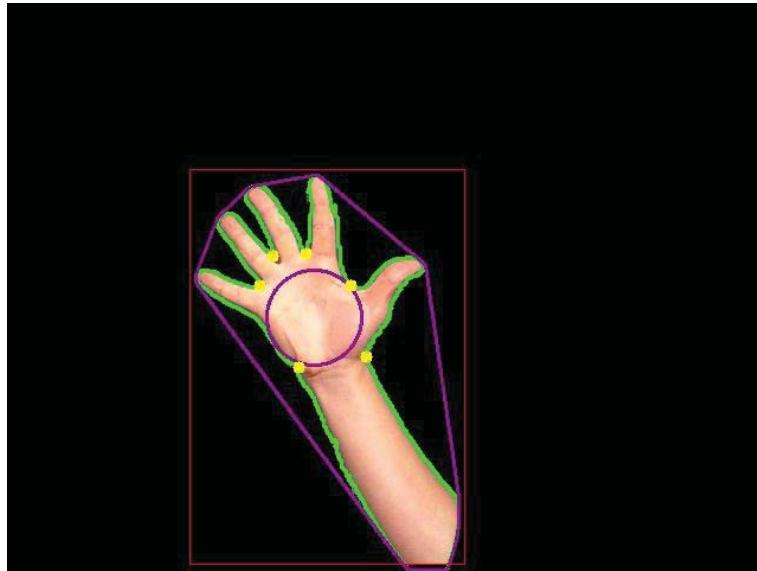


Figure 7. Inner palm circle (the purple circle).

2.1.4.3 Final palm position

The final palm position is estimated based on the parameters of the outer and the inner circles (Fig. 8).

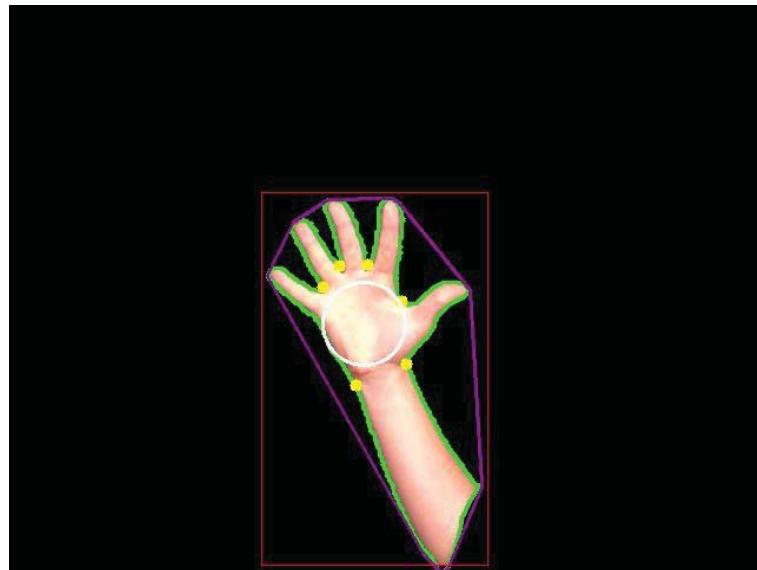


Figure 8. Final palm circle (the white circle).

The purpose of using both the inner and the outer circle is to minimize the false palm detections when the palm position and size are not correct due to wrong convex points. Since both the inner and the outer circle parameters might change with the user's hand movement in the video sequence, some rules on using them are applied for the final palm detection.

Since the radius of the inner circle is a lot more stable than the other radius, the final circle equals the inner circle radius. Three possible cases are considered for calculating the final center of the palm (Fig. 9):

- If the outer circle is inside the bounding rectangle of the hand (the red rectangle), and if the inner circle is inside the outer circle, this means that both circles are correctly positioned on the palm. In this case the final palm center is the average of those two centers (Fig. 9A).
- If the outer circle is on the palm, but the inner circle is not inside it in case there is some wider part of the hand than the palm, then the final center will be the center of the outer circle (Fig. 9B).
- If the outer circle is outside the bounding rectangle, this means that there are not enough convexity defects detected to define this circle. In this case the center of the final palm circle will be the center of the inner circle (Fig. 9C).

The worst case is when both circles are not correct and in that case the palm circle will not be positioned on the palm (Fig. 9D). To prevent such errors, additional features should be used in order to define the palm position on the biggest skin contour.

Once both the palm circle's center and size are defined, the hand can be used for many different purposes that require palm detection and distinction by the arm. In this paper, a palm animation is only applied on the hand palm and is controlled by the palm movement.

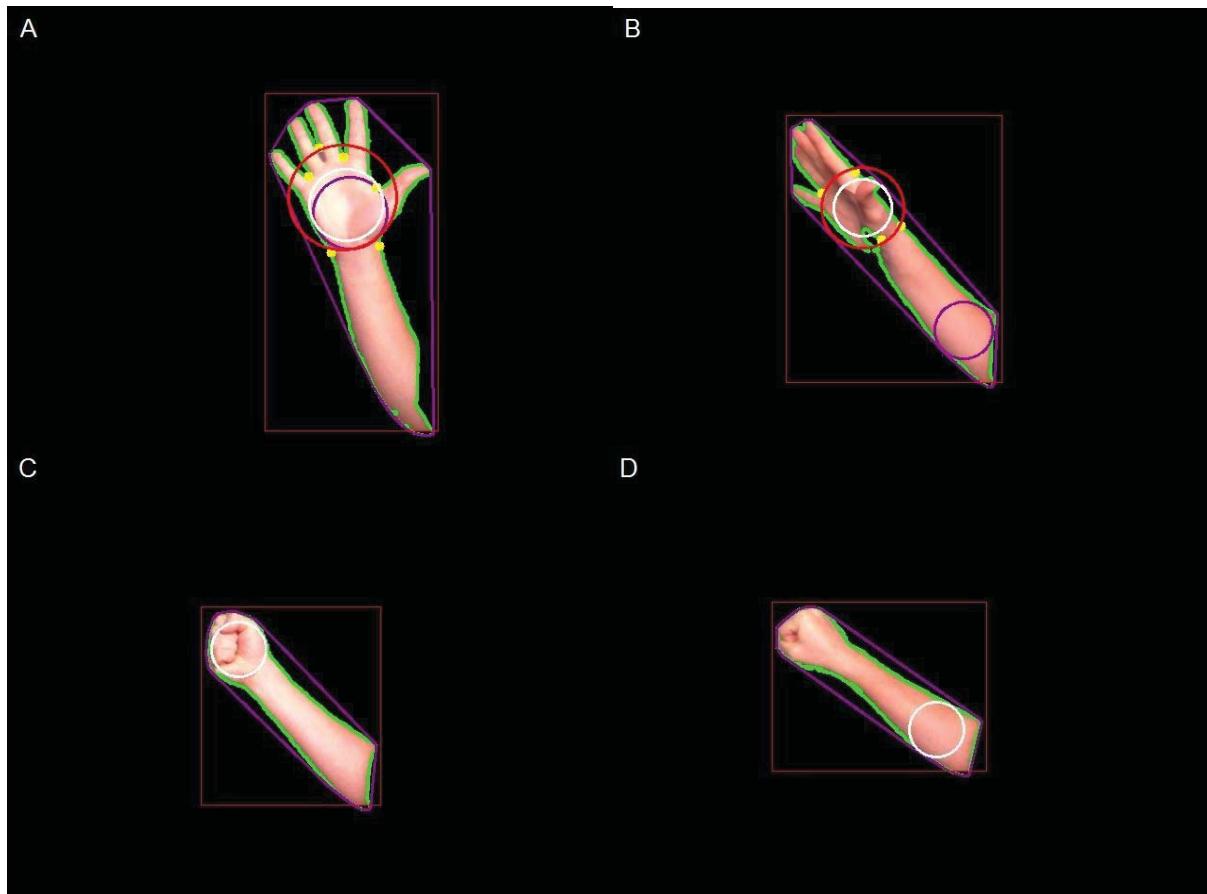


Figure 9. Final palm circle scenarios. A) Best – both circles are on the palm; B) Outer circle defines the final one; C) Inner circle is the final circle; D) Worst – both circles are not on the palm

2.1.5 Fingertip detection

In order to be able to animate the hand palm and to control and change the animation based on the palm position and size, hand gesture recognition is suggested to be used based on the fingertips. The fingertips are detected as the points on the hand contour, that intersect with the convex hull. Since more than one point on one finger can intersect with the convex hull, the length of the hand contour is used to filter the required points of interest: two consecutive fingertip points must have at least 20 other points from the hand contour between them.

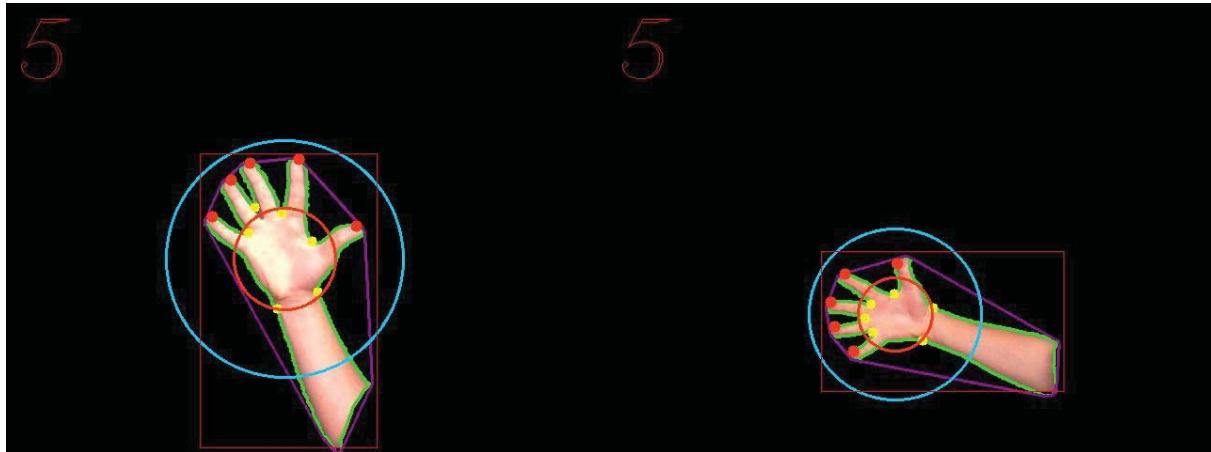


Figure 10. Fingertips (orange dots), lower limit circle (orange) and upper limit (cyan circle).

In addition, in order to prevent the appearance of false fingertips positioned outside of the palm, two limiting circles are used (Fig. 10). Both their centers coincide with the center of the final palm circle, but the radius of the lower limit circle is calculated as 1,5 times the radius of the final palm circle, and the radius of the upper limit is 3,5 times the latter. Only the fingertips that are positioned between the lower and the upper limiting circles are further processed and taken into account for hand gesture recognition required at the palm animation stage.

2.2 Palm animation

Before the animation of the hand to be done, the position of this animation in the frame has to be defined. Since the goal is to animate only the hand palm, it is not feasible to use the bounding rectangle of the whole hand since it's too big for a proper animation effect. That is why, palm circle defined at the previous stage is used as a placement of the 2D animation texture. A rectangle is defined with a size 3,5 times bigger than the palm circle radius, but not bigger than the hand bounding rectangle (Fig. 11).

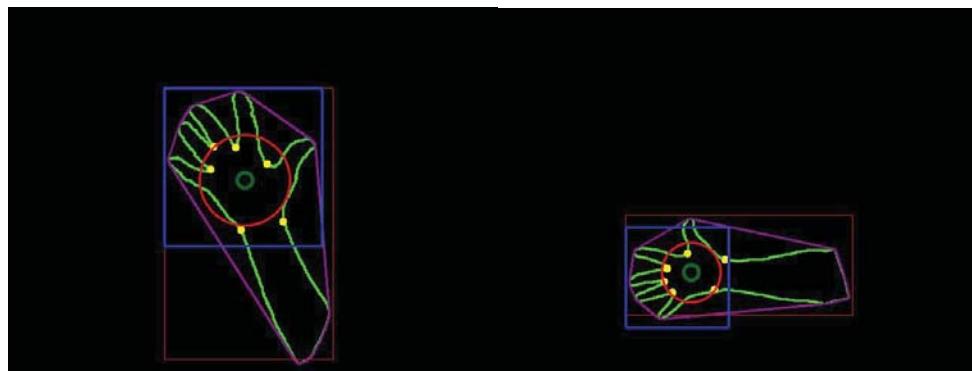


Figure 11. Palm bounding rectangles (in blue color).

Once the position of the animation texture is defined, the next steps for a proper augmented reality effect require modifications of the hand and/or application of some masks on the current video frame. First, a black mask is painted on a copy of the original frame, where the hand is supposed to be. Then, a proper color mask is applied to the hand frame in order to create the effect of an extension or background for the animation. The color depends on the texture chosen for the animation. After that, the animation frame is applied only to the palm using the palm rectangle. Finally, the colored and textured hand is rendered over the position of the black mask of the original frame (Fig. 12).



Figure 12. The hand with orange color mask and fire texture [6].

Since for the calculation of the size of the palm rectangle the palm circle is used, which on the other hand defines the size of the animation frame, the animation size depends on the palm convexity defects. Thus, if the palm shape changes, the animation will shrug or enlarge. However, since the textures used for the animation are 2D, the palm rotation will result in the animation to only shrug but not rotate.

2.3 Gesture recognition and animation control

The selection of the 2D texture to be used for augmented reality animation of the hand palm can be user-driven and can be controlled through a user interface. Another possibility, is to use hand gestures and thus to provide remote change of the animation texture based on a haptic human-computer interaction.

The gesture recognition requires selection of an approach for gesture classification and application of that approach for an unknown gesture. The classification methods use training set of positive and negative gesture samples and some approach for extracting features, that distinguish the required gestures. The requirement for creation and collection of images for the training set, as well as the time that is usually needed for the recognition system training, makes this approach useful, but mostly applicable for more complex recognition tasks. Since the gesture recognition in the solution described in the paper will only be used to change the animation texture among several predefined options and will only make use of few distinct gestures, the recognition is based on some geometric rules for the position of the palm feature points.

The fingertips and the hand circle center are the most important features of the hand, that are used to distinguish different gestures. The gestures are discriminated by comparing the position of the fingertips to the position of the palm center. Since the application of the described augmented reality will be haptic based control with four predefined animation textures, only four gestures are

required. The selected gestures are: one finger up, one finger left, two fingers up, two fingers left. Several simple classification rules are defined, for example one of the rules for the gesture with two fingers is: if both fingertip positions are left of the palm circle (center minus radius), then the gesture is recognized as two fingers left.

The different gestures and the corresponding textures used in the described solution are shown on Fig. 13 – Fire [6], Lightning Ball [7], Lightning Bolt, Ice. The gestures are shown in the left column images, together with their features: the points of interest, circles and rectangles. The results of rendering the selected texture over the video frame image are shown in the middle. The images to the right are the same textures that are picked up by the hand gesture recognition, but here they are rendered over a flat (open) palm. There is a possibility for erroneous texture change, when the palm is opened after gesture-based selection of the preferred animation. However, the described approach for gesture-based control of animation texture has the advantages of simplicity and efficiency over the complex and more sophisticated classification methods.

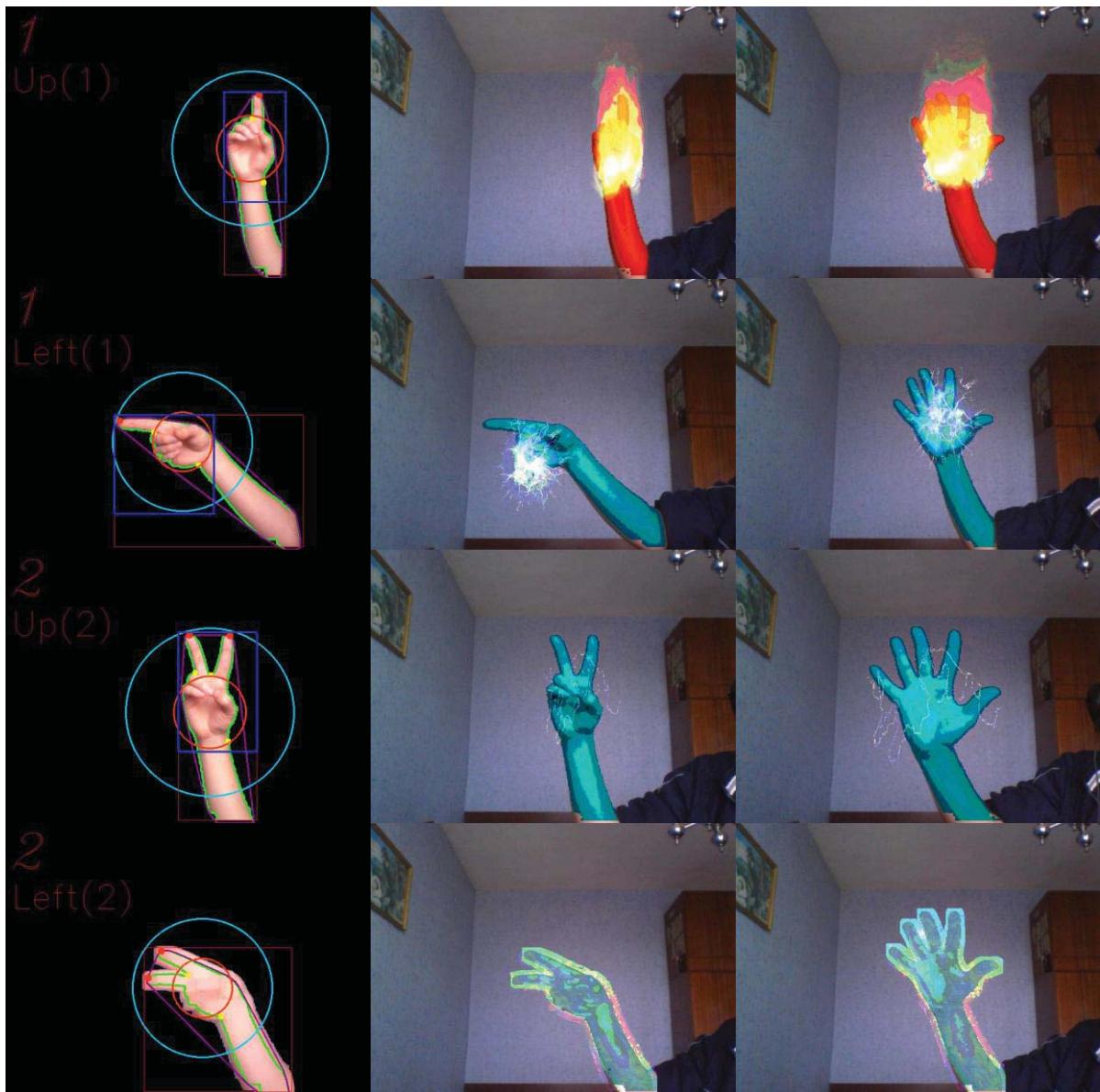


Figure 13. Gestures (left), animated gesture (middle), animated open palm (right).

This method will be effective and applicable, as the gestures used for animation control are very few and can be easily discriminated using simple geometric rules for the palm features, that are already computed at the previous processing steps.

3 Experimental analysis and future improvements

The experimental analyses of the described approach are based on a software application developed in C++ using OpenCV library. The experiments were made in order to verify the processing goals of each step of the processing pipeline and to evaluate the accuracy of each step and the overall accuracy of the augmented reality application. The results can be summarized as several suggestions for further improvements in order to provide more precise operation and thus better and more stable usage of the haptic based control system.

One of the most important steps, which has great influence on the results, is the step of background subtraction and skin detection. One of the further improvement of the system will provide better background and skin filtering, that are least affected by the room lighting and light direction and allows for fast and efficient calibration and adjustment of the filtering parameters according to the light changes.

The final palm circle calculation is the next step that can be improved, so that the center and the size of the palm circle do not change when the hand is not moving. Convexity defects that are used for palm position and size estimation will have to be filtered more precisely in order to provide for better accuracy of the palm detection.

At the face detection step the classification is based on a training set comprising positive and negative face samples. The classifier used in the current implementation is the one provided with the OpenCV library. The experimental analyses showed some false positive errors, meaning that in a specific rotation and shape the hand is recognized as a face. The reason is lack of images of a hand used as negative samples in the training set. Further improvements of the described system will reduce that type of errors by adding hand images to the training set and retraining the face recognition system.

The described gesture recognition approach only uses few geometric rules for the position of the hand and palm features. The rules are highly dependent on the correct computation and detection of the fingertips. The experimental analyses showed that some errors occur in the gesture recognition mainly due to errors in the fingertips identification. Better accuracy will be achieved if the gesture recognition involves more complex approach based on classification and recognition stage using training set of sample images. Nevertheless, as the purpose of the described system for haptic control will only require few gestures to be recognized and the real-time performance is an essential requirement, the suggested gesture recognition using geometric rules provides both reasonable accuracy and computational simplicity.

4 Conclusion

The paper described a system capable of real-time hand palm detection, tracking and animation. The final result of the animated hand provides augmented reality view that is similar to the motion capture approach widely used in filmmaking and video game development. The specific of the motion capture is the offline processing: it first records the movement of human parts (full body or just the hand) along with markers attached to them, and applies the animation effects later using the data captured.

The approach described in the paper provides real-time animation of 2D texture over a hand palm that is controlled by the hand movement and gestures. Due to lack of any guiding markers, there are some inaccuracies in the hand detection that can lead to incorrect system response. But since all the computations are done in real time, further improvements by more precise parameter selection and better filtering and recognition may affect the system performance.

There are several applications of the described process mainly targeted at haptic human-computer interactions. The developed implementation of a gesture driven animation of the human hand can be embedded in a computer game and be used for real-time remote control of the game characters behavior.

5 References

- [1] M. Kitagawa, B. Windsor, *MoCap for Artists: Workflow and Techniques for Motion Capture*, Focal Press, 2008.
- [2] *Open Computer Vision Library*, opencv.org
- [3] *Shadow removing with OpenCV*, 2011, hi.baidu.com
- [4] P. Viola, M. Jones, *Robust Real-time Face Detection*, International Journal of Computer Vision, Vol. 57, No.2, 2001, pp. 137-154.
- [5] Wei-chao Chen, *Real-Time Palm Tracking and Hand Gesture Estimation Based on Fore-Arm Contour*. National Taiwan University of Science and Technology, 2010.
- [6] *Fire Animation Loop*, WGStudio, 2012, opengameart.org.
- [7] *Electrical energy ball - digital animation*, shutterstock.co

Vasil Ivanov TSOLOV
Technical University of Sofia
Department: Computer Systems
Technical University of Sofia , 8 Kliment Ohridski St., Sofia-1000
BULGARIA
E-mail: vasil.tsolov@gmail.com

Unlimited register machine simulator

**Eva Tuba
Teacher Coordinator: Milan Tuba**

Abstract

One of the fundamental questions that the theory of computation deals with is whether a particular problem is algorithmically solvable or not. The theory of computation also examines the resources needed for solving the problem. For researching these questions mathematical abstractions of machines for computations were developed. This paper covers one of the aforementioned models, unlimited register machine, URM. It presents simulation of the device giving the user full control over the registers and command lines and allowing him to watch problem solving in real time while choosing the speed of commands execution. Few of the algorithms for URM programs are presented.

1 Introduction

Theory of computation is a part of mathematics and theoretical computer science that deals with the following question: “which problem, simple or complex, is computationally solvable using different models of computation, tools and techniques”. The theory shows that some problems cannot be solved, but also that for solving some problems we need infeasible amount of resources, such as thousands of years of computation time. This theory gives us approaches for recognizing such problems.

In computer science, solving the problem by using an algorithm and writing the program is one of the main tasks. For an actual problem, besides finding a solution, we have to consider how to solve it, how complex the algorithm is, the execution time, and the needed memory space. For all that we use models of computation. Models of computation are sets of operations which we use in calculations and their respective costs.

Today we have different processors, but all of them can calculate the same things. Current processors are 32-bit or 64-bit. When you are installing some software usually you have to choose between the software for 32-bit processor and for 64-bit. Software will calculate and execute the same thing on the both of processors, only question is how fast. Looking back, before this, we had 16-bit processors, which replaced 8-bit processors. At the very beginning processor was 4-bit. None of this upgrading didn't change what we can calculate. When we toss a coin, which side will fall cannot be calculated by any processor. We could theoretically use 1-bit processor, and operate with it, and this will not make any difference on what we can do. This is why models of

computation are very useful and needed. We do not want to deal with complexity of the processor, we want to see what we can do with it. Model of computation such as unlimited register machine is a very effective way to give an answer to that question.

2 Formal model of computation

For writing a program, first we have to design an algorithm. That algorithm must give us a correct output for the input. Also, we are trying that the algorithm is effective as it can be and we have to know how effective that is. Using theory of computation we can calculate all of that.

Purpose of program based on the algorithm is that for given input, returns result. However, it is possible that a program never stops, so we never get the output. Whether this is going to happen or not formally known as halting problem. In computer theory, halting problem is equivalent to the decision problem. The algorithm which solves halting problem for all possible program-input pairs doesn't exist. This is proved by Alan Turing. He succeeded in that by defining a computer and a program, Turing machine which is one model of computation.

Based on what algorithm should do we can describe it as a procedure that executes some instructions and provides the output in finite time for a certain input. A valid algorithm for the input must provide a desired output.

There are a lot of well-known algorithms (for example algorithm for multiplication of integers, Euclid's algorithm, tabular method for testing if the propositional formula is tautology...). If a procedure of solving a problem is known, we can easily say whether or not the problem is algorithmically solvable. However, moving forward in mathematics we constantly discover problems that we are unable to solve. It is a question whether that is because of our incapability to solve it or it is impossible. For example, is there an effective algorithm for finding nonnegative solutions for some Diophantine equation, $p(x_1, \dots, x_m)=0$, where $p(x_1, \dots, x_m)$ is polynomial with integer coefficients and variables x_1, \dots, x_m (e.g. $x_1^3x_3 - 5x_2^4 + 7$) . One way to solve this would be enumeration of all m-tuples of integers and checking whether or not they represent zero polynomial. If there is a solution we will find it sooner or later. For some equation that doesn't have integer solution (for example $x^2 - 2 = 0$) this algorithm will never finish. Checking if there is a solution for Diophantine equation is also known as Hilbert's tenth problem.

To give an answer to this question we have to define what an effective algorithm is, either to find a solution that falls under that definition or to determine that there is no solution with the required performances. There are a few approaches to define an effective algorithm: the system of computation that is represented in the formal system of arithmetic (Gödel-Herbrand-Kleene recursive functions), Church's λ -calculus, systems based on automata, rewriting systems, while programs (program includes only assignment command, sequencing of commands and while command)... Even though these systems are very different, at the end they are all equivalent. This equivalency and the failed attempts to construct a problem whose solution doesn't belong in any of these systems, entitles us to believe that we achieved some absolute concept and that all algorithms can be expressed in any of this systems. That is written in the Church thesis.

For computer science the most interesting system is system based on automata. This system contains Turing machine, Post machine, Unlimited Register Machine (URM)... This system formalizes the notion of algorithm describing ideal models of computers. The interesting fact is that some of these systems were made before digital computers. Even though we use word machine, we should know that we are actually talking about abstract mathematical concepts and not about real objects.

In theoretical computer science we study a few classes of machines. A finite automaton is a model where no memory is used. Push-down automata is model where memory is specially organized and the string automaton is a model where input and output is limited on words.

2.1 Turing machine

One of informal description of Turing machine is that it is composed of a tape divided into cells (memory registers), head and state register. The tape is unlimited to the left and to the right (number of cells is unlimited). The head is always above one register and it can read from the register, write either character 0 or 1 into the register, or move one position to the left or to the right.

Turing machine potentially has infinite memory. Initially or in any time of calculation only finite number of memory registers is used (there is no limit how big that number can be). Also at any point in time Turing machine can ask for a new register and every time that request will be approved. Even though only few operations are allowed they are enough to describe an arbitrary algorithm. The advantage of Turing machine in relation to programming language with more operations precisely is the simplicity, which provides us with easier analysis.

2.2 Post machine

Post machine is system developed before the paper that describes Turing machine is published, but it is issued later. Post machine is very similar to Turing machine. It also contains unlimited tape divided into cells. Every cell can be marked (dash character written in a cell) or unmarked (space character written in a cell). Marked cell represents 1 and unmarked cell represents 0. Post machine also has a head that moves to the left or to the right and reads a content of the cell or write into the cell [1], [2].

3 Unlimited register machine

Unlimited register machine (URM) is theoretical concept of an ideal computer. It is considered to be ideal because of the unlimited memory, unlimited time of executing and infallibility. The URM and Turing machine are equivalent. Functions that can be calculated using a Turing machine are the same function that can be calculate with URM.

If URM program for a function $f: N^n \rightarrow N$ (n is natural number), for any n -tuples (a_1, a_2, \dots, a_n) as input, gives the output $f(a_1, a_2, \dots, a_n)$, function f is URM computable. For any natural number n , registers R_1, R_2, \dots, R_n contain a_1, a_2, \dots, a_n , and at the end the value of register R_1 is $f(a_1, a_2, \dots, a_n)$.

URM contains space for the program, infinite array of memory cells (registers) and counter. URM programs use only finite number of registers, but there is no limit for that number. Counter is a natural number that indicates the instruction that program should do in that particular moment. There is one natural number in every register. Common notation for register is upper letter R with a natural number in subscript (for example R_1 , where 1 is index of register). Value of the register is marked with lower letter r with index (for example value in the register R_1 is r_1).

URM program is finite array of commands. All commands of URM program are described in the following table.

Notation	Name	Effect	Description
$Z(i)$	Zero	$0 \rightarrow R_i$	Change the content of R_i with 0 ($r_i=0$).
$S(i)$	Successor	$r_i+1 \rightarrow R_i$	Increment the number in R_i .
$T(i, j)$	Transfer	$r_i \rightarrow R_j$	Replace the number in R_j with value from R_i (R_i stays the same). This instruction is also called Copy.
$J(i, j, k)$	Jump	$r_i=r_j \ ? \Rightarrow k$	If the number in R_i is equal to the number in R_j ($r_i=r_j$), jump to the command number k

At particular point in time, state of URM program is defined by the value of the counter and the values of every register. Program ends when non-existent command is reached (this is often achieved by jumping unconditionally to an undefined command). It is possible for a program to never end. It happens when program gets into the endless loop. At the end, the result of program is in the first register.

URM is deterministic finite automata. For fixed input, each command determinate the next command. That means that the URM program, for fixed input will finish every time exactly the same way. Either it will produce identical output (URM program terminates) or it will runs forever (URM program is non-terminating).

URM is used for studying extensions of the process algebra axiom system [3]. Nowadays, computer scientists are also using the modified Unlimited Register Machines (which are equivalent to URM) for proving theories, finding better algorithms for some problems [4]... Infinite time register machine limits the registers contents as infinitive limes of the previous register content. It is used for proving that a real number is computable if it is hyperarithmetic [5], [6].

4 URM Simulator

URM Simulator is written in Microsoft Visual Studio. For making the simulator more pleasant, for animations (moving commands and registers during the execution) and for the possibility of choosing the speed of executing I used the XNA game studio, that extends Microsoft Visual Studio with support for XNA Framework and tools.

Registers are shown at the top of the window. Every register contains the value and the registration index. Commands are at the left side of the window. Commands are being written and executed from top to bottom. On the right side of the command array there are buttons for every

instruction and a button for deleting selected (previously written) command. To add a command first you have to choose from a list of available commands (S, Z, T or J) by pressing their respective buttons, and then selecting a register or a command. Once the button for selecting a command is pressed the user is prompted to select one of the following: one register, two registers or two registers and a command to jump to (depending on the command). On the left side of the window there are options for deleting all commands, saving the program, debugging, choosing the speed of debugging, stop debugging and a button used for returning back to the main window. All of the saved programs can be found in the folder called Programmes. They have an extension .urms. When choosing a speed the user should be aware that the selection of speed three will almost instantly execute all of the commands, speed one is rather slow so you can see exactly how the program works and follow the current state of the registers. Speed two is somewhere in between. In any time during the execution you can stop the debugging process. After stopping the process the user has an option to continue debugging by clicking on the debug button, program will continue its work from the point where it stopped.

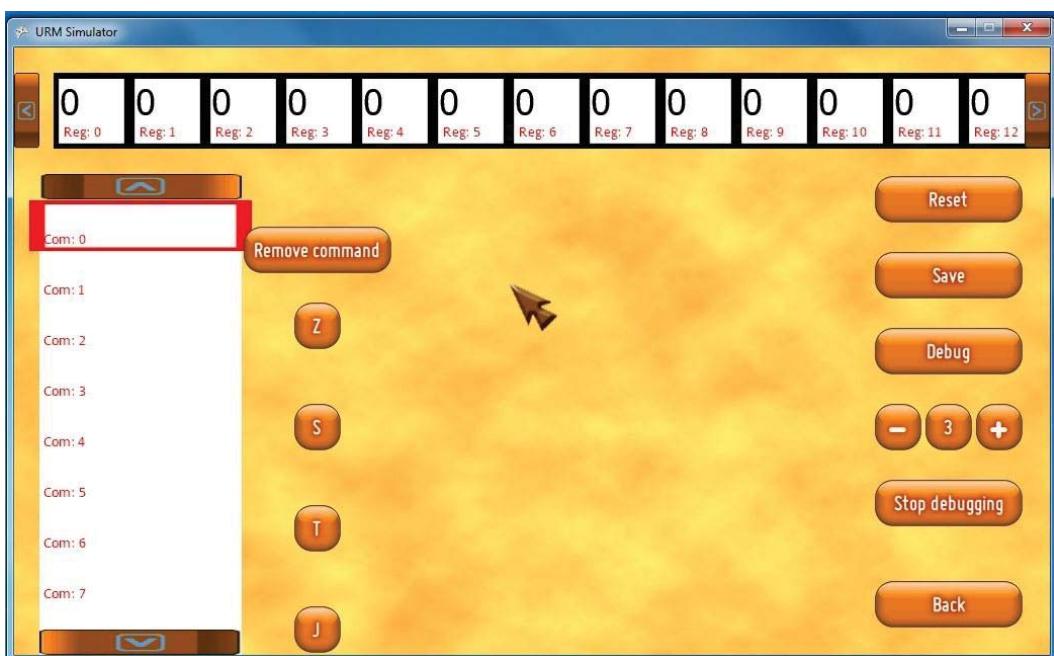


Figure 1: URM simulator main window

In this application program can be written from GUI using the buttons for adding commands or it can be written in text editor, using the template file, and save it with .urms extension. Template file can be found in the same folder where all the programs are (...)\Programmes). If the user chooses to write program in text file, the concept is fairly easy. The commands should be placed in between the string tabs (for example <string> S(5) </string> or <string> J(1,2,7) </string>).

5 Code examples

The examples below are shown as they are in .urms files (opened with any text editor). Two algorithms are explained, the algorithm for addition and one for multiplication of two numbers.

5.1 Sum of two numbers

Addition is reduced to finding the successor. If we have to find sum of x and y , the problem is actually equivalent to finding the y -th successor of x . We need a counter to monitor how many times we add one to the first register (which contains the value of x in the start). R_0 will contain the value of x , R_1 contains the value of y , and for the counter we will use R_2 . At the end R_0 will contain the result.

```
<?xml version="1.0"?>
<SaveGameData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<saveCommands>
<string>S(0)</string>
<string>S(0)</string>
<string>S(0)</string>
<string>S(0)</string>
<string>S(1)</string>
<string>S(1)</string>
<string>S(1)</string>
<string>J(1,2,12)</string>
<string>S(0)</string>
<string>S(2)</string>
<string>J(5,6,7)</string>
</saveCommands>
</SaveGameData>
```

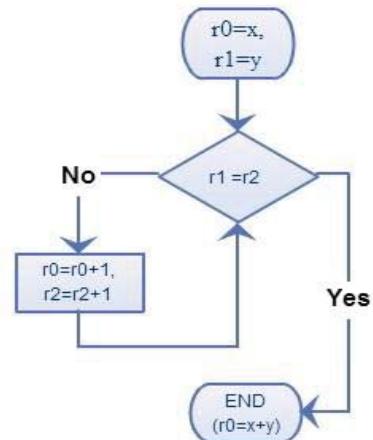
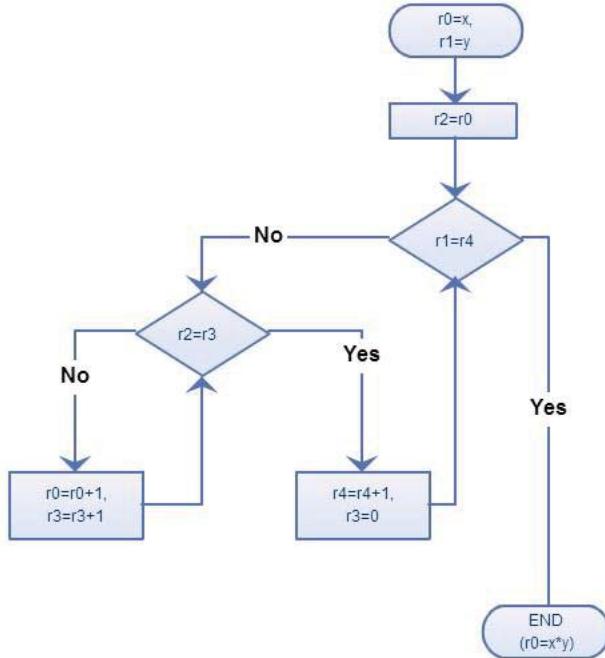


Figure 2: End state for addition

5.2 Multiplication of two numbers

Multiplication can be reduced to addition. In previous example we saw that addition is reduced to problem of finding the successor. Actually $x*y$ is $x+x+...+x$ where y is the number of summands, so we need a counter for how many times we added x . Also we need the second counter which

will be used for handling addition. Variable x is in R₀, y is in R₁. In R₂ we will save x, because R₀ will be changed during process, in R₃ will be counter for x, and in R₄ will be counter for y.



```

<SaveGameData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <saveCommands>
    <string>S(0)</string> //set the input
    <string>S(0)</string>
    <string>S(1)</string>
    <string>S(1)</string>
    <string>S(1)</string>
    <string>S(1)</string>
    <string>S(4)</string> // one x is already in first register
    <string>T(0,2)</string> // saving the x, because in first register we will put x*y
    <string>J(1,4,19)</string> // if R1=R4 jump to command 19 (command 19 is empty, so program stops)
    <string>J(2,3,13)</string> // if R2=R3 jump to command 13 (S(4))
    <string>S(0)</string> // increase R0
    <string>S(3)</string> // increase R3
    <string>J(9,10,9)</string> // if R9=R10 jump to command 9 (unconditional jump to J(2,3,13))
    <string>S(4)</string> // increase R4
    <string>Z(3)</string> // set R3 to 0
    <string>J(9,10,8)</string> //unconditional jump to command 8 (J(1,4,19))
  </saveCommands>
</SaveGameData>
  
```

6 Conclusion

URM represents very significant model of computation used in theoretical computer science. It is used to ignore the complex architecture of computers in aim to focus on its ability of calculations. Also it gives us an information about needed resources, so we can easily check if it is possible to do on the computer in reality.

In this paper software with graphical user interface which is able to process URM commands sets by the user is represented. The purpose of this software is to experiment with different algorithms,

to try other approach to solving the problems... URM simulator enables us to unquestionably correctly execute the written algorithm in a very short period of time. This simulator saves time needed for testing. If the mistake is made, it is easy to find it by following the execution in the slow mode.

Some of the problems caused by development environment and design choices will be fixed in future version. It will be able to set directly input, besides using the command S. The plan is to add an option that translate program written in the other models of computation to URM program, and reverse. The aim of the option is to experiments with different models, compare algorithms etc.

References

- [1] A. Pettorossi, *Elements of Computability, Decidability, and Complexity*, Aracne Publisher, p. 184, 2014.
- [2] J. D. Hamkins, R. Miller, *Post's Problem for ordinal register machines: An explicit approach*, Annals of Pure and Applied Logic, Vol. 160, No. 3, pp. 302-309. 2009.
- [3] J. A. Bergstra, A. Ponse, *Register-machine based processes*, Journal of the ACM, Vol. 48, No. 6, pp. 1207-1241, 2001.
- [4] V. Cortier, *About the decision of reachability for register machines*, Rairo-Theoretical Informatics and Applications, Vol. 36, No. 4, pp. 341-358, 2002.
- [5] M. Carl, T. Fischbach, P. Koepke, *The basic theory of infinite time register machines*, Archive for Mathematical Logic, Vol. 49, No. 2, pp. 249-273, 2010.
- [6] P. Koepke, *Infinite time register machines*, Lecture Notes in Computer Science, Vol. 3988, pp. 257-266, 2006.

Eva TUBA
University of Belgrade
Faculty of Mathematics
Studentski trg 16, Belgrade
SERBIA
E-mail: eva.tuba@gmail.com

Milan TUBA
Megatrend University
Faculty of Computer Science
Bulevar umetnosti 29, N. Belgrade
SERBIA
E-mail: tuba@ieee.org

On Investigating Promoter Areas for Regulatory Motifs in the Solanum Lycopersicum Genome

Dagmawi Zegeye
Teacher Coordinator: Sofia Visa, Esther Van der Knaap

Abstract

Promoters are very important for gene expression, but at the same time are difficult to identify in particular because they can occur anywhere up to 3,000 nucleotides upstream of a gene. Here we computationally investigate potential promoters in the 1,000 and 3,000 bases promoter regions of 79 tomato fruit genes of interest. The potential promoters found in the 1,000 bases region are also found in the 3,000 bases region albeit rather diluted. However, we find that most of the potential promoters found in the 3,000 bases region are not present in the 1,000 bases region.

1 Introduction

Identifying promoter elements (also known as regulatory motifs) is very important to understanding how genes are expressed and how networks of genes work. However, finding these elements is very difficult and a true *finding a needle in a haystack-type* of problem. These motifs are very short (5 to 8 DNA nucleotides), can have one variable nucleotide at one or more positions, and they can occur anywhere before the start of a gene. Due to these factors, it is a real challenge to identify promoter elements.

Because promoter elements can occur anywhere upstream of a gene, the identification must start computationally, and only after a smaller set of potential promoters are identified based on some a priori assumptions, laborious and time-consuming laboratory experiments can confirm true promoters. For example, to confirm that a particular sequence is a true regulatory motif for a particular gene, the researcher removes this short sequence from the DNA, and if that particular gene is no longer transcribed, it is a good indication that the gene is regulated by this particular motif. This laboratory experiment is known as the knock-off of a gene. However, these experiments are very laborious and therefore can not be used as a trial-and error type of research for regions as large as 3,000 bp. Therefore, initially the researchers try to computationally identify a smaller pool of potential promoter elements, which later can be managed with laboratory experiments such as the one described above.

Here we computationally search for promoters in several tomato fruit genes of interest, which might be involved in regulating the flat shape in a tomato fruit. They are of interest to us, because the research presented here is part of a larger research project that aims at identifying genes and networks of genes that regulate size and shape in the tomato fruits.

Promoters are short DNA sequences, which help in the transcription (i.e. expression) of a gene. Little is known about promoters in the tomato fruit genome, and here we computationally try to identify promoter elements, which can be further analysed in lab experiments. As illustrated in Figure 1, the promoter elements can be found anywhere upstream of a gene, but scientist suspect

that most of them are located within the 1,000 bp (base pairs). While many promoter elements are located within the 1,000 bp upstream region of a gene, some can also be found further away within the 3,000 bp upstream region or, less frequent, even downstream of a gene. For our particular data sets we investigate the relationship between the most frequent sixmers (i.e. short sequences of six DNA nucleotides) found in the 1,000 and 3000 bp regions, to identify which region should be best considered when mining for potential promoter elements. Of course, these frequent sixmers are computationally identified and only some of them are promoter elements, but only additional laboratory experiments will identify the true ones.

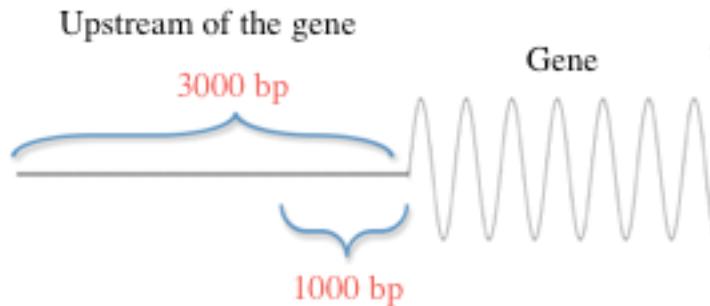


Fig. 1 Promoter elements can be found anywhere up to 3,000 bp upstream region of a gene.

2 The data sets

The genetic sequences used in this research are obtained from two wild type tomato fruit mutants, one having the *LC* gene mutation, and the other one having *LC* and *fas* mutations [1]. Both these tomato plants produce flat tomatoes (see Figure 2), but the plant caring the two mutations displays a more accentuated flat shape.



Fig. 2 Illustration of a flat tomato fruit. The *fas* and *LC* are two genes known to contribute to the flat shape of a tomato fruit [1].

The steps taken to obtain the data sets are briefly described next.

First, using the free Bioinformatics package described in [8], the differentially expressed genes were isolated, meaning that only those genes that are much higher expressed in one mutant than the other were considered. This step is necessary because the larger objective of this research is to identify genes, and networks of genes, that affect the shape of the tomato. In particular, it will be interesting to discover other genes that interact with *LC* and *fas*, together leading to the flat shape in tomato fruits.

Second, we used R [9] to cluster these differentially expressed genes according to their expression level observed over six time-points. Several clusters resulted, but here we focus on two particular ones: *clv3* consisting of 40 genes, and *wuschel* consisting of 36 genes. The corresponding expression levels of these two clusters over the six time-points are shown in Figures 3 and 4. These two clusters are further studied because some genes in these clusters show an

interesting interaction. It has been observed that several genes in these clusters work in a feedback regulatory loop, meaning that the expression of one or more *clv3* genes triggers the expressions of some *wuschl* genes and the other way around.

Finally, the 1,000 bp and 3,000 bp upstream regions of these genes were extracted from the tomato genome. This step produced four text files having the corresponding 1,000 and 3,000 DNA nucleotides for each of the two clusters, *clv3* and *wuschl*. To access the tomato genome and extract the desired region, we wrote Python code that used the Biopython package [10].

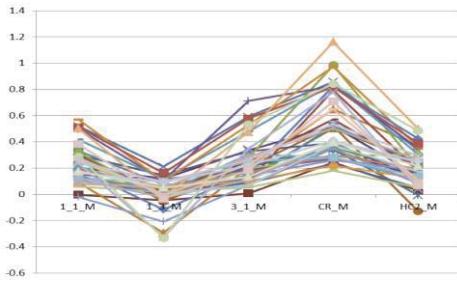


Fig. 3 The 40 genes in *clv3* cluster display similar expression pattern measured at six different developmental time-points.

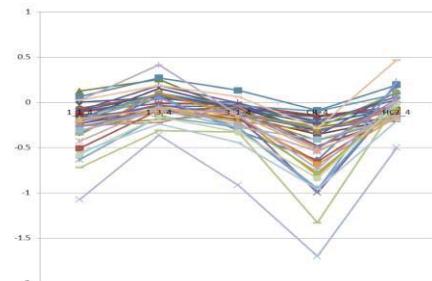


Fig. 4 The 39 genes in *wuschel* cluster display similar expression pattern measured at six different developmental time-points.

3 The experiment – 1,000 bp vs. 3,000 bp

In this section we present the algorithm we used to identify promoter elements, and we compare the results obtained from searching the two upstream regions, namely the 1,000 and 3,000 bp regions. In particular, we investigate whether the two search regions will produce similar results, and try to understand which of the two regions are more suitable for computationally mining promoter elements.

There are several free software packages available for identifying promoters [4]-[7]. However, past experiments presented in [2] and [3] show that these particular software packages did not identify many promoter elements for our tomato genes. Therefore, we use the algorithm proposed in [2] which makes use of the observation that a true motif is more frequent in the upstream region of a gene than in the corresponding shuffled region.

Using the algorithm described in [2], in each of the four data sets we compute the most frequent sixmers (i.e. a DNA sequence of length six) and we record these frequencies denoted by f . Next, we shuffle 10,000 times the individual data sets, and recompute these frequencies for each shuffle. The averaged shuffled frequency is denoted by f_{rand} , and the ratio f/f_{rand} is computed for each individual sixmer. This ratio is a good indicator of whether a particular sixmer occurs by chance or has a biological significance. If this ratio is close to 1 (meaning that the corresponding sixmer is found with equal frequency in the original and shuffled sequences) the sixmer is not likely to have a biological function, and therefore, in particular it is not likely to be part of a promoter element. On the other hand, the larger this ratio, the higher is our belief that the sixmer has some biological functionality. Here we further analyse only the sixmers having a ratio of three or larger.

4 Results

By using the algorithm described above and proposed in [2], 38 and 108 sixmers are found in the 1,000 and 3,000 bp regions of the clv3 cluster, respectively. Figure 5 shows in blue the frequency ratio of the 38 sixmers identified in the 1,000 bp region, and in red is plotted the same ratio for the same 38 sixmers searched in the larger region of 3,000bp. As it can be observed from Figure 5, all 38 sixmers are found in the larger region, but with a lower ratio score. However, the converse experiment is not true. Namely, as Figure 6 shows, only 8 of the 108 sixmers identified firstly in the 3,000bp region are also found in the 1,000 bp region.

Figures 7 and 8 show the results for the same experiments applied to the wunschel cluster. In this experiment, 73 and 105 sixmers with a frequency ratio of 3 or higher were identified in the 1,000 and 3,000 bp regions, respectively. Again, very few sixmers identified first in the 3,000 bp region are also found in the 1,000 bp region (see Figure 8).

One obvious observation resulting from the analysis of both clusters is that most of the promoter elements found in the 3,000 bp region are not found at all in the 1,000 bp region. This might be a slight indication that most of the promoters are located within the 1,000 bp region, and therefore it is better to start searching for such elements in this region. This is just a hypothesis, and future research on a set of genes for which the promoter elements are known can further investigate this hypothesis.

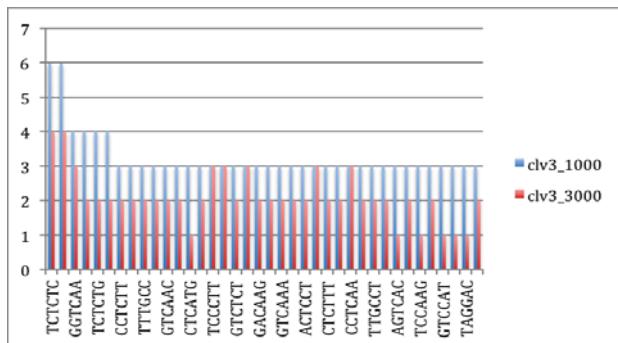


Fig. 5 Analysis of promoter elements found in clv3, 1000 bp region; y-axis shows the ratio of these promoters for 1000 (blue) and 3000 (red) bp regions.

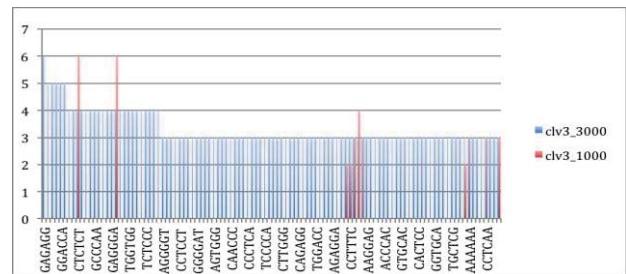


Fig. 6 Analysis of promoter elements found in clv3, 3000 bp region; y-axis shows the ratio of these promoters for 3000 (blue) and 1000 (red) bp regions.

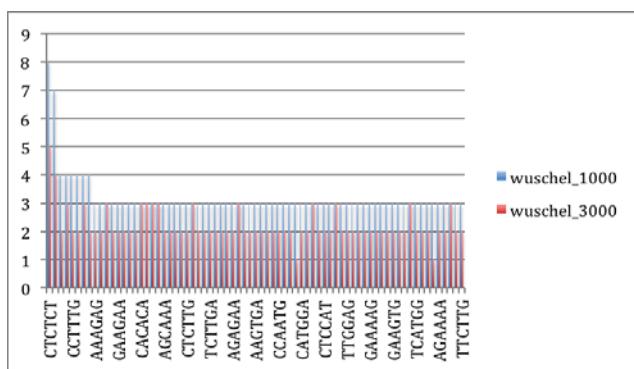


Fig. 7 Analysis of promoter elements found in wuschel, 1000 bp region; y-axis shows the ratio of these promoters for 1000 (blue) and 3000 (red) bp regions.

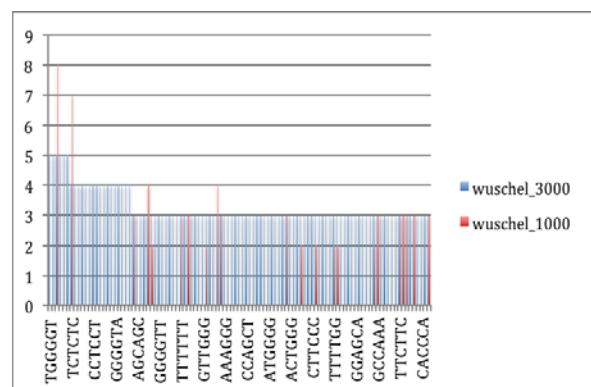


Fig. 8 Analysis of promoter elements found in wuschel, 3000 bp region; y-axis shows the ratio of these promoters for 3000 (blue) and 1000 (red) bp regions.

Another possible exploratory direction is to use the search results from both regions and analyze the common sixmers which were first identified in the 3,000 bp region. In our particular data sets most of these sixmers were repeats (such as CTCTCT and TTTTTT), which are uninteresting for identifying promoters. But there were a few non-repeats. Namely, six such common sixmers were found in clv3: GGTCAA, CACATG, CACAAG, CCTCAA, TCCCTT, and CCTTTC; and seven where found in wuschel: GCCAA, TGCTTC, TTCTTC, TCTCCA, TTGGTG, GTTTGG, TTTGGC.

5 Conclusions and future work

Here we computationally searched for promoter elements in a shorter (1,000 bp) and a larger (3,000 bp) upstream region of a gene. Two clusters of genes from tomato fruit genome were investigated. Because these genes display the same expression pattern over six developmental time points, the assumption is that they share some common promoter elements, and therefore these genes can be used together in this search. Computationally, many potential motifs (i.e. motifs occurring more frequent than expected) of length six are identified, but further laboratory experiments are needed to confirm the true ones. When comparing the search results of the two upstream regions, it is observed that most of the sixmers identified by our search algorithm in the 3,000 bp region are not found in the 1,000 bp region, but all the sixmers identified by the algorithm in the 1,000 bp region are found in the larger region. Therefore, if further experiments will show that most of the true promoter elements are located in the 1,000 bp region, the above observation indicates that searching a larger area dilutes the true frequencies of the promoter elements and diminishes the significance of the frequency score.

These experiments might not be conclusive because we do not know if these genes really do share common promoter elements. We simply assume this to be true due to their similarity in expression pattern. But, if these genes do not have several such common motifs, then the frequency analysis presented here is not relevant. Therefore, in future it will be very useful to repeat this experiment for some genes for which the promoters are already known, like genes from Arabidopsis for instance. Such a future work will reveal true relationships between the two regions and a frequency-based promoter search.

Another future work concerns the development and implementation of an algorithm that will combine these sixmers into promoter motifs. Here we analysed only the individual sixmers which occur with high frequency and not likely due to chance. But since true promoter elements can have one or more positions variables (for example, AGTTCA and ACTTCA can be two instances of the same motif), these individual sixmers must be merged according to their similarities in order to obtain the true promoter elements.

Acknowledgement: This research is funded from the research Grant NSF DBI-0922661(60020128).

References

- [1] G.R. Rodriguez, S. Munos, C. Anderson, S.C. Sim, A. Michel, M. Causse, B.B. McSpadden Gardener, D. Francis, and E. van der Knaap. *Distribution of SUN, OVATE, LC, and FAS in the Tomato Germplasm and the Relationship to Fruit Shape Diversity*. Plant Physiol 156: 275-285. 2011.
- [2] M. Blackwood, S. Visa, E. van der Knaap. *Finding regulatory motifs in the Solanum Lycopersicum genome*. In Proceedings of the International Students Conference on Informatics - Imagination, Creativity, Design, Development Conference, Lucian Blaga University, Romania, pages 29-37, 2013.
- [3] M. Blackwood, H. van Horn, S. Visa, E. van der Knaap. *Using Hadoop to Find Transcription Factor Binding Sites in the Solanum Lycopersicum Genome*. In Proceedings of the Midstates Conference for Undergraduate Research in Computer Science and Mathematics, Ohio Wesleyan University, OH, pages 69-75, 2013.
- [4] O. Bembom, S. Keles, M.J. van der Laan (2007). *Supervised Detection of Conserved Motifs in DNA Sequences with cosmo*. Statistical Applications in Genetics and Molecular Biology: Vol. 6: Iss. 1, Article 8. <http://www.bepress.com/sagmb/vol6/iss1/art8>, [Accessed on July 2012].
- [5] MEME, <http://meme.nbcr.net/meme/>, [Accessed on July 2012].
- [6] Tobias Marschall and Sven Rahmann. *Efficient exact motif discovery*. Bioinformatics (Proceedings of ISMB), 25(12):i356–364, 2009. DOI: 10.1093/bioinformatics/btp188, [MoSDi accessed on July 2012].
- [7] Timothy L. Bailey, Mikael Bodén, Fabian A. Buske, Martin Frith, Charles E. Grant, Luca Clementi, Jingyuan Ren, Wilfred W. Li, William S. Noble. *MEME SUITE: tools for motif discovery and searching*. Nucleic Acids Research, 37:W202-W208, 2009.
- [8] S. Anders, W. Huber. *Differential expression of RNA-Seq data at the gene level the DESeq package*. European Molecular Biology Laboratory (EMBL), Heidelberg, Germany, 2012. (bioconductor.org/packages/2.11/bioc/vignettes/DESeq/inst/doc/DESeq.pdf).
- [9] The R Project for Statistical Computing, <http://www.r-project.org/>, [Accessed on December 2013].
- [10] P.J. Cock, T. Antao, J.T. Chan, B.A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M.J. de Hoon. *Biopython: freely available Python tools for computational molecular biology and bioinformatics*. Bioinformatics 2009 Jun 1; 25(11) 1422-3. doi:10.1093/bioinformatics/btp163 pmid:19304878. PubMed HubMed (biopython.org).

Dagmawi ZEGEYE
The College Of Wooster
Department of Mathematics and Computer
Science
1189 Beall Avenue
Wooster, OH, 44691
U.S.A.
E-mail: dzegeye16@wooster.edu

Sofia VISA
The College Of Wooster
Department of Mathematics and Computer
Science
1189 Beall Avenue
Wooster, OH, 44691
U.S.A.
E-mail: svisa@wooster.edu

Esther VAN DER KNAAP
The Ohio State University
Department of Horticulture and Crop
Science
204 A Williams Hall
Ohio Agricultural Research and
Development Center (OARDC)
1680 Madison Avenue
Wooster, Ohio, 44691
U.S.A.
E-mail: vanderknaap.1@osu.edu

Workflow management using QR codes

Rusu Victor Gheorghe

Teacher Coordinator: Ralf Fabian

Abstract

This paper presents an application, aiming to more targets, the main target being the management of objects which can be of several types, actually represents a workflow for people who work with them. The application can be seen as a combination of several technologies including C#, Java, PHP, all of them aiming to facilitate the acquisition, storage, and retrieval of information about a particular object. The application consists of three parts: A desktop type module which was designed in C #, an another one that was developed for the Android operating system, which was implemented in Java, and the last module is a Web Service that has been developed in PHP and whose primary purpose is to manage the flow of data input or output and the information in the database.

As a result the application represents a modular system, communication between the three modules taking place via a serialized information flow, this kind of communication representing the link between the three programming languages. As a result the purpose of this application is to reduce the time spent for processing an object through the optimization and facilitation for storage and extraction of information about the subject in question.

1. The management of information

Application has been created based on the multiple programming languages such as PHP, JAVA, C Sharp. Although all of these applications are developed on the basis of different programming languages, they work together for a better management of objects. Enumeration of objects will be performed using type QR codes that will be automatically generated and what are the unique type, which are stored in the database as well as information about the objects processed. The representation type QR codes, is a two-dimensional (2D) representation and interpretation of these codes are being made on the basis of the two dimensions as we can see in figure 1. On the basis of the information, it will perform some operations of the web service in the data, they can be of many types as follows: extraction, insertion, modification and deletion of data.

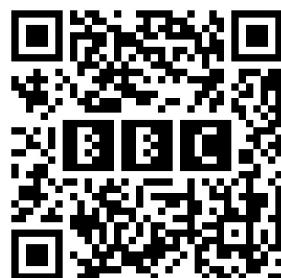


Figure 1- Example of imagine encoded as QR ¹

Relevant information about a specific objects can also be accessed remotely via web service that makes it possible to extract information from the database, although we are not connected to the local network.

¹ Figure 1 - <http://2d-code.co.uk/images/bbc-logo-in-qr-code.gif>

2. About QR code

This type of encoding, namely QR comes from Quick Response, and it appearing in 1994. This encoding type having been created by the Japanese company Denso Wave. By doing this, the company aimed to create a decoder faster compared to the other existing decoders at that moment.[1] To better understand how QR coding work, we can see a representation of the encoding mode, used for type QR images in Figure 2. The QR code represent in fact an package of the standards that is used to translate an information in QR image.

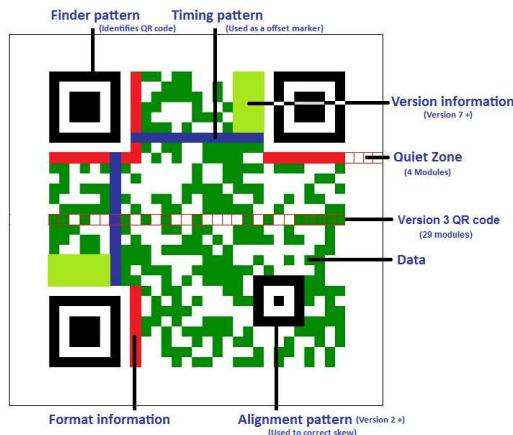


Figure 2 – How QR coding work²

On the picture we observe very clear as it contains the various types of structures, some of this are used to assist recognition program, to determine the information contained in these. Why I chose this type of identification for the workflow object? Because it is a very flexible and can be used on more versions of operating systems, such as desktop applications designed for Windows and not only. At the same time this way is very encoding used on Smartphones and tablets, these applications being created for systems such as Android, Windows Phone, etc.

3. Structure of the applications and how is work

As I said earlier, the application is structurated into three sections, each of them requiring certain resources to be able to carry out the operations required in the current context. Working principle of application in general is relatively simple, because it was created with this purpose, namely to ease the work of a person due to the fact that the processing of information about a particular object will make it very easy. As a result, we will take it as a particular case a library where there is a large amount of the book have not been inventoried. Considering the cards as we actually got some objects, they actually represent the workflow that you will need to interpret information.

In this case, we see put in the situation to sort through a large amount of objects, in a very short time. That situation could be solved very easily by using the application presented above that we use each section of the application for the purpose for which it was created.

² Figure 2 - http://www.opentiersystems.com/wp-content/uploads/2011/09/qrcode_overview4.png

3.1 The desktop application

Section that will run on the desktop, as part of the project, is in fact an application created in C Sharp programing language, aiming to manage the information that will be collected. This information will need to be first processed by an operator, just after they have been processed will be stored in the database as the final information about a particular item, in our case we refer to the books from the library. Also, in the framework of this application we will be able to extract information about an object through a friendly interface in which we will be able to find an object after more features or we can simply scan the QR code on the object about which we want to find out more information. To be able to scan the QR codes within the desktop-like application I used more than one library, and through them the application can retrieve the data contained by a picture of the QR Type. Libraries that are used is AForge.Net and ZXing.

AForge.NET library is a library specific to C Sharp programming language, being in fact a framework used in several areas such as: image processing, neural networks, genetic algorithms, fuzzy logic, machine learning, robotics, etc. [2] On the basis of the above AForge.NET, takes the video signal from a video camera, then sends video signal to the second library that is called ZXing. The ZXing library is a library that can be implemented on multiple operating systems and that aims to decode the QR image formats, with which the library works as follows: UPC-A and UPC-E, EAN-8 and EAN-13, Code 39, Code 93, Code 128, ITF, Codabar, RSS-14 (all variants), RSS Expanded (most variants), QR-Code, Data Matrix, Aztec (beta quality), PDF-417 (alpha quality) [3]

Through this library, we will decode the QR images based on images taken from a video source. The result received from interpretation of QR image will be sent via the HTTP protocol to a web service that will take over. Based on this unique code, the web service will try to identify the object in the database. If the object has been identified the data about him will be returned to the desktop application to display it. An exemplification of how its work the desktop application, can be show on the figure 3.

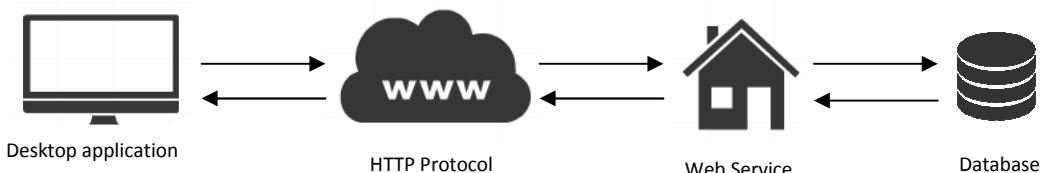


Figure 3 – Example of how its work the desktop application

3.2 The Android application

The application developed in Java for operating systems like as Android can be seen as a portable scanner through which you can glean information about certain objects. At the same time using this application you can retrieve data about the objects that have been added in the database based on the QR code. To retrieve unique code from an image I used a library that is called Zbar because this library having support for Android applications. Zbar is an open source software suite used for reading bar codes from various sources. It supports many types of bar codes including: EAN-13/UPC-A, UPC-E, EAN-8, Code 128, Code 39, Interleaved 2 of 5 and QR Code. [4] In the case shown above, the information that will be retrieved are about the books from a library. One way in which the application can be used to work efficient is the following: Knowing in this case, as all the books from the library must be inventoried, we will be able to use both applications. So we are going to use the desktop application and Android application to optimize the workflow. So that, a person will be able to retrieve the information about the books from the library while the other will be able to process the data received from the first person, as is shown in the figure 4.

Because the application must communicate with the web service, information regarding the objects being processed will be transmitted using kSOAP 2 class. I used this class because like the Zbar class provides support for Android applications. The kSOAP2 is a lightweight library for use in constrained devices. It is not meant to be an all encompassing do-it-all library. However it is meant to interoperate with

most popular SOAP engines . The class of kSOAP2 is made up of an XML parser, a de/serializer, and a transport layer.[5]

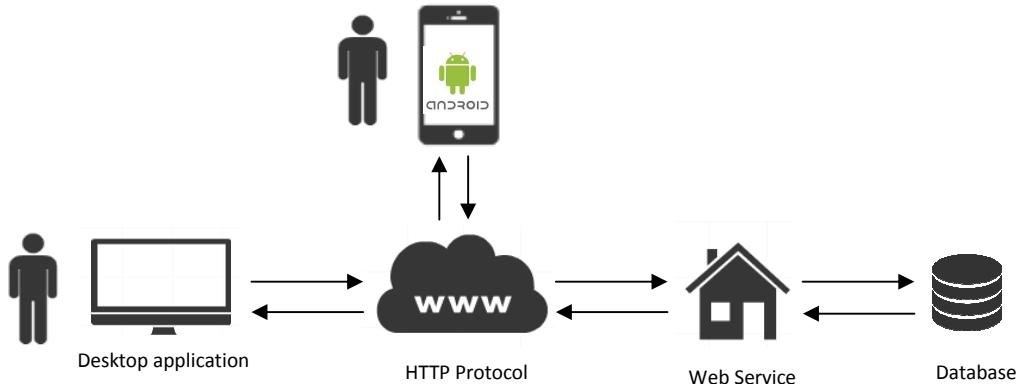


Figure 4 – Optimizing the work

3.3 The Web Service

The web service is an important part of the project, which is responsible for managing all the data. Through it, other applications will retrieve, insert, modify, and delete the information from the database. Also, through this web service information about a particular object can be retrieved from a distance, the only condition is to have an internet connection and data required for logging. The web service is developed in the programming language PHP, and to be able to use this application as a web service I used a class created in the same programming language, which is called NuSOAP.

The NuSOAP is a rewrite of SOAPx4, provided by NuSphere and Dietrich Ayala. It is a set of PHP classes that allow developers to create and consume web services based on SOAP 1.1, WSDL 1.1 and HTTP 1.0/1.1.[6] In terms of use, the web service is very easy to use, because it generates an xml file that will represent the web service description. Based on this file, applications will create their own proxies that will be used to communicate with the server. For a better example of how is represented an web service, see figure 5.

```

▼<message name="GetPhotoListRequest">
  <part name="email" type="xsd:String"/>
  <part name="password" type="xsd:String"/>
</message>
▼<message name="GetPhotoListResponse">
  <part name="result" type="xsd:String"/>
</message>
  
```

Figure 5 – Sequence of Web Service representation

4. References

- [1] http://ro.wikipedia.org/wiki/Cod_QR
- [2] <http://www.aforge.net/framework/>
- [3] <https://code.google.com/p/zxing>
- [4] <http://zbar.sourceforge.net/index.html>
- [5] <http://kobjects.org/ksoap2/index.html>
- [6] <http://sourceforge.net/projects/nusoap/>

Rusu Victor Gheorghe
 "Lucian Blaga" University of Sibiu
 Department of Mathematics and Informatics
 Str.Dr.I.Ratiu, No.5-7, Sibiu, 550012,Romania
 ROMANIA
 E-mail: victor-gheorghe.rusu@ulbsibiu.ro

**Fourth International Students Conference on Informatics
Imagination, Creativity, Design, Development
ICDD 2014, May 15-17
Sibiu, Romania**

J2XML Metro Converter

Gyunesh Shefkedov

Teacher Coordinator: Senior Assistant Valentin Velikov

Abstract

We live in a world where life without a computer would be impossible. But real architects of this modern world – software developers need more and more flexibility in software development process. The presented metro application J2XML (converter) helps developers to convert java code to xml and vice versa in windows 8/8.1.

1. Introduction

Microsoft released Windows 8 for general availability on October 26, 2012. Despite many user's complaints and doubts about the functionality of metro-based operating system and its applications, Microsoft maintain that this is the new style of their operating system and Metro is the future. These disputes affects negative to software developers because after a year and a half there are still many gaps in windows application store. One of these gaps is lack of software development tools in Metro. J2XML is a solution for very small part of that gap. In addition to useful functionality, J2XML has user friendly interface and optimized structure.

What would be the life of a developer in the next 5 to 10 years after the desktop environment is fully discharged or already unnecessary and nonfunctional? What would happen to the software engineers who will need a program like J2XML which is an irreplaceable tool in the software development process? This applications proves how simple application can be one of the most popular products - not only in the software engineering but also in the Java designing forms.

There are variety of parsers as J2XML in internet, but none of them works in Metro. Web applications which allows this kind of code conversion require an internet connection. For now, J2XML is the only option for developers to convert their code from java to xml and vice versa in Metro. J2XML gives more flexibility to developers in the software development process.

2. Parsing from java to xml / xml to java

2.1 Developers need a parser

Why developers need parsers from java code to xml and vice versa? There are many software systems for automation of engineer's work. Most of them require data transfer between software products of different companies. Some of them are open source or free, but others are too expensive or web based. But the price is not the only obstacle for developers. Actually the main obstacle is, that these apps use unknown file formats for data internal machine representation.

Sometimes developers have limited resources for full development of big software products, that's why, they split the program to small parts (modules). For modules which are not developed, programmers rely to different external subsystems. To solve these kind of problems, many of big software products support import/export of data in widely used text format as xml.

Why developers need their own parser? During the development of the main system, developers need to have the source code of the subsystems. Moreover – XML is an open source format which allows every developer to create his own tags, and after that these tags will be recognized by the relevant software modules.

2.2 Main functionality

2.2.1 Loading Application

When J2XML launch for a first time the app loads a settings page which allow users to customize the program according to their needs.

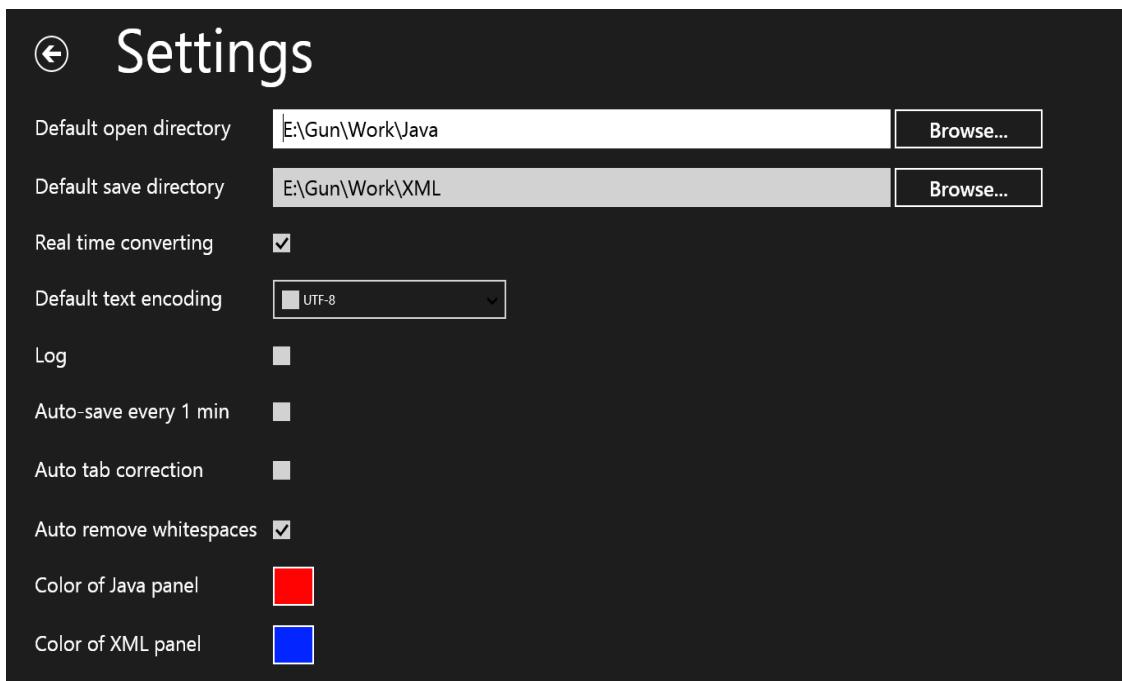


Fig. 1. Startup settings page

After launching the program users have several options to customize:

- Users can set the default open directory to find easily the *.java or *.xml file to load. It can be workspace directory or any other folder in the computer. Similarly users can set the “default save directory”.
- Users can choose whether the J2XML convert in real time. This option would be appropriate for small pieces of code.
- Users can set the text encoding, usually it is an utf-8, but sometimes developers prefer to specify the exact encoding.
- The next 2 options are for safety reasons. “Log” and the “Auto-save every 1 min.” help developers to trace their activity in J2XML and to auto-save the result after parsing
- And finally 4 options to customize the appearance of J2XML.

In J2XML home page, there are two text panels - for java code and xml.

- In “Java code” panel users can write, copy/paste or load *.java file. And if the conversion is from xml to java in the “Java code” panel will be printed the converted code.
- In “XML” panel users also can write, copy/paste or load *.xml file. And if the conversion is from java to xml, there will be printed the converted code.

2.2.2 Opening an existing *.java/*.xml file

At the bottom of the home page there is an app bar (application bar), which contains 5 buttons. The first one (left to right) is to open/load an existing *.java or *.xml file. When users press the “open” button they automatically will be redirected to the “default open directory”, but this doesn’t mean that they can’t load a file from any other directory.

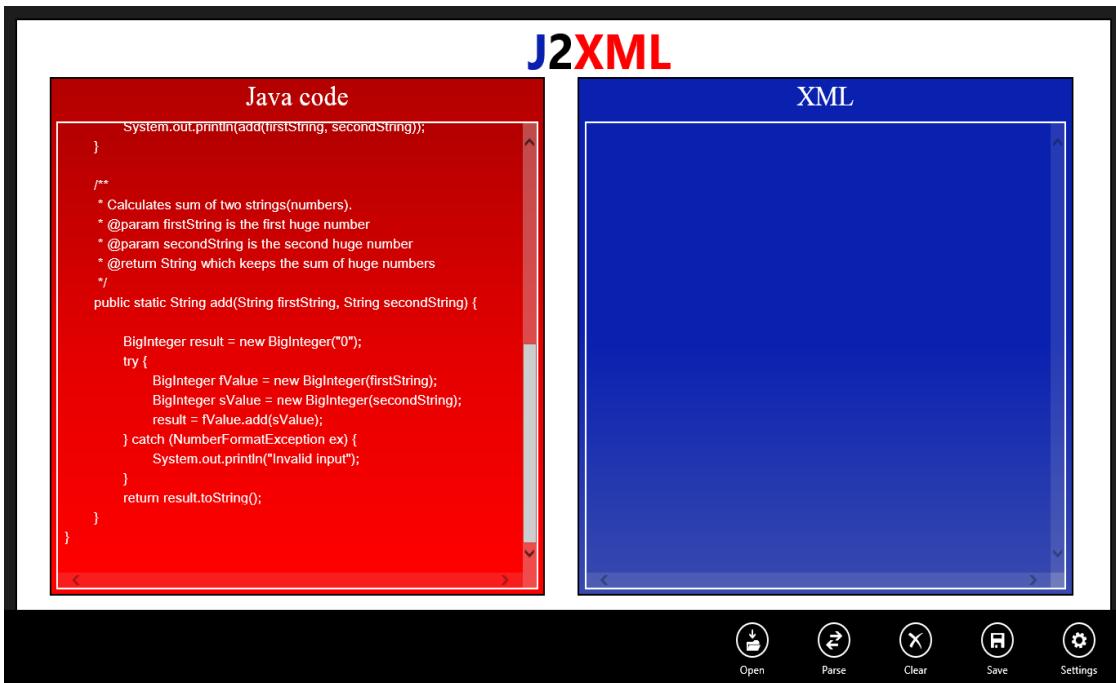


Fig. 2. Loading an existing *.java / *.xml file

After user opens a *.java file the content of *.java file will be printed on the “java code” panel. Whether the text will be formatted or not depends on “auto tab correction” settings. If it is “on”, that means printed text will be formatted and easy to read and modify. The other way text will be printed in one line. “Off” option is appropriate when the user is interested in only from the result. Result text is always formatted.

2.2.3 Parsing/Converting – Java2XML & XML2Java

Parsing from/to java code is the main function of this converter. Converting java code to xml is so important because users can use xml in other application to modify it or to add functionality then re-parse the xml to java.

Why XML? XML (Extensible Markup Language) [2] is some kind of text file which is standardized and widely known. It can be open with any text editor. XML is similar to HTML, but it is more flexible. XML has no restriction of tags and it has simple and logical structure. Other advantage of XML is the possibility to manipulate many programming languages.

Why Java? Java is a computer programming language that is concurrent, class-based, and object-oriented and specifically design to have as few implementation dependencies as possible [3]. Despite C# popularity increases there are still many software developers who prefer Java. The main advantages of Java are: Java Development Kit, Java Runtime Environment and most popular IDE (Integrated Development Environment) are open source or free. Other advantage of Java is that it is intended to let software engineers “write once, run everywhere” [3], which means code that run in one platform does not need to be recompiled to run on another.

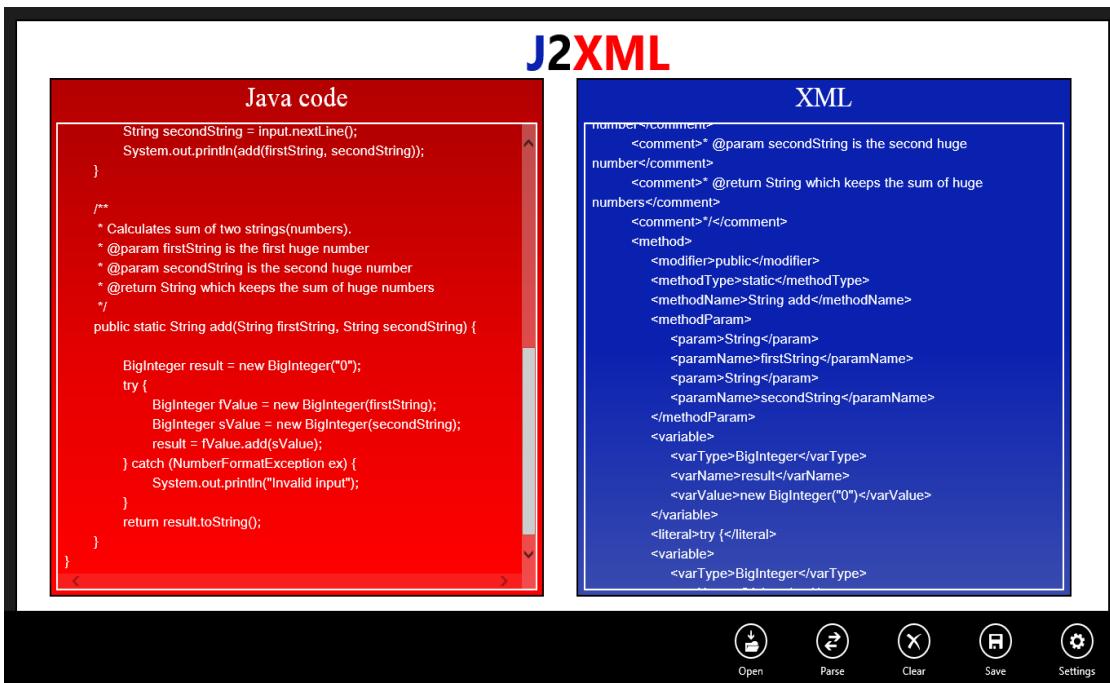


Fig. 3. Converted code java to xml / xml to java

In Fig. 3. It is shown how J2XML looks like after converting/parsing the code. Formatted code helps developers and even non-developers to read the converted code as a “book”.

2.2.4 Remove or Save converted code

The last two buttons (left to right) are “Clear” and “Save” button. Logically after conversion of the code these two buttons on the app bar give users two options to proceed.

- To remove everything and start all over again. This function can help users when they load the wrong file to the panel, or when they just want to see the structure of their java program.
- To save the java/xml file in the default save directory or any other folder for later use. This is very useful because users don’t need to open another text editor, copy/paste the converted code and save it.

3 Conclusion

In the next few years when “desktop” will be gone, software developers will need more and more software development tools in Metro. This new platform will be a true test for experienced software engineers. J2XML can be very useful tool during these hard times.

The author plans expanding the capabilities in the following areas:

- Add new function to modify XML in class diagrams
- Load java/xml files from internet (SkyDrive, Google Drive etc.)
- Exporting converted code to cloud

References

- [1] Developing apps with HTML:
<http://msdn.microsoft.com/en-us/library/windows/apps/br229565.aspx>
- [2] XML Wikipedia: <http://en.wikipedia.org/wiki/XML>
- [3] Java Wikipedia: [http://en.wikipedia.org/wiki/Java_\(programming_language\)](http://en.wikipedia.org/wiki/Java_(programming_language))

Gyunesh Shefkedov
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
gunesh.shefkedov@gmail.com

List of authors:

Cristian-Ionel ALDULEA

Transilvania University of Brasov
Departament of Mathematics and Informatics
Str.Juliu Maniu Nr. 50 Brasov , 500091
Romania
E-mail: alduleacristi@yahoo.com

Atanas ANASTASOV

University of Rousse
Natural Sciences and Education Department
Studentska Str. 8
BULGARIA
E-mail: atanas.y.anastaasov@gamil.com

Adrian BĂRBOS

Tehnical University Cluj-Napoca
CUNBM – Faculty of Science
Department – Computer science
76 Victoriei
ROMÂNIA
E-mail: adrian.barbos@tencadis.ro

Paul - Gheorghe BARBU

“Lucian Blaga” University of Sibiu
Department of Computer Science and Electrical Engineering
10, Victoriei Bd., Sibiu, 550024, Romania
ROMANIA
E-mail: barbu.paul.gheorghe@gmail.com

Petrica BOTA

“Lucian Blaga” University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012,
ROMANIA
E-mail: petrica.bota2@yahoo.com

Andrei CĂLBĂJOS

Tehnical University Cluj-Napoca
CUNBM – Faculty of Science
Department – Computer science
76 Victoriei
ROMÂNIA
E-mail: andrei.calbajos@tencadis.ro

Cosmin Constantin CALIN

“Lucian Blaga” University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012,
ROMANIA
E-mail: calin.cozmin@yahoo.com

Georgi CHAVDAROV

University of Ruse
8 Studentska str., POB 7017, Ruse,
BULGARIA
E-mail: powerbsd@abv.bg

Radu CIGMAIAN	“Politehnica” University of Timișoara Computer and Software Engineering Department 2, Piața Victoriei Street, 300006 Timișoara, ROMÂNIA E-mail: radu.cigmaian@gmail.com
Bogdan CIMPOESU	“Politehnica” University of Timișoara Computer and Software Engineering Department 2, Piața Victoriei Street, 300006 Timișoara, ROMÂNIA E-mail: cimpoesu.bogdan@gmail.com
Octavian-Gorge CIMPU	Transylvania University of Brașov Departament of Mathematics and Informatics Str.Iuliu Maniu Nr. 50 Brasov , 500091 ROMANIA E-mail: octavian.cimpu@gmail.com
Mykhailo DOROKHOV	Simon Kuznets Kharkiv National University of Economics Department of Information Systems 9A, Prospect Lenina, Kharkiv UKRAINE E-mail: michaeldorokhov@gmail.com
Ionuț DRAGOMIR	Application author Petroleum-Gas University of Ploiești Information Technology, Mathematics and Physics ROMANIA E-mail: ionutz_dimy@yahoo.com
Diana Maria DUCA	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: duca.diana92@yahoo.com
Bogdan-Mihai FIERARU	Transylvania University of Brașov Departament of Mathematics and Informatics Str.Iuliu Maniu Nr. 50 Brasov , 500091 ROMANIA E-mail: fieraru_bogdan2000@yahoo.com
Andrei George FLOREA	Article & application author Petroleum-Gas University of Ploiești Information Technology, Mathematics and Physics ROMANIA E-mail: george91ro@yahoo.com
Teodora-Vanessa GLIGOR	Transilvania University of Brasov Departament of Mathematics and Informatics Str.Iuliu Maniu Nr. 50 Brasov , 500091 ROMANIA

Ionut HODINA	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: ionut.hodina@gmail.com
Mariia HONCHAROVA	H.S. Skovoroda Kharkiv National Pedagogical University Department of Roman Languages 29, str. Artem, Kharkiv UKRAINE E-mail: mariehoncharova@gmail.com
Daniela ILIE	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: daniela.ilie@studentpartner.com
Sorina-Gabiela IRIMIE	Transylvania University of Brașov Departament of Mathematics and Informatics Str.Iuliu Maniu Nr. 50 Brasov , 500091 ROMANIA E-mail: sorina_irimie@yahoo.com
Delia-Elena IUGA	Transilvania University of Brasov Departament of Mathematics and Informatics Str.Iuliu Maniu Nr. 50 Brasov , 500091 ROMANIA
Milos JORDANSKI	University of Belgrade Faculty of Mathematics Studentski trg 16,Belgrade SERBIA E-mail: jordanski90@hotmail.com
Gerhard KONNERTH	„Samuel von Brukenthal“ National College Piața Huet, 550182 Sibiu ROMANIA E-mail: gerhard.konnerth@yahoo.de
Alexandru Marian LITĂ	Application author Petroleum-Gas University of Ploiești Information Technology, Mathematics and Physics ROMANIA E-mail: alexandrumarianlita@gmail.com
Narine MANASYAN	Vladimir State University Management and Computer Science in Technical and Economic Systems Gorky St., 87, Vladimir RUSSIA E-mail: narinemanasyan@mail.ru

Constantin Ionuț MARICA	Application author Petroleum-Gas University of Ploiești Information Technology, Mathematics and Physics ROMANIA E-mail: m_costyn_i@yahoo.com
Lucian OPREA	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: lucian_opreaa@yahoo.com
Istvan Benjamin PALLO	Faculty of Science “Lucian Blaga” Department of Mathematics and Informatics Str. Dr. I. Ratiu, No. 5-7, Sibiu, 550012, ROMANIA E-mail: eu.stefy@yahoo.com
Nebojsa T. PERIC	University of Belgrade Faculty of Mathematics Studentski trg 16, Belgrade SERBIA E-mail: nebojscha.perich@gmail.com
Kristina PETKOVA	Technical University Sofia Faculty of Industrial Technology bul.Kliment Ohridski 8 Sofia 1756, Bulgaria E-mail: kristinapetkova.tu@gmail.com
Octavian Florin PETRESCU	University of Pitesti Department of Mathematics and Informatics Str. Targul din Vale nr. 1, Pitesti, Arges ROMANIA E-mail: petrescu.octav@gmail.com
Victor Gheorghe RUSU	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: victor-gheorghe.rusu@ulbsibiu.ro
Robert SANDICA	“Lucian Blaga” University of Sibiu Faculty of Science Department of Mathematics and Informatics Dr. I. Ratiu St., No. 5-7, Sibiu, 550012, ROMANIA E-mail: novaspace_robi@yahoo.com
Radmila SEKULIC	University of Belgrade Faculty of Mathematics Studentski trg 16, Belgrade SERBIA E-mail: radmilasekulic90@gmail.com

Gyunesh SHEFKEDOV	University of Ruse Department of Informatics and Information Technologies 8, Studentska str., Ruse 7017 BULGARIA E-mail: gunesh.shefkedov@gmail.com
Emanuel SILIVĂȘAN	„Samuel von Brukenthal“ National College Piața Huet, 550182 Sibiu ROMANIA E-mail: emanuel.silivasan@hotmail.com
Maria SOKOLOVA	Vladimir State University Faculty of Information Technology Department of Management and Informatics in technical and economic systems Str. Gorky, 87, Vladimir RUSSIA E-mail: sok.masha2011@yandex.ru
Iulia ȘTIRB	“Politehnica” University of Timișoara Computer and Software Engineering Department 2, Piața Victoriei Street, 300006 Timișoara, ROMÂNIA E-mail: julia_s24@yahoo.com
Marius-Alin ȚĂPORDEI	University of Craiova Department of Informatics Craiova, St. A. I. Cuza No.13 ROMANIA E-mail: alin.tapordei@yahoo.com
Vasil Ivanov TSOLOV	Technical University of Sofia Department: Computer Systems Technical University of Sofia , 8 Kliment Ohridski St., Sofia-1000 BULGARIA E-mail: vasil.tsolov@gmail.com
Eva TUBA	University of Belgrade Faculty of Mathematics Studentski trg 16, Belgrade SERBIA E-mail: eva.tuba@gmail.com
Kim VELKER	Tehnical University Cluj-Napoca CUNBM – Faculty of Science Department – Computer science 76 Victoriei ROMÂNIA E-mail: kim.velker@tencadis.ro
Vasile-Sebastian VIDREA	Transylvania University of Brașov Departament of Mathematics and Informatics Str.Juliu Maniu Nr. 50 Brasov , 500091 ROMANIA E-mail: vidreasebastian@gmail.com

Dagmawi ZEGEYE

The College Of Wooster
Department of Mathematics and Computer Science
1189 Beall Avenue, Wooster, OH, 44691
U.S.A.
E-mail: dzegeye16@wooster.edu

Co-organizer:



IQUEST Romania

Organized with support of Romanian Ministry of National Education

SPONSORS



EBS Romania – Expectations Beyond Software



ROPARDO



Global Solutions for Development



Top Tech



PAN FOOD