

**Silesian University of Technology**

**Faculty of Automatic Control, Electronics   
and Computer Science**

##### Final Project

##### (choose appropriate)

Implementation of face detection algorithms in video sequences

Author: Wojciech Mazur

Supervisor: PhD Michał Staniszewski

Consultant: PhD Name Surname

Gliwice, January 2019Załącznik Nr 2 do Zarz. Nr 97/08/09

**Oświadczenie**

Wyrażam zgodę/nie wyrażam\* zgody na udostępnienie mojej pracy dyplomowej/rozprawy doktorskiej\*

…………….………., dnia …………………………

……………………………………..……………….……

*(podpis)*

……………………………………………………………

*(poświadczenie wiarygodności podpisu przez Dziekanat)*

*\* właściwe podkreślić*

**Oświadczenie promotora**

Oświadczam, że praca „Implementation of face detection algorithms in video sequences” spełnia wymagania formalne pracy dyplomowej inżynierskiej.

|  |  |
| --- | --- |
| Gliwice, dnia ……………………… | ………………..……………….……  *(podpis)* |

Contents

[1. Introduction 1](#_Toc526891482)

[2. [Problem analysis] 5](#_Toc526891483)

[3. Requirements and tools 9](#_Toc526891484)

[4. External specification 12](#_Toc526891485)

[5. Internal specification 15](#_Toc526891486)

[6. Verification and validation 18](#_Toc526891487)

[7. Conclusions 21](#_Toc526891488)

[Bibliography i](#_Toc526891489)

[List of abbreviations and symbols ii](#_Toc526891490)

[Contents of attached CD-ROM iii](#_Toc526891491)

[List of Figures iv](#_Toc526891492)

[List of Tables v](#_Toc526891493)

# Introduction

This chapter contains following elements:

* introduction into the problem domain,
* settling of the problem in the domain,
* objective of the thesis,
* scope of the thesis,
* short description of chapters,
* clear description of contribution of the thesis’s author – in case of more authors table with enumeration of contribution of authors.

According to Maslow’s Hierarchy of needs, safety is one of our most fundamental needs. Without it, it is hard to think about friends, relationships, accomplishments, or self-fulfillment. Over the last few years it can be observed, that the biological identification is rising in its popularity. There are new ways found to utilize our biological footprints. What we can also notice, is that most of the use cases revolves around security, and for a good reason. Fingerprints are used to unlock locks for decades. Fingerprint scanners have also found a way to consumer electronics. We can’t imagine a world where we can’t secure our smartphone or notebook just with a password. Fingerprint scanners have expended rapidly when it comes to smartphones and made our lives better.

But it’s not the only way we are trying to secure our privacy, including our increasingly valuable resource, which is data. We have developed ways to recognize people by other features like iris, or even from a face in general, using different kinds of sensors. Cameras, infrared light sensors to depth detectors that remember given points on a face. And everything happens within a blink of an eye.

What we want to do apart from protecting our data, is something that we crave to do since beginning of the humanity. We want to protect ourselves. We want to feel safe and comfortable, it is, in fact, required, as stated previously.

One of the main issues arising in recent years, is public safety and detection of dangerous people. We can observe increasing activity of terrorism in a lot of European countries. The awareness of people is also increasing, we are learning that prevention is better that treatment. National safety agencies recognize most of European countries at least on moderate threat of terrorism. A lot of countries are considered even as a high threat for regular travelers. This is something that needs to be addressed as quickly as possible, and we need to work on the technology, that can be effectively applicated for new solutions.

## Goal of the thesis

Main goal of this thesis is to implement face detection algorithms, both with the face recognition. This could provide us with useful tools to analyze and examine, for example video footages that are collected every day on thousands of city cameras around the globe. We can use that existing architecture, not only to look for people that have already broke the law, but also to prevent crime. Simple camera system doesn’t provide us with the possibility of detecting the face. We would need to do that manually which is a complicated and tedious work, that not every person is able to do. The main focus is to accomplish filtering dangerous, or at least for some reason crucial for safety people, implementing algorithms that detect and recognize their faces.

## Scope of work

Main scope of the work contains a few elements that needed to be combined to give full spectrum of the problem. First of all, I needed to choose programming language, that would be easy to use, yet contain all the elements needed to process images, like easily accessible and usable libraries that would greatly accelerate the progress of work. Then, analysis of current solutions, and choice of a few algorithms that comply with what this thesis is trying to achieve and comparison of their effectiveness. Next thing was the actual implementation of the solution in the programming language of choice, but also processing the testing material consisting of video sequences that were analyzed and graphically modified that the human can easily observe effects of the detection and recognition in real time.

# [Problem analysis]

This chapter contains following elements:

* problem analysis,
* state of the art, problem statement,
* literature research (all sources in the thesis have to be referenced [1, 2, 4, 3]),
* description of existing solutions (also scientific ones, if the problem is scientifically researched), algorithms, location of the thesis in the scientific domain.

## Face detection methods

There are a lot of ways to detect given features, including face features, that would allow us to recognize position of the face on a given media sample. Although one that got very popular since its release is based on machine learning approach, from the work of P. Viola and M. Jones [1] [2]. Their work is described as extremely rapid in terms of effectiveness, and this is something that is crucial for effective and real time face recognition. The original study was performed on Intel Pentium III clocked on 700MHz, which is a rather old processing unit considering performance of newest units. This would give over 15fps on 384x288 pixel images. We can compare similar CPUs using tool UserBenchmark [2]. Comparing closest processor that this tool provides us, which is a successor to Pentium III, Pentium 4. Its performance can be taken as comparable, especially in units that were on similar clock speeds [3]. Unit that was used for calculations in this thesis was Intel Core i5 4690k, which is according to UserBenchmark, over ten times faster on average, often going up to twenty times faster and above [2]. This gives us the opportunity to take higher resolution of video samples and photos, and still achieve a very satisfying frames per second rate.

Idea is that if given element, item or other object has a unique shape, we can assume, that it also has unique properties when it comes to light that is projected onto that object. This gives us a tool to check whether chosen parts of an image have lighting that is changing in a way that was previously seen and is recognize for example as a face.

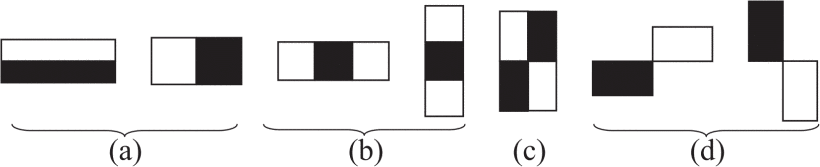


Figure 1Basic Haar-like features [1]

On the Figure 1 we can see what features are used to distinguish between light and dark spots on a picture. Although this can be misleading, what we really want to see if we want to understand is to see the image in a way shown in Figure 2.



Figure 2Haar-like feature on face [5]

So, the colors itself are not important, but the pixel values within. For example, if we want to find a face, we can safely assume, that forehead pixels will be on average brighter than pixels that are within eyes region, due to the shadow dropping from the eyebrows. The same goes mostly for things like nose, where we can assume either vertical line, that is bright and has darker surroundings due to the shadows. But this is also something that we need to be careful with, due to the changing light conditions. We can compensate for that looking for w line that has a darker region only on one side.

# Requirements and tools

This chapter contains following elements:

* functional and nonfunctional requirements,
* use cases (UML diagrams),
* description of tools,
* methodology of design and implementation.

.

# External specification

This chapter contains following elements:

* hardware and software requirements,
* installation procedure,
* activation procedure,
* types of users,
* user manual,
* system administration,
* security issues,
* example of usage,
* working scenarios (with screenshots or output files).

The entire document should contain references to the illustrations contained therein (Fig. 4.1).

|  |
| --- |
|  |
| Fig.4.1. *The variation funkstioni* |

# Internal specification

This chapter contains following elements:

* concept of the system,
* system architecture,
* description of data structures (and data bases),
* components, modules, libraries, resume of important classes (if used),
* resume of important algorithms (if used),
* details of implementation of selected parts,
* applied design patterns,
* UML diagrams.

A short code insertion in the text line is possible, e.g. class Main. Longer fragments should be written in *Courier* or *Courier New* font size 10 in frames (Listing 4.1) with a space between the lines of the value 1. All lines of code should be numbered so that they can be referenced in the text of the document.

Listing 1. Generating random numbers

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | **package** polsl.iinf.lab;  **import** java.util.Random;  **public** **class** Main {  **public** **static** **void** main(String[] args) {  Random r = **new** Random();  // drawing a number from the range 1..10  **int** a = r.nextInt(10 + 1);  System.*out*.println(a);  // drawing a number from the range -5..15  System.*out*.println(r.nextInt(21) - 5);  }  } |

# Verification and validation

This chapter contains following elements:

* testing paradigm (eg. V model),
* test cases, testing scope (full / partial),
* detected and fixed bugs,
* results of experiments (optional).

# Conclusions

This chapter contains following elements:

* achieved results with regard to objectives of the thesis and requirements,
* path of further development (eg. functional extension . . . ),
* encountered difficulties and problems.

# Bibliography

|  |  |
| --- | --- |
| [1] | Name Surname, Name Surname. *Webpage title*. http://adres/w/sieci.html [access date: 2018-09-30]. |
| [2] | Name Surname, Name Surname. *Title of a book*. Publisher, Warsaw, 2017. |
| [3] | Name Surname, Name Surname. Title of an article in the journal. *Journal title*, 157(8):1092–1113, 2016. |
| [4] | Name Surname, Name Surname, Name Surname. Title of a conference article. In *Conference title*, pages 5346–5349, 2006.  . |

# List of abbreviations and symbols

|  |  |
| --- | --- |
| *DNA* | deoxyribonucleic acid |
| *MVC* | model–view–controller |
| *N* | cardinality of data set |

# Contents of attached CD-ROM

The thesis is accompanied by a CD-ROM containing:

* thesis (PDF file),
* source code of applications,

# List of Figures

# List of Tables