NTNU – Trondheim
Norwegian University of
Science and Technology

Support of remote medical diagnostics using HTML5

Possibility to detect changes in the facial functions caused by face paralysis in order to evaluate therapy□s effectiveness or find some new abnormalities.

Alicja Kwasniewska

Master of Science in Electronics
Submission date: July 2015
Supervisor: Ilangko Balasingham, IET

Norwegian University of Science and Technology
Department of Electronics and Telecommunications

To Jacek Rumiński who taught me that "only as high as I reach can I grow, only as far as I seek can I go, only as deep as I look can I see, only as much as I dream can I be."

-Karen Ravn

To Ilangko Balasingham who showed me that "there are so many doors to be opened, and I should be not afraid to look behind them."

-Elizabeth Taylor

To my parents, my sister, my brother and my friends who helped me become the person I am.

Abstract

In this project I propose to cast the problem of remote patient monitoring into an application available on every device that supports a web browser. The research was done with taking into account the aim of the project which was focused on checking whether it is possible to detect changes in the facial functions caused by the facial paralysis. The motivation for this work was to provide algorithms for detecting geometrical and colours differences between two halves of the face for patients who suffer from facial paralysis. The features were collected during making sequence of special facial gestures. Moreover I took advantage of existing solutions to magnify local colour and motion changes. This combination may uncover new dependencies between data set and lead to conclusions that were previously undiscovered.

In my thesis, I focused on analyzing different factors that can have influence on the processes of the extracting functional and geometrical features. The research I present aims at reviewing face, eye and mouth recognition performances in function of: distance from/to a camera, hardware used to decode information, personal features, head rotation and light conditions. Moreover, I also analyze accuracy of counting distances between specific facial points and colours differences between two halves of the face for the same parameters.

Considering possible signs, symptoms and diagnose methods of facial paralysis, I present a fully functional application, which can potentially provide a faster access to medical care and help to evaluate effectiveness of the rehabilitation methods. I believe that it may be very useful, because the telemedicine allows to reduce health care costs and give medical institutions faster access to information about their patients.

Additionally, during the research the restrictions of the measurement conditions affecting the reliability and repeatability of the results were examine and the

application was tested on the group of potential users. My first conclusions are the facts that following factors may have influence on implemented algorithms:

- distance from the camera
- rotation and slope of the head
- personal features
- light conditions

Under preliminary studies, using different devices did not affect the results.

Based on the results it can be assumed that by analyzing distances between characteristic facial features it is possible to detect changes in the facial functions caused by face paralysis in order to evaluate therapy's effectiveness or find some new abnormalities. For colours differences it is much more difficult to formulate a clear diagnose as this parameter is influenced by measurement conditions.

In the last part of this thesis, I conclude the project and describe possibilities for future work in this area which include for example combining this application with professional diagnosis methods.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces the dissertation titled *"Support of remote medical diagnostics using HTML5/WebRTC"* by presenting its context, motivation, project's objectives, hypothesis and an outline on sections 1.1, 1.2, 1.3, 1.4, 1.5 respectively.

1.1. Context

Face paralysis may be caused by many different factors: infection or inflammation method for recovery of face through measuring the percentage of distance (PD) of certain facial nerve, head trauma or head/neck tumor. It can appear suddenly (Bell's palsy) or intensify over a longer period of time (tumors). It may last long or short and can be reduced to a certain degree by performing specially defined exercises. However very often it is hard to decide whether actual therapy helps or not and whether it should be modified. In this project I would like to cast the problem of monitoring the effectiveness of the rehabilitation therapy into the application which combines advantages of telemedicine and exisiting metrics which allows to determine the state of the patient health during facial paralysis.

1.2. Motivation

The motivation for this work is to provide algorithms for detecting geometrical and colours differences between two halves of the face during making sequence of special facial gestures and taking advantage of existing solutions to magnify local colour and motion changes. This combination may uncover new dependencies between data set and lead to conclusions that were previously undiscovered. The described project will result in a proposal of new scientific solutions which allows to make the process of matching therapy to the patient easier and faster. Many new aspects analysed and defined in this paper may turned out to be important in the future researches and significant for development in many areas.

Since it is a contact-free way and a faster method of assessing information about facial features than traditional methods, this research project has potential for advancing fields like telemedicine, personal health-care, and ambient assisting living.

Due to using HTML5 technology, the application is very useful as it can be launched on many different devices.

1.3. Objectives

In this project an application for monitoring effectiveness of a rehabilitation therapy will be developed and implemented. It will consist of two main parts:

1. client side
2. server side

The client side will have to fill following requirements:

- works on every device which supports a web browser in a proper version
- sends data to a server
- displays feedback which includes detected colours and distances in real time

The server side will have to fill following requirements:

- receives data from a client
- detects geometrical and functional facial features based on the Eulerian Video Magnification method and proposed algorithms
- send back processed frames

In order to accomplish that, the feasibility of algoritms for extracting facial featuresand magnifying colours and motions have to be tested. Moreover, the application performance should be evaluated through tests perofrmed with several patients.

1.4. Hypothesis

As already mentioned, the aim of the project is to develop new algoritms for obtaining both structural and functional features of the face. Moreover it is important to examine the restrictions of the measurement conditions affecting the reliability and repeatability of the results. The following hypothesis is defined assuming controlled measurement conditions and the use of frames sequences captured from a webcam: *possibility to detect changes in the facial functions caused by the facial paralysis in order to evaluate therapy effectiveness or find some new abnormalities* As a consequence, both geometrical and behavioral features which are related to the way the face changes over time will be examined. It is important to remember that these features are characteristic for each individual and implicitly represents changes in facial geometry and function.

1.5. Outline

The rest of the document is structured as follows:

1. **Chapter 2** - Analysis of the state of knowledge in the field of: video registration in web browsers, methods of measuring chosen diagnostics parameters using remote cameras, evaluating effectiveness of these methods.

2. **Chapter 3** - Methodology, choice of algorithms for automatic analysis of facial expressions and analysis of the possibilities of their realization in HTML5 WebRTC technology.

3. **Chapter 4** - Design of the system, aim of the project, potential users, system requirements.

4. **Chapter 5** - Description and implementation of the system.

5. **Chapter 6 -** Development of the test scenarios, performing the tests and conducting the experiments in different measuring conditions. Description of the achieved results.

6. **Chapter7 -** Drawing conclusions from the results of work and possibilities of the future work.

Chapter 2

State of the art

2.1. Introduction

The aim of this chapter is to present the main goal that telemedicine should achieve and describe a variety of concrete examples of systems for monitoring patients. Moreover, in the second part of this chapter, the problem of face paralysis is briefly described, including methods of diagnosing and treating patients who suffer from loss of facial movements. Finally, taking into consideration three aspects: geometry, temperature and micro movements of any structures innervated by the facial nerve, the concepts of analyzing and detecting face are explained.

2.2. Telemedicine

Telemedicine or e-health can be defined as various aspects of medical help provided at a distance [1]. Fast growing trend based on using the advanced telecommunication technologies for both providing health care services and exchanging medical data allow to cross many barriers, especially geographic distances or time obstacles. It is also said that telemedicine has a lot of advantages, for example it can provide access to health care for people living in rural areas, lead to reducing health care costs and give medical institutions faster access to information about their patients. There are many ways to help lay people and their caregivers better understand health problems, choose proper health care solutions, cope with a lot of challenges and implications and keep in touch with health care providers. Moreover e-Health systems have been shown to improve quality of life of patients and they can have positive effects on family [2]. Possibilities of patient-focused systems are endless, from websites, portals, tools and reminders to specially created monitoring systems equipped with professional sensors. That is why systems of this kind are becoming more and more popular among not only many medical institutions but also among individuals [3].

2.2.1. Requirements of e-Health

Transformation of health care systems into a wide range of online medical tools raises doubts and it requires re-engineering of clinical care services [2]. Successful implementations of e-health systems requires careful planning and understanding requirements that it should fill. Moreover, while using newly adopted technologies, it is a necessity to understand technology itself [4], [5], because in order to use this technology, a consumer need to be familiar with it. Designers of telemedicine systems should remember about making a product that is easy to use. There are also other factors that characterize success of e-health systems, like organizational environment, motivation, technology usefulness, promotion and health personnel awareness. Each of these factors are connected with meeting special requirements. Yet, among many needs, security and privacy of patients data is one of the most important issues that should be taken into account [6]. Health care providers should especially follow security safeguards [7]. Some basic rules for creating safe systems include principles for:

1. user authentication – the act of establishing or confirming someone as authentic [8]

    (a) avoid possibility to get user enumeration (authentication error message should not point out which credential is wrong)

    (b) avoid possibility to login using guessable (dictionary) user accounts

    (c) avoid possibility to brute force (systematically enumerating all possible candidates for solution) by setting a time required to login again or locking a user who failed to login for a number of times

    (d) not let users to bypass authentication schema

   (e) create carefully password reset mechanism
   (f) automatically log users out, destroy all session tokens, check session state, disallowing to reuse previous session tokens
   (g) prepare multiple factors authentication
   (h) avoid possibility to produce an unexpected result by a system because of race conditions

2. user authorization – concept of allowing access to resources only to those permitted to use them [8]
   (a) avoid possibility to execute path traversal
   (b) not let users to bypass authorization schema
   (c) avoid possibility to modify privileges inside the application by a user

3. system protection
   (a) firewall
   (b) running antivirus

4. safe hardware disposal

What is more, criteria for health information on the Internet are being developed. The Health On the Net Foundation is one of the institutions directly contributed to this developement [3], [9], [10]. These criteria include:

1. Authoritative: necessity to indicate the qualifications of the authors
2. Complementarity: the system should support, not replace relationship between the doctor and the patient
3. Privacy: the system should respect privacy and confidentiality of patient data
4. Attribution: published information should include citation of the sources and date of its access
5. Justifiability: the system has to back up claims relating to performance
6. Transparency: for ex. accurate contact e-mail addresses
7. Financial disclosure: funding sources specified
8. Advertising policy: advertising distinguished from the main content

2.2.2. Integrating telemedicine with WBAN

A wireless body area network (WBAN) can potentially replace traditional health care systems. This technology was introduced to enable real-time monitoring of human vital signals by sensing it with various sensors and transmitting over a wireless channel to a main station [11]. These wireless biomedical sensors can be centrally coordinated or distributed [12]. It is also possible to integrate body area networks on telemedicine systems by specially designed frameworks [13]. In this solution sensors and actuators are connected to the BAN coordinator that serves as a data acquisition center. However instead of processing information locally by base station, it is transmitted to telemedicine systems through the faster network connectivity. Fig. 2.1 illustrates one example of framework which integrates body area network with telemedicine system. However system based on WBAN are more

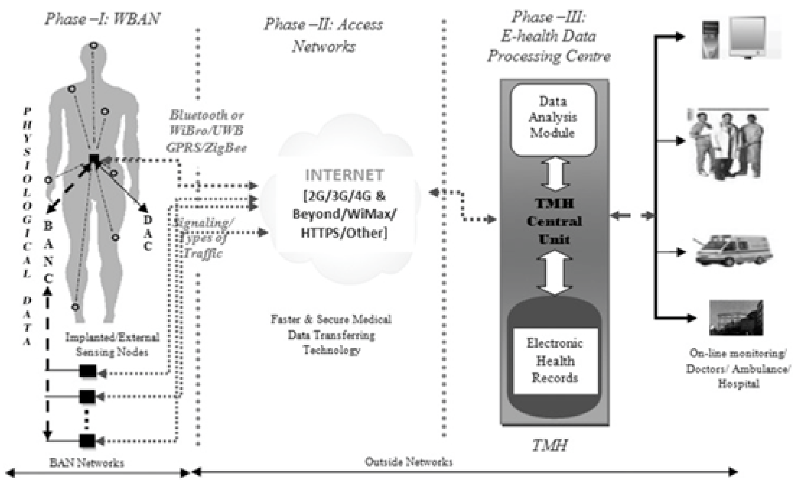


Figure 2.1. Integration of BAN with telemedicine system [13].

complex, need a lot of functional support elements and the price of monitoring patient is growing rapidly. It is caused by the fact that WBAN systems have to face a lot of technical challenges. Usually sensors are not deployed with high redundancy, so the possibility to meet failure should be consider. WBAN require using additional nodes from two categories: sensors/actuators in or on a human body, and router nodes around WBAN wearers [14]. The idea of Remote Patient Monitoring can be an alternative for WBAN.

2.2.3. Remote patient monitoring

The near future promises increase in number and quality of health care services available for individual patients by exploiting solutions from all of the fields of telemedicine and especially from the field of Remote Patient Monitoring, which ensure effective, reliable, and low-cost services [6]. What is more, Information and Communication Technology (ICT) developments allow to monitor patients state of health during their daily activities in a convenient way. Formerly, Remote Patient Monitoring (RPM) has used telephone lines to provide healthcare. Nowadays, technology revolution, high speed Internet infrastructure, and sensor networks make it possible to monitor patients in many other ways, for example by using mobile applications [15].

Usually, RPM systems consists of two main modules which are responsible for: data acquisition and its transmission to a central server. There are a lot of possibilities of gathering specific vital signals, however only a few of them do not require using additional sensors. One of the idea is to build system based on image and video processing. Unfortunately accuracy of these systems may suffer from poor quality of recorded data because professional monitoring equipment, capable of performing this kind of task are not widely available and ordinary equipment may not provide sufficient quality of gathered information. Therefore, light conditions and similar factors have to be considered, while projecting and testing systems which use video recorders.

2.2.4. Example solutions

Some medical institutions managed to take benefit from telemedicine by introducing systems which provide effective care at a low price [16].

1. A system based in Boston allow patients to measure weight, blood pressure and other metrics and send them to Partners. Then, acquired data is analyzed by clinical decision support tool, which decide which patients need intervention [17]. As a result a risk of a heart failure is reduced.

2. Idea of remote patient monitoring, health informatics and disease management technology have been combined in a system created by Veterans Health Administration. The mail goal of this program is to help

patients who suffer from six chronic conditions (from depression to diabetes) [18].

3. Dermatologists in San Diego introduced a system that allows to analyze images and patient information uploaded and sent to them over a secure server. As a result dermatologists do not have to see every patient in person, so a time required to make a diagnosis is shorter (they can handle 50 percent more cases than if they relied on face-to-face visits) [19].

4. At the University of Massachusetts Memorial Medical Center in Worcester eICU critical care nurses and intensivists – physicians have been trained to monitor patients 24/7 from a remote support center. By using advanced telemedicine technology mortality have been reduced by 20 percent. Moreover lengths of stay and care costs decreased by 30 percent [20].

5. Some e-Health systems help patients to remember about their daily medication regimes. A prototype created at the Center for Connected Health use a wireless pill bottle that remind patients to take their blood pressure medication [21].

6. CHESS, the Comprehensive Health Enhancement Support System provides lay people with Internet access. It offers a wide range of functionalities: from education platforms and self-care advises to many professional tools and communication applications, which allow to talk with people who suffer from similar disease. For example in health tracking tool people fill health survey every 2 weeks in order to compare results presented on graphs. CHESS also helps to make decision about therapy by claryfying possible alternatives [2].

As can be seen telemedicine is focused on different aspects of helping people. There are many real-life solutions that have been successfully introduced by medical institutions. Some of these e-health systems take advantage of the possibility to detect facial expressions, body movements and physiological reactions, which are the atomic units of the non-verbal communication. By measuring these parameters, a state of the patient health can be known even without taking with this patient. As a result systems of this kind are really comfortable and people can use them during daily activities. The rest of the chapter is primarily concerned with face analysis in the context of the facial nerve paralysis.

2.3. Face

2.3.1. Function

The face is a highly sensitive region of the body, that is responsible for expression of emotion. Facial appearance may change when the brain is stimulated by any of the five senses [22]. Moreover, face is crucial for human identity, recognition and communication among people [23]. In face-to-face communication only 7% of the communicative message is transferred via linguistic language, 38% via para-language, while 55% via facial expressions [24].

2.3.2. Facial muscles contraction

The muscles of the face play a prominent role in expressing emotion [25]. The role of muscles can be described by neuromuscular activity [26]. The sequence of events leading to contraction is initiated in the central nervous system. In the case of expressing emotions, it is caused by voluntary activity from the brain. The central nervous system creates a nerve impulse and when a muscle fiber is properly stimulated, after getting this impulse through the network of nerves, it generates a tension. Under this tension the muscle provide motion and it may lengthen, shorten, or remain the same length.

2.3.3. Facial nerve paralysis

Classification

Generally, paralysis is a state in which function for one or more muscles is impaired and they can't contract due to the lack of transmitting nerve impulses to a muscle. This state can also be accompanied by a loss of feeling in the affected area. There are four main forms of paralysis:

1. **Flaccid paralysis - neurogenic origin** - caused by damage to the peripheral nerves which conduct nerve impulses from higher neural centers to effectors (muscles) (eg. trauma, post-infectious, tumors)

2. **Flaccid paralysis - myogenic origin** - associated with damage to the muscle or muscle group (trauma, feebleness, myopathy)

3. **Spasticity paralysis** - nervous system, especially the spinal cord may be damaged and can't properly stimulate fibers

4. **Sleep paralysis** - a phenomenon in which a person experiences an inability to move both during , either falling asleep or awakening

Facial nerve paralysis is a disease that involves the paralysis of any structures innervated by the facial nerve, also known as cranial nerve VII (fig. 2.2 [27]).

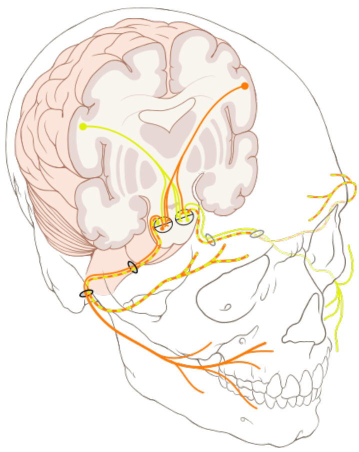


Figure 2.2. Facial nerve

There are a number of causes that may result in facial nerve paralysis [28]. We can divide them into five groups [30]:

1. Congenital (eg. Birth trauma),
2. Infectious (eg. Bell's Palsy),
3. Neoplastic (eg.tumors),
4. Traumatic (eg. injuries),
5. Neurologic (eg. autoimmune diseases)

However no matter what the source of paralysis is, it always leads to discomfort of the patient and decrease in the quality of his life, especially when symptoms appear suddenly as in Bell's palsy. According to the National Institute of Neurological Disorders and Stroke, Bell's palsy is the most common form of facial paralysis [29]. Patient doesn't exactly know when signs of the Bell's palsy may occur. In comparison to face paralysis caused by tumors, symptoms don't intensify over a longer period of time, but appear unexpectedly.

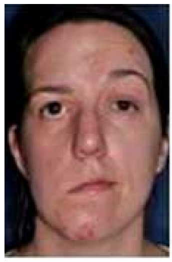


Figure 2.3. Effects of face paralysis on muscle contraction during brow raise [33]

Signs and symptoms

Paralysis can result both in significant psychological and functional disability. For example, we can observe inability to express emotion because of a lack of movement in some parts of the face, usually only on one side of the face (as shown in the Fig. 2.3), causing a drooping mouth, drooling, and excessive tearing from one eye.

Facial weakness during facial nerve paralysis can be also combined with other symptoms including loss of taste, hyperacusis and decreased salivation [34]. What is more it can cause headaches, earaches and difficulty with talking. Taking it into consideration signs and symptoms of face paralysis are vary various, from the impairment of facial expression and difficulties with communication among people to loss of the eye protection [31].

Diagnosis

In 1983 House Brackman proposed a special kind of assessment of facial paralysis - grading system [32]. This system has became a North American standard for the evaluation of facial paralysis [31]. During examination, the patient performs a sequence of specific movements. Then, basing on clinical observation and subjective evaluation, the specialist assigns a grade of palsy ranging from value I (normal) to value VI (no movement).

The main advantage of the HB grading scale is the fact that it is easy to describe facial function in a single figure. However it is based on a subjective judgment and it does not detect regional differences of facial function.

Obviously, there are also other methods that are used for diagnosing face paralysis. These test can help to find the cause of the paralysis and include electromyography, imaging scans, and blood analysis.

However there is still no automatic way of detecting paralysis and that is why most of the examinations are based mainly on observation. As previously mentioned, in order to detect changes in the face animation some characterictic movements that are affected most during face paralysis are used [31], [33]. These movements include for example brow raise, tight eye closure, gentle smile, open mouth or raise nose as presented in the figure 2.4.

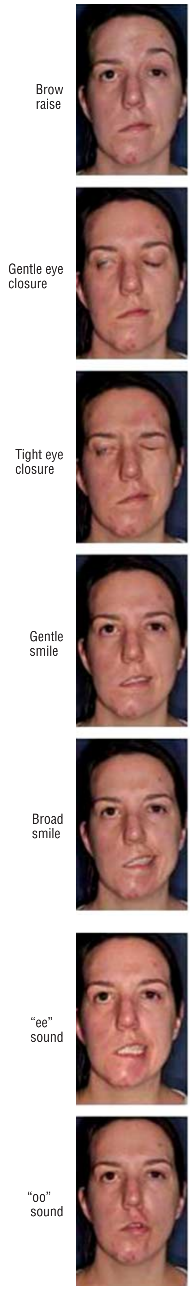


Figure 2.4. Sequence of specific movements[33]

Treatment and rehabilitation

Evaluation of the state of the disease is essential to choose proper medical treatment [35] and investigate the possibility to recover, because the degree of nerve injury determines the success of the therapy [36].

Unfortunately in spite of treatment and rehabilitation some cases of facial paralysis may never completely go away. However there are various methods that can help people who suffer from face paralysis even if they can't recover completely.

Acupuncture is an effective method for threatening face paralysis, but only if a kind of needles is correctly chosen taking into account patient's paralysis degree. Taking steroids and other medications may boost chances of complete recovery.

One of the symptoms of the facial paralysis is loss of eye protection. It is a serious problem that can lead to eye damage. Taking it into account patients should use artificial tears or special lubricant and wear a plastic moisture protecting cover to keep the eye wet. For those patients who suffer from this problem and are not able to recover fully, cosmetic surgery can be the only way to correct their eyelids. Patients may also benefit from other surgeries, for example repairing or replacing damaged nerves or removing tumors.

Yet, it is said that physical therapy and rehabilitation can primarily protect from permanent damage by strengthening muscles. The treatment is based on muscle reeducation using feedback from surface electromyographic (EMG) combined with daily home exercises. At first, abnormal synkinetic muscle activity is decreased and then voluntary movement is increased. Regular exercises are very valuable, because they have been shown to be efficacious in the treatment of facial nerve paralysis [36]. Patient should exercise in short sessions, but repeat the routine 2-3 times a day. Sometimes even after few sessions, the results are promising [37].

2.4. Facial motion and features recognition

Facial expression analysis includes not only measurement of facial motion but also special features recognition. The general approach to automatic facial expression analysis (AFEA) consists of three steps [38] presented in the figure 2.5.

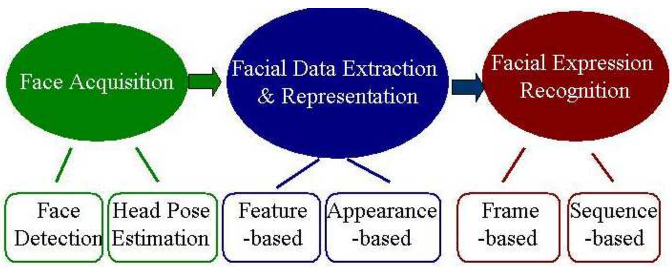


Figure 2.5. Facial expression analysis system [38]

In the first step the face region for the input images or sequences should be automatically found, then facial changes caused by facial expressions are extracted. Finally, the last step is to recognize specific facial expression [38].

2.4.1. Geometry

In this section three main problems are considered: detection of an image segment as a region of interest (ROI), algorithms for improving obtained sequence and extraction of the facial expression information.

Detection of an image segment as ROI

Usually automatic face analysis tools allow to control the conditions under which an image is taken. Most often location of the face in the scene is known a priori, because in many systems to achieve best results, the image has the face in frontal view. However, the scale, orientation and angle of the face may vary between different frames. Finding the exact position of the face and fixed template for the whole system may be difficult [39]. Moreover fitting the model to the real world scene is also challenging because of the possible changes in light conditions and image noise.

In general, determining the exact location of the face can be achieved in two ways [39]:

1. analytic approach - feature based method,
2. holistic approach - template based method,

Analytic approach

In this approach the face is detected by finding some special features of an analytic face model or tracking them in sequences of the images [39]. These features may include for example position of the outer corners of the eyes, the height of the eyes, and the height of the mouth. Automatic tool for extracting these values, normalizing size of the facial area and placing rectangle over the image was proposed in [40] One idea is to take advantage of the brightness distribution data of the face in the monochrome image taken by CCD camera [41]. However, it is said that recognition process can be more effective if we look for features that encode some information about the classes instead of analyzing the luminance and the colour of the radiation obtained from isolated pixel values. One solution for making recognition process more efficient is to take advantage of the Haar-like features. These features allow to encode the presence of oriented contrasts between two regions [42]. The features which can be computed in a short time irrespectively of their position [43] are presented in the fig. 2.6. This method is a machine

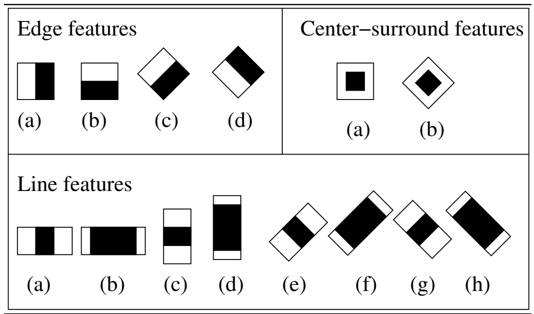


Figure 2.6. Examples of the Haar-like features [42]

learning based approach where a cascade function is trained with a lot of images. Created cascade classifier is then used to detect faces in other images.

Holistic approach

In this approach the face is detected as a whole unit by fitting a holistic model to the input image or tracking it in sequences of the images. A point distribution model (PDM) is a one of the template based method of representing the face. The model that is a combination of the PDM and the mouth template was proposed in [44]. This model is generated from 90 manually localized points and the mouth is approximated with parabolic curves. To detect face region, an estimation of its position is obtained by using Canny edge detector. After an initial placement of the PDM in the image, the proposed method moves and deforms the PDM [39]. Then, the difference between pixel intensity of the mouth and two symmetrical lines which represent the boundaries of the face is used to locate the face.

Another solution is based on dual-view facial images. The histogram of the frontal view image is used to determine boundaries of the head and the algorithm which exploits HSV color model allows to find the contour of the face [45].

Extraction of the facial expression information

Not only static but also dynamic features of the face play an essential role in facial analysis [46]. In order to measure the dynamics of the facial muscles the Active Appearance Models are used [47]. It is a computer vision algorithm which fits a statistical model of object shape to the new image.

Generally, an ideal system should be able to automatically extract all visually perceptible facial expressions. Well-defined face model is required as a prior condition for achieving this [39]. Well-defined means that it allows to uniquely reveal as many facial actions as possible (a reference is a total of 44 expressions defined in FACS [39], [48]). In analytic approach visual actions of the face can be extracted by describing the movements of specific points belonging to defined facial features and analyzing their interactions [49].

In order to extract facial expression information and find facial feature points locating some characteristic regions may be really helpful. Many systems start from finding the region or characteristic points of the eyes, nose or mouth. An example procedure for locating these features was proposed in [31]:

1. apply horizontal Integration Projection for region detection
2. normalize the histogram of Integration Projection, find the first minimum point and set it as the y coordinate of the mouth, the eyes or the nose
3. separate this region and apply the vertical Integration Projection to it
4. find two minimum points and set them as the x coordinate of specific points of the eyes, the nose or the mouth
5. separate this region
6. apply the Susan edge detector
7. apply the Harris corner detector

Taking into consideration one example of the template based method which couples together the frontal view model and the side view model, one should remember that this combination may result in redundant information about the facial features, as the frontal model include 30 features and the side model is composed of 10 points (presented in the figure 2.7 and 2.8). However the motivation for combining both models is the increase in the quality of the whole model [24].

Restrictions on face geometry recognition systems

As pointed in [44] the best results are achieved when the patient do not have facial hair and glasses. Moreover, systems work correctly when no

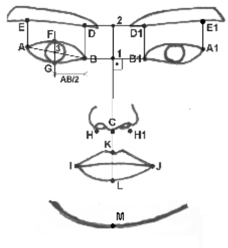


Figure 2.7. Frontal view model [24]

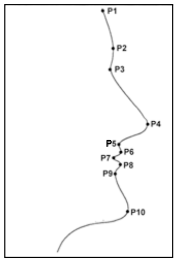


Figure 2.8. Side view model [24]

head motion is encountered and illumination variations are linear. In the method which uses PDM, the mouth template can't be created correctly if the teeth are visible [39]. Another problem is the fact that in some methods, for instance in Haar-feature approach the computation effort may be really high [42].

2.4.2. Thermography

The human body has the remarkable capacity for self regulating its core temperature. When the temperature rises, the body because of its heat-regulating mechanisms, dilates the blood vessels in skin to reduce the overall temperature. Without doubt, the thermoregulation process is connected with blood flow and should also be considered in the context of the face. Besides a wide range of applications of thermal imaging, this technique has been found valuable in diagnosing paralysis of the face during stroke. The results show that some significant differences between paralysed and helathy parts of the face have be discovered. These findings were also confirmed by thermographic measurements and the Doppler imaging method [50]. Fig. 2.9 presents thermography data of healthy people and Bell's palsy patients.

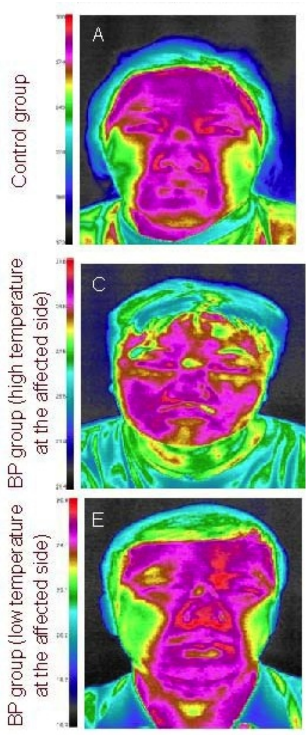


Figure 2.9. Thermography data of healthy people (A) and Bell's palsy patients (C, E) [51]

2.4.3. Micro movements

Previously there have been some attempts to discover and expose subtle motions in videos, however relying on motion compensation and similar algorithms may lead to computionally expensive techniques. Eulerian Video Magnification method is able to reveal low-amplitude motion and subtle changes that normally are invisible for humans without using explicit information of motion. Instead of it, EVM method amplifies temporal color changes at specific positions [52]. This method takes a standard video sequence as input. Then, it applies spatial decomposition and temporal filtering to the frames. Finally, the resulting signal is amplified to reveal hidden information [53]. EVM may be used for example to visualize the flow of blood as presented in the fig. 2.10. Using Eulerian Video Magnification for discovering or exposing subtle changes in facial acivity should be considered.

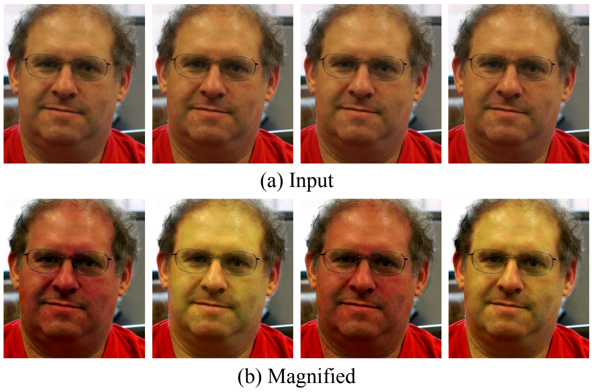


Figure 2.10. Eulerian Video Magnification framework for visualizing the human pulse [53].

2.5. Remote patient monitoring for detecting facial activity

2.5.1. Measuring face paralysis from video sequence

Although many algorithms are really promising, traditional medical methods should be taken into account during designing recognition and feature extraction systems.

In some solutions degree of facial paralysis is estimated by using Harris corner detection [31]. Several computer systems have been created to measure automatically single-sided paralysis of the face. Maximum static response assay (MSRA) method is based of measurements of the displacement of specific facial features. It allows to determine the amplitude of the movement by measuring position of the points during rest and during contraction [54]. These points may be used to obtain the differences between the two halves of the face [31].

2.5.2. Measuring face activity from video sequence

The Facial Action Coding System (FACS) is the most popular study on facial movements [48], which allow to find even small facial features changes caused by muscle contraction. FACS describes 44 so-called Action Units (AUs) which represent all possible visually detectable changes [24].

The department of the TU Delft has proposed a solution for an automatic non verbal communication tool [55]. The aim of this system is to extract information about specific verbal signal in two steps:

1. detect signal

2. categorise signal

Another solution is based on 3D dynamic face model which is a combination of facial tissues with facial muscles processes. Face muscles contraction parameters are estimated by using deformable curve techniques. This technique allows to animate the human face [56].

Neural networks are also used for recognition and extraction of facial expressions as presented in [57] and [58].

Features extraction and classification can be also performed by using an optical flow method with geometric, physical and motion models [59] or an optical flow model of image motion for extracting six basic emotional expressions (sadness, happiness, anger, disgust, fear and surprise) [60].

2.5.3. Video registration in web browsers

There is a wide range of medical applications which measure physical parameters using webcams. For example a non-contact method for evaluating cardiac activity using webcam was described in [61]. Considering the fact that more and more devices allow to use web browsers, video compression and frame rate conversion may turn out to be essential to correctly play recorded sequence of facial movements.

Video compression and frame rate conversion

Frame rate conversion is necessary because of the fact that different devices support different frame rates. Sometimes a video has to be converted to a different format because of the country, where it is emitted. Moreover, digital video transmission, which ousted the analog one and is very popular nowadays, need to be compressed in order to be used over typical channels. Compression is a process, which allows to reduce size of the original data and as a result reduce redundancy from the original stream. Furthermore, generating more frames can help to improve displaying images on liquid crystal display. LCD often shows ghost phenomenon or motion blur around moving objects, because of the their long response time. By generating more frames LCD frame rate can be improved. In order to get good video rendition with smooth moves it is necessary to find algorithms which will be able to increase temporal resolution of images sequence [62]. There are a lot of popular methods of frame rate conversion that are commonly used. Unfortunately they can introduce so called jerky motions or blurring artifacts [63].

1. Frame repetition (fr) - this method uses the current frame directly as an interpolated frame [64]. As can be seen in the fig. 2.11 generated frame (in the middle) is identical to the first one. This algorithm is really fast, not complex and doesn't increase the frame rate of the resulted video, so it can be considered as a lower bound of performance measurement [64]. However when a repetition value is not an integer, the frames are repeated a different number of times [63] and the jerky motions are discovered in a resulted video, especially near to moving objects.

Figure 2.11. Frame repetition [63].

2. Frame averaging (fa) – this method is based on blending frames together. It is achieved by interpolating pixel values, which are transmitted with a weight, that is based on their relative temporal position. The weighting parameter can be adjusted based on frame content [64]. This method can lead to reduction of flickering, but unfortunately moving objects may be blurred [63]. As can be seen in the fig. 2.12 generated frame (in the middle) suffer from blurring artifacts.

   There are two types of this method: straightforward and with temporal shifting. Frame averaging with temporal shifting were developed in order to reduce number of blurred frames. Moreover it is discovered that jerkiness introduced in movies by this method does not lead to discomfort while watching them.

Figure 2.12. Frame averaging [63].

3. Frame dropping (fd) – this method is helpful while performing evaluation. It can be done by dropping every other frame and reconstruct it. After that the reconstructed frame and the dropped frame are compared with each other.

4. Motion projection based techniques - can be performed in pixel or macroblock level (pixel level is more complex but has better performance). These techniques are based on motion information of two adjacent frames. As a result each macroblock of both of the frames is projected into the interpolated frame [64].

Unfortunately it is impossible to perform a completely flawless frame rate conversion. Hence, there is a wide range of algorithms which are useful during generating approximated intermediate frames. To avoid disadvantages (for ex. artifacts) of above methods some algorithms based on estimated movements between successive frames were developed [63]. These techniques are used for frame rate up conversion, which has generally higher quality than frame repetition or frame averaging methods [63] and are useful for reducing or even eliminating devices' flaws which are a result of physical issues.

1. Block motion estimation

   (a) Block distortion methods are based on measuring distortion between two blocks in order to get accurate motion estimation. The lower distortion is, the higher similarity is discovered [63].

      - Sum of absolute differences (SAD) counts a sum of all differences in intensity between two corresponding pixels in two blocks [63]. The problem of this method is the fact that two blocks with only one pixel much different from the pixel from the other block can result in lower distortion than overall distortion [63].

- Mean absolute difference (MAD) SAD divided by the number of all compared pixels.
- Squared differences (SSD) this method can solve the problem caused in SAD [63].

(b) Estimation Techniques In these techniques movement between two frames is described by special fields of motion vectors.

- Uni and Bi-directional Motion Estimation
- Bilateral Motion Estimation
- Overlapping Block Motion Estimation

2. Motion Vector Refinement - this method is used to improve move precision as it provides a good quality of movements in objects borders and only finite number of movements are considered [65].

3. Motion Compensation – this method is one of the most effective in coding source for digital video. It is employed in the encoding of video data for video compression. A main aim of this method is an elimination of frames' redundancy. It is achieved by motion estimation between neighboring frames. MC exploits the fact that most often the only difference between the frames is the motion of the camera and/or objects. As a result the only information needed to be stored is the information necessary to transform the previous frame into the next frame. It allows to reduce a lot of redundant information, as a video stream does not have to contain all of the full frames [66].

2.6. Conclusion

Although there are some speech recognition systems, a precise system which can analyze facial gesture from video sequence has not been developed yet [24]. Moreover each of the methods described above is not an ideal one, because of being semi automatic or even manual procedures [67], [68] and because of the lack of giving the interpretation of the certain expression [69]. Furthermore, it is worth mentioning that existing Active Appearance Models were not used for diagnosis of the facial paralysis . In the view of the foregoing there is a need to design an automatic tool for facial activity analysis. The state of the art presented in this section provides a complete source of knowledge required to implement an application of this kind. Moreover restrictions mentioned previously should be taken into account and tests of different measuring conditions should be performed. The state of the art presented in this chapter was prepared in reference to the hypothesis of the project.

Chapter 3

Methodology

3.1. Introduction

In second section of this chapter the work plan is presented according to significance of the project briefly explained in the first section. In next sections the methodology and designed algorithms are presented taking into consideration the possibility to convert the medical diagnose vision mechanisms into efficient computer algorithms.

3.2. Significance

Analyzing face features plays an essential role not only in communicating expressions but also in diagnosing many diseases. Face discrimination is also one of the most developed perceptual skills of visual processing. Therefore, the problems which are the consequences of the facial nerve paralysis and other pathological changes may lead to discomfort and have a negative impact on the personal development. During this period, any hope for improvement is priceless. There is a need to find a tool which can evaluate actual state of the health and determine whether the patients had made a progress during the rehabilitation cycle. This solution based on using innovative technologies can doubtless have a very positive impact on personal well-being and comfort. The motivation for this work is to provide algorithms for detecting geometrical and colours differences between two halves of the face during making sequence of special facial gestures. This combination may uncover new dependencies between data set and lead to conclusions that were previously undiscovered. Many new aspects analyzed and defined in this paper may turned out to be important in the future researches and significant for development in many areas, for example:

1. biomedical engineering for healthcare informatics and biometry
2. medicine for telemedicine and remote diagnostics
3. computer science for image analysis and recognition

3.3. Work plan

According to results that should be achieved, the plan of the project is divided into following parts:

1. Task 1 Implementation and validation of algorithms for detecting face
2. Task 2 Development and validation of algorithms for magnifying motions and colours of the video sequence
3. Task 3 Development and validation of algorithms for obtaining multi modal features of the face, including not only those related to the face structure but also function (changes in skin colours).
4. Task 4 Analyzing data collected at different time periods
5. Task 5 Description of structural and functional properties of a face

   Task 2 and 3 are the main scientific parts of the project.

3.4. Methodology

The scientific methodology of this project is based on critical review of the actual state of the art, run experiments and proposal of the own solution for detecting changes in face during rehabilitation cycle. The final method will result in possibility of confirming or rejecting the proposed hypothesis. It is also assumed that after implementing and testing the project the hypothesis may be modified. The experimental part of this work focuses on data acquisition and verification of proposed algorithms in the context of possibility to measure state of the face paralysis over time.

3.4.1. Detecting face

In the first step of the project face will be detected using Haar Feature-based Cascade Classifiers. This is an effective object detection algorithm proposed by Paul Viola and Michael Jones [70]. By using machine learning approach, algorithm is able to process images very rapidly. Initially, the algorithms needs a lot of positive and negative images. By positive images we treat images with a face while by negative images without a face. The first step is extracting features from images with the means of haar features 3.1, which are a single values obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle.

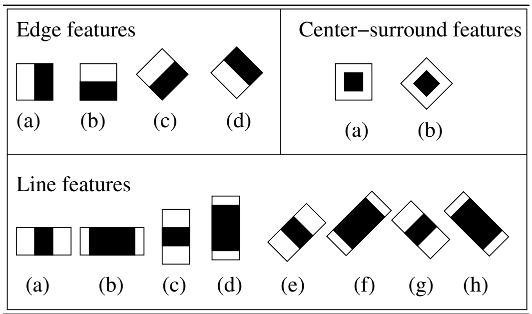


Figure 3.1. Examples of the Haar-like features [42]

The calculation is simplified by introducing integral images. However there are still plenty of features, so there is a need to select the correct ones from irrelevant ones. To do this, Adaboost algorithm is used. Firstly, each of the feature is applied on the images from training set. The classification of the face to positive or negative one is done by finding the best threshold. Then, features are divided into groups by error rate and the group with minimum error rate is chosen. The process of selecting features with minimum error rate is presented in the figure 3.2.

Separately the classifiers are weak, but together they form a good classifier. Final classifier is a weighted sum of these separate classifiers. After this step number of features is highly reduced. Yet, it still may require much computation effort and be a little inefficient. To solve this problem, it turned out that a successful idea is to skip non-face areas. The algorithm checks if a window is not a face region. If there is no face, this area is not processed again. However not all of the features are applied at once. They are divided into groups called Cascades of Classifiers and applied separately. If a window fails the first stage, the region is dismissed. If it passes, the next cascade is applied. The process is continued until all of the stages of features are applied. The window which passes all stages is a face region.

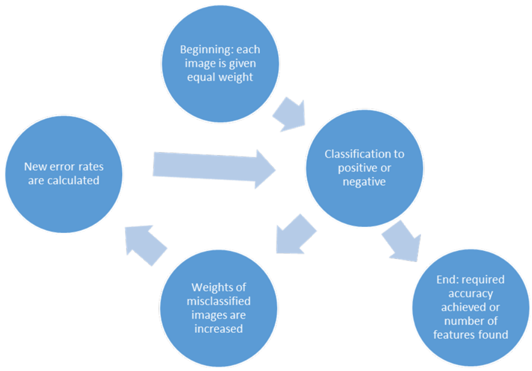


Figure 3.2. The process of selecting features - part of Paul Viola and Michael Jones algorithm

3.4.2. Algorithms for magnifying motions and colours

Recently a new technology out of Massachusetts Institute of Technology was discovered. This new algorithm focuses on revealing subtle variations which are normally invisible for the eyes. Such variations can be magnified computationally to extract new exciting features by taking advantage of Eulerian Video Magnification. This method was inspired by Eulerian perspective based on the fact that properties of voxel of fluid, like velocity, evolve over time [52]. The approach of this new technology is based on amplifying temporal colour changes at fixed positions and revealing subtle changes in video sequences by combining spatial and temporal processing as presented in the figure 3.3 Firstly, the video sequence in decomposed to spatial frequency bands and a signal from each of the band if processed temporally. Then, band-pass signal is magnified by a special factor and added back to original signal. Finally the output image is built from previously created pyramid of spatial frequency bands.

Based on the approach described above, the algorithm for magnifying motions and colours of the skin during making face gestures will be implemented. It will be divided into following parts:

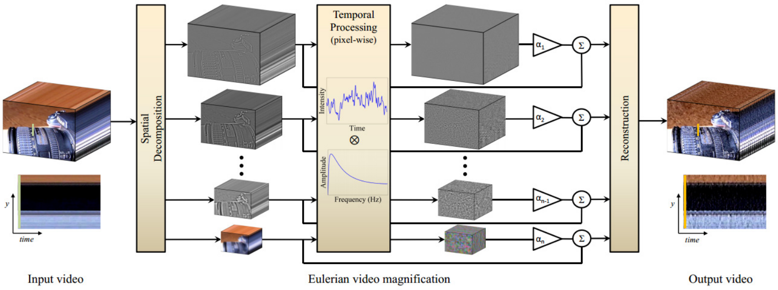


Figure 3.3. Eulerian Magnification Method overview

1. **Motion magnification**

   (a) **Apply spatial filter: Laplacian Pyramid** The Laplacian pyramid is frequently used in image processing and pattern recognition because of its ease of computation. By creating the Laplacian Pyramid we are able to to represent images as band-pass filtered images, each sampled at successively sparser densities. As a result we get a series of images which are weighted using a Laplacian transform. After repeating this technique multiple times, it creates a stack of successively smaller images. Each picture consist of pixels containing a local average that corresponds to pixels on a lower level. The process of the Laplacian Pyramid decomposition is presented in the figure 3.4.

   Example of Laplacian Pyramid with four levels is presented in the figure 3.5 and 3.6.

   Each band

$$[\overrightarrow{I_0}, \overrightarrow{I_1}, ..., \overrightarrow{I_N}]$$

   of the Laplacian pyramid is the difference between two adjacent low-pass images of the Gaussian pyramid as follows:

$$\overrightarrow{b_k} = \overrightarrow{I_k} - E\overrightarrow{I}_{k+1}$$

   where

$$E\overrightarrow{I_{k+1}}$$

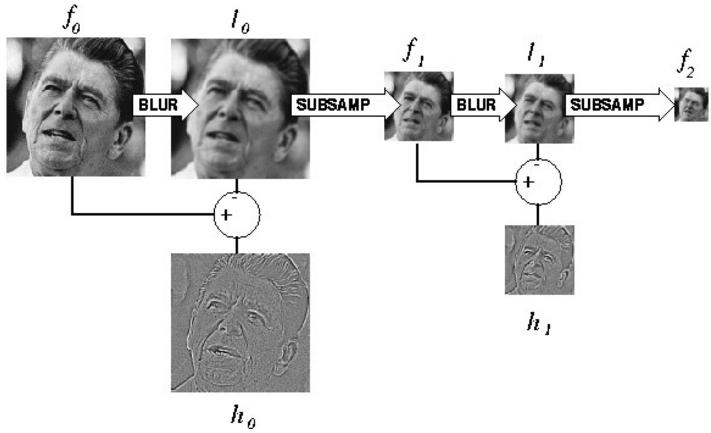


Figure 3.4. Two-level Laplacian Pyramid. The final pyramid consists of the two high-pass bands and one low-pass band [71]

Figure 3.5. Laplacian Pyramid with four levels, from the left level 0, 1, 2, 3

is a smoothed version of

$$\overrightarrow{I}_{k+1}$$

produced as follows:

$$E\overrightarrow{I}_{k+1} = \underbrace{\begin{bmatrix} . & & & \\ & -g & & \\ & & -g & \\ & & & -g \\ & & & & . \end{bmatrix}}_{convolution} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & ... \\ 0 & 1 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & ... \\ ... & ... & ... & ... & ... \end{bmatrix}}_{upsampling} \overrightarrow{I}_{k+1}$$

Figure 3.6. Laplacian Pyramid with four levels, first row from the left level 0, 1, second row level 2, 3

(b) **Apply temporal filter: IIR band-pass filter** IIR filters are digital filters with infinite impulse response. They are distinguished by having an impulse response which does not become zero past a certain point, but continues indefinitely. In spatial filtering a weighting factor is usually applied to express effect of the filter on each pixel of the frame. In this step of the algorithm an IIR band-pass filter will be applied to improve the quality of the image.

Noise is often introduced during the analog-to-digital conversion process as a side-effect of the physical conversion of patterns of light energy into electrical patterns [72].

Experimentally low cut off weight will be set to 0.05 and high cut off weight will be set to 0.85. However during experiments, these values should be compared and tested.

An ideal band-pass filter was applied in order to remove any amplification of undesired frequency from the color variation of each pixel.

(c) **Amplify filtered matrix** After applying temporal filter, the matrix will be amplified by the Amplification Booster for better visualization. By default the value will be set to 2.

(d) **Add magnified image to the original** Pictures will be combined together to preserve all of the features from original and magnified images.

(e) **Reconstruct image from Laplacian Pyramid** Reconstruction step for a two-level Laplacian Pyramid is presented in the figure 3.7. It is an inverse operation to previous sampling.

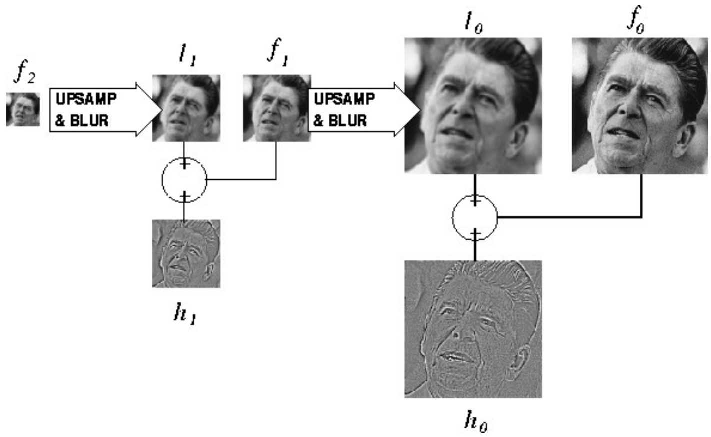


Figure 3.7. Reconstruction step for a two-level Laplacian Pyramid [71]

(f) **Attenuate** The reason of attenuating is to smooth irrelevant noise that may appear during magnification in previous procedures.

2. **Colours magnification**

(a) **Apply spatial filter: Gaussian Pyramid** A Gaussian pyramid is a technique which allows to create a series of images that were weighted using Gaussian blur. Similar as in Laplacian Pyramid case, after applying this technique many times, a stack of successively smaller images is created. The idea of Gaussian Pyramid can be explained using a set of layers in which the higher the layer the smaller the size, as presented in the figure 3.8.

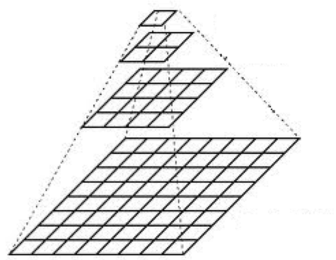


Figure 3.8. A set of layers in Gaussian Pyramid

Every layer is numbered from bottom to top, so layer i+1 is smaller than its predecessor i. To produce layer i+1, layer i is being convolved with the following Gaussian Kernel:

$$\frac{1}{16}\begin{bmatrix}1 & 4 & 6 & 4 & 1\\ 4 & 16 & 24 & 16 & 4\\ 6 & 24 & 36 & 24 & 6\\ 4 & 16 & 24 & 16 & 4\\ 1 & 4 & 6 & 4 & 1\end{bmatrix}$$

Then, every even-numbered is removed. The resulting image is one-quarter the area of the initial image.

(b) **Concat arrays into matrix** In this step every small frame from the pyramid is joined in one large mat (1 column for 1 frame) to ease the computation in the next steps.

(c) **Apply temporal filter: ideal band-pass filter** A signal can be expressed in many different domains. However let us remember that neither of the domain is more informative. It means that information contained in a signal is exactly the same regardless of the domain and different transformations do not change it. So why one domain may be preferred to the other? Because certain information may become more visible in one domain than in the other. To apply frequency domain filters on a digital image, the discrete equivalent of the continuous Fourier transform of the original image will be computed. This transformation converts a finite list of equally spaced samples into frequency-based data, which contains finite combination of complex sinusoids, ordered by their frequencies. After this operation filtering in frequency domain will be possible. The DFT of the original image

$$P(u, v)$$

will be multiplied by the impulse response function of the filter in frequency domain

$$H(u, v)$$

and as a result we will get the DFT of the filtered image

$$Q(u, v) = H(u, v).P(u, v)$$

In this step of the algorithm an ideal band-pass filter which is a linear combination of low-pass and high-pass filter will be applied to attenuate all frequencies outside the band-pass. Initially low cut off frequency will be set to 0.05 and high cut off frequency will be set to 0.85. However during experiments, these values would be compared and tested. The calculation of the inverse DFT of the

$$Q(u, v)$$

will result in filtered image back in the space domain.

(d) **Amplify filtered matrix** Procedure the same as for motion magnification.

(e) **Add magnified image to the original one** Procedure the same as for motion magnification.

(f) **Deconcat matrix into arrays** The whole matrix will be split back into small frames.

(g) **Reconstruct image from Gaussian Pyramid** Reconstructing image is done by up-sizing the image to twice its size in each dimension. As a result of this operation new even rows and columns are added and filled with zeros. After that, a convolution with the same kernel as presented in 2a multiplied by 4 is used to approximate the values.

3.4.3. Algorithms for obtaining multi modal features of the face

After preparing frames by extracting and amplifying colours of the area of the face, it will be easier to obtain multi modal features of the face. Two approaches to analysis of the face are proposed:

1. features related to the face structure

2. features related to the face function (changes in skin colours)

Face structure

In order to obtain features related to the face structure, the outer corners of the mouth and the centres of the eyes will be detected using Haar Cascade Classifiers in the same way as for face detection. Taking into consideration assessment of facial paralysis proposed by House Brackman and described in chapter 2.4, face will be analyzed during performing 3 specific movements (figure 3.9) (smile, e sound, sad) which allow to determine the actual state of the facial paralysis.

Figure 3.9. Assessment of facial paralysis based on specific movements, from the top: e sound, sad, smile

During maximum contraction distances between eye centres and mouth corners will be automatically calculated for both halves of the face as the Distance Formula which is a variant of the Pythagorean Theorem used in geometry. Four points will be given as presented in the figure 3.10 and they will be compared in pairs:

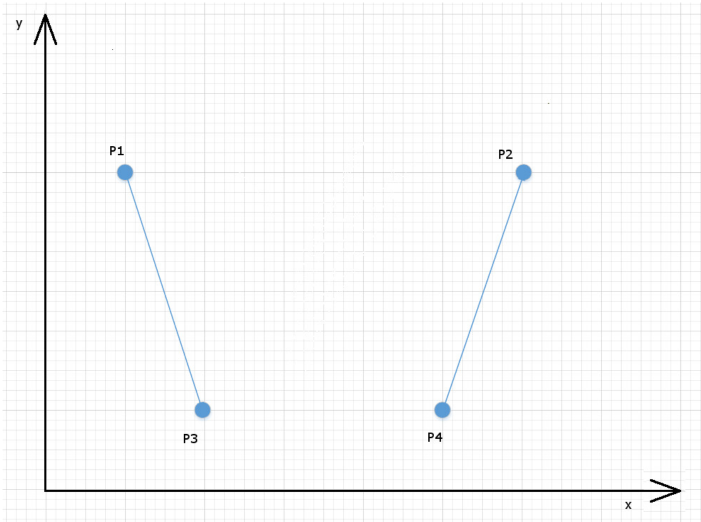


Figure 3.10. Geometrical arrangement of the points representing eye centers and mouth corners

- center of the left eye with the left outer corner of the mouth
- center of the right eye with the right outer corner of the mouth

Given two points, one can plot them, draw the right triangle and find the length of the hypotenuse, which is a distance between two points, defined as:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

After finding following distances relation between a left distance and a right distance will be computed and two approximation errors will be calculated for:

1. geometrical relation
2. colour difference

The approximation error in some data is the discrepancy between an exact value and the calculated value. It is defined as:

$$\epsilon = [V - V_{approx}]$$

where in the best case

$$V_{approx} = 1$$

The numerical stability of an algorithm indicates how the approximation error is propagated. This propagation will be analysed between time periods.

Face function

There is a possibility that lack of facial movements in paralyzed parts of the face will affect the local colour of the skin. In order to confirm that, most frequent colour for each half of the face will be found and compare with each other. The difference between these colours (CIE 1976) will be calculated as the distance metric:

$$\Delta E^*_{ab}$$

created by the International Commission on Illumination [73] with reference white D50. Collected results will be analyzed over time periods. Obviously, a lot of issues arises here. Firstly, it is doubtful whether colour of the face may be changed if the face is paralyzed. Secondly, illumination conditions may have a very big influence on obtained results and any change of testing environment may affect a diagnosis. Taking it into consideration, recognition performances in term of illumination conditions should be considered reliable.

3.4.4. Analysing data collected over time

Collected data will be analyzed over time to decide whether the patient is making progress or not. The actual approximation error will be compared with the previous result. Both points will be plotted and the straight line that passes through them will be drawn (figure 3.11).

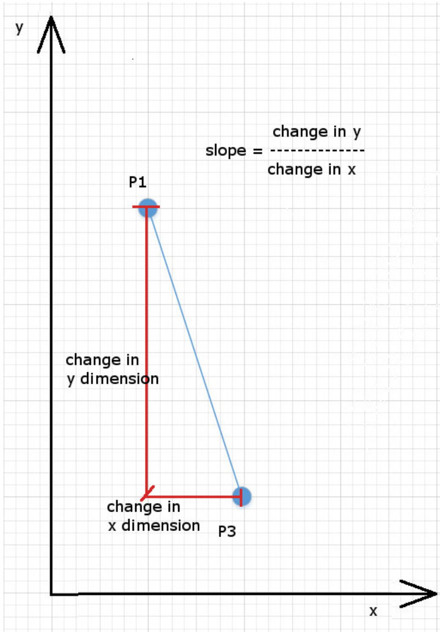


Figure 3.11. Straight line passed through points of the approximation error

The slope of this line will show how steep this line is and how big the change was. The direction of the horizontal change will confirm progress (negative direction) or its lack (positive direction or slope equals zero).

3.4.5. Description of structural and functional properties of a face

Gathered information will be analyzed in order to find dependencies between achieved results. Research will focus on testing the proposed algorithm which combines analysis of structural and functional properties of the face. This innovative method may allow to introduce some new possibilities for patients monitoring.

3.5. Measurement restrictions

Obviously, the measurement methods for obtaining physiological parameters from face are valid under limited set of conditions (type of light, position of the light, type of camera, etc.), Taking it into consideration influence of different factors, mainly related to the environment, should be evaluated. The correctness of the implemented algorithms will be tested under different conditions in order to specify requirements which allow to obtain reliable results.

3.6. Conclusion

This chapter includes a complete description of the methodology used to prepare analysis of the requirements for the created application. The proposed algorithm takes advantage of the possibilities of new technologies to provide medical help at a distance. Magnifying colors and counting both geometrical and colours dependencies is the main scientific part of the solution. The proposed algorithms should be verified in order to confirm their correctness.

Chapter 4

System design

4.1. Introduction

In this chapter the aim of the project and potential users of the application were defined. Secondly the functional, non functional, hardware, software and quality requirements were specified. Then, the project of the system was proposed and described, based on created UML and classes diagrams. Moreover the prototype of the interface was designed and finally used architecture and technologies were defined.

4.2. Aim of the project

Main aim of the project is to create a tool which will allow to control state of the facial paralysis and monitor efficiency of the therapy. This goal will be achieved by releasing online a program which will automatically compare multi modal features of the face over time. The application will be created with the intent of encouraging patients to regular test and self control of the rehabilitation capabilities. It should also indicate how important are regular exercises and monitor their effectiveness as in many cases a correct therapy increases the chance of recovery and the possibility of shortening the duration of rehabilitation process. The project may become a useful tool used for monitoring any changes. Yet, it will not replace clinical solutions and will not be a substitute of professional diagnostic methods.

4.3. Potential users

The application will be dedicated mainly to patients who have already been diagnosed with face paralysis. For this group of users it will be useful in monitoring size of the potential progress or decline. On the other hand users who do not suffer from this kind of disease will also be able to use this tool to detect changes in face function and if necessary consult these results with specialists as the faster facial paralysis is diagnosed, the more possible recovery becomes. The application will be used by the whole society, no matter what gender or age the patients are.

4.4. System requirements

4.4.1. General requirements

1. **Business context** The program will be available on web browsers, so one will be able to use it on every device that has an access to the Internet. The application sharing with a remote access will make the rehabilitation procedure more attractive in the context of:

   (a) preventive therapy for patients who do not have facial paralysis

   (b) monitoring therapy for patients who have already been diagnosed on facial paralysis

   Furthermore monitoring state of the health in both cases will be more comfortable and easier. The examination will take place without having to leave home so patients will willingly check their facial function and features what may result in faster response to anomalies. Moreover by using telemedicine access to health care will be available for people living in rural areas, health care costs will be reduced and medical institutions will get faster access to information about their patients.

2. **Prospectiveness**

   Created application can be used as a module of a larger system and expanded with additional tests of other health parameters. This fact indicates the prospectiveness of a proposed solution which may become a part of the whole platform for remote patient monitoring.

4.4.2. Functional requirements

In order to access the content of the application, a user will have to register an account. During registration he will have to accept terms of use pointing that information included in application are indicative and neither gives professional medical diagnose nor substitutes a professional therapy. While using this application a user should be still under the care of a specialist because the application is created for general informational purposes and cannot be a substitute for specialist medical advice. A user who doesn't accept these conditions, won't be able to use an application.

The main page of the project will contain a login form. After filling it with a correct combination of a user name and a password, an examination page will be opened and an alert box asking user for the permission to use a camera will appear. After allowing the page to access the camera, the examination will be started. During the entire examination the user will see two separate canvases on which two frames sequences will be drawn:

1. video before processing
2. video after processing

There will be three buttons **smile**, **e sound**, **sad** below canvases, each responsible for a chosen gesture. A user will have to make this gesture and push the appropriate button to mark a current frame as a gesture frame. After that a frame will be send to a server and all of the features described in 3.4.3 will be calculated. Finally, results will be send to a user. Moreover, results and frames will be also saved to a file in order to allow specialists to analyse them later.

Processed frames will contain a box with following information:

1. marked face
2. marked eyes areas and centered
3. marked mouth area an mouth corners
4. magnified color of a face area
5. ratio of the left distance to the right distance (distance from eye center to mouth corner)

6. most frequent color of the left and the right cheek in the RGB color model

The user will be able to decide about the order of the examination.

Examinations and exercises included in the application will not be time-limited. The application will work on-line and is expected to use by one user at a time. The application will not have different levels of difficulty. The idea is to compare results obtained at different moments of time and achieved during making the same exercises because it will allow to control changes and make an objective comparison of the state of the health during rehabilitation progress. Moreover, at any time a user will be able to make a break in examination and continue it after a while.

On the main page there will also be a link to a user guide explaining how the application should be used. Other functionality includes access to a compendium of knowledge that explains what symptoms may appear during facial paralysis and which diagnosis and rehabilitation options are available and possible.

Diagram presented in the figure 4.1 shows general functional requirements available in the application. The aim of the diagram is to describe possible interactions between actors and functions offered by the application. Each oval shows a single use case within which different flows called scenarios may occur. Every use case characterizes possible system functionalities and defines steps that lead to the effect observable by the actor. A dependency marked as **extend** describes some additional behavior, which in some cases may be included into the base use case. The base use case is the one which is pointed by the arrowhead of the connection. The extend use case continues the behavior of the base use case by inserting additional actions into the base sequence when the appropriate extension point is reached in the base use case and the extension condition is filled. An **include** dependency is a relationship which describes the inclusion of another use case responsible for different behavior. The including dependency is the one pointed by the arrowhead of the connection.

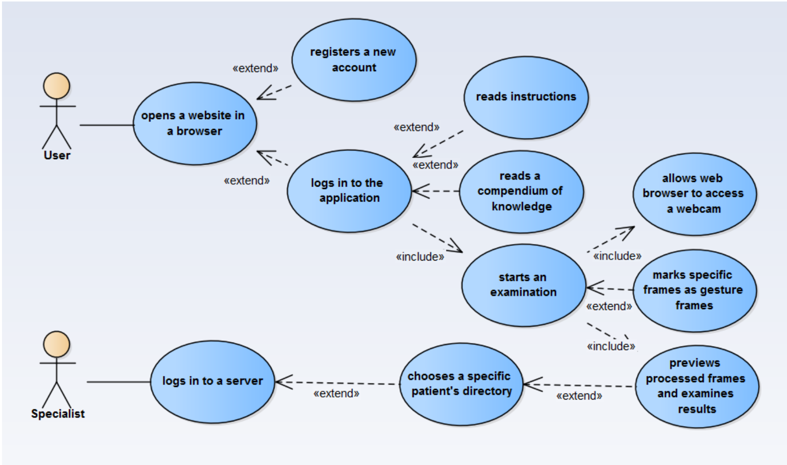


Figure 4.1. Use case model of the application - all functionalities

To present more detailed interactions, use cases from following categories were extended and shown on separate diagrams:

1. an examination 4.2
2. recording and previewing the results 4.3

In a first version of the application a logging and registration mechanisms will be only a prototype. The project will be focused on the examination part.

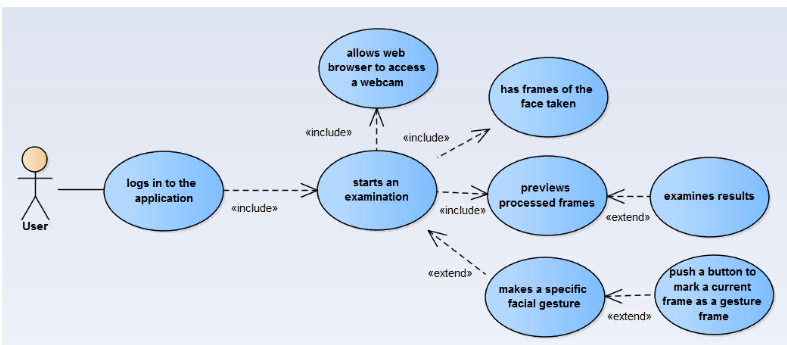


Figure 4.2. Use case model of the application - examination process

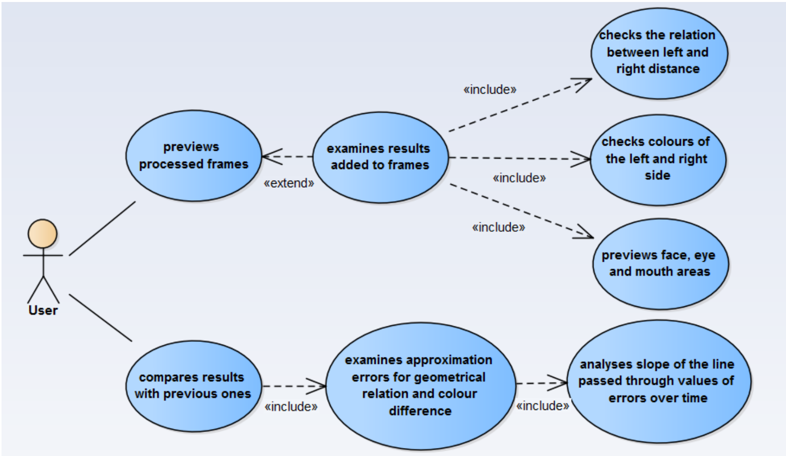


Figure 4.3. Use case model of the application - results preview process

Descriptions of all of the use cases including their initial, alternative and results were collected in the table 4.1.

Use case	**Initial conditions**	**Alternative conditions**	**Results**	**Description**
opens a website in a browser	has a device with a supported web browser and access to the Internet	any	any	a user opens a web browser and a website of the application.
reads instructions	a website of the application opened, access to the Internet	any	any	a user opens a website of the application, chooses an instruction page and read it

reads a compendium of knowledge	a website of the application opened, access to the Internet	any	any	a user opens a website of the application, chooses a compendium of knowledge page and read it
registers a new account	a website of the application opened, access to the Internet, valid values in the registration form, terms of use accepted	any	any	user opens a website of the application, chooses a registration page and fills a form with correct values
log in to the application	a website of the application opened, active user account, access to the Internet, valid combination of a username and a password	any	access to the examination page	a user opens a website of the application, chooses a log-in page and fills a form with correct values to get access to the examination page
logs in to a server	correct ssh keys which allows to log in to a server	any	a user is logged in	a user logs in to a server using ssh client and has access to results

starts an examination	a website of the application opened, access to the Internet, user logged in, user accepted turning on the webcam	any	the face appeared on the canvas	a user opens a website of the application, put a valid combination of a username and a password into login form, gave a web browser access to a webcam, which starts to capture a face of the user, draws it on the canvas and sends it to a server in order to process frames.
allows a web browser to access a webcam	a user has logged in to the application	any	an examination started	this action is required in order to start the examination, so this use case is included by the previous one, the user logs in to the application and in order to start an examination, he has to allow a browser to use a webcam

marks specific frames as gesture frames	a user started the examination, access to the Internet	any	marked frames are drawn in specified places on the website	a user started the examination, makes a specific facial gesture and pushes and appropriate button, after that current frames are marked as gesture frames and they appear on the website
chooses a specific patient's directory	a user is logged to a server	any	any	a user logs in to a server using ssh client and opens chosen directory
previews processed frames and examines results	a user is logged in to a server or a user starter the examination	any	any	a user is logged in to a server or a user started the examination and examines results
has frames of the face taken	a user started the examination	any	frames are sent to a server	a user started the examination and gave a browser access to a webcam, which takes pictures of his face every 250 milliseconds

makes a specific facial gesture	a user started the examination	any	any	a user started the examination, makes a specific facial gesture
push a button to mark a current frame as a gesture frame	a user started the examination and made a specific gesture	any	marked frames appear on specific places on the website	a user is taking the examination, he can make chosen gestures in any order and then push an appropriate button to mark current frames as gesture frames
checks the relation between left and right distance	a user started the examination, both eyes and mouth corners were detected	any	any	a user is logged in and a browser has access to a webcam, the user started the examination, points needed to calculate distances were detected and all of the values were computed, the user can examine them

compares results with previous ones	the examination was taken, all necessary values were calculated	any	any	both during the examination and after finishing it, a user can examine results as they are saved on the server and compare them
examines approximation errors for geometrical relations and colour difference	the examination was taken, results and their approximation errors were calculated	any	any	approximation errors of geometrical relations and colour differences are calculated for every examination
checks colours of the left and right side	a user started the examination and a face was detected	any	any	a user is logged in and a browser has access to a webcam, the user started the examination, the face was detected, colours were calculated and displayed on frames and the user can examine them

previews face, eyes and mouth area	a user started the examination, all areas were detected	any	any	a user is logged in and a browser has access to a webcam, the user started the examination, area of the face, eyes and mouth were detected and displayed to the user
analyses a slope of the line passes through values of errors over time	approximation errors were calculated	any	decision about the size of a progress	approximation errors of geometrical relations and colour differences are calculated for every examination, so it is possible to plot them and pass a line through points calculated over time, finally a slope of this line can be computed to decide how big the progress is

Table 4.1. Use cases description

4.4.3. Non functional requirements

Quality requirements

Communication with a user is especially important in medical applications as any mistake may be very expensive and have negative impact on the patient. In applications of this kind there is no place for any error. The interface of the application is presented schematically in the section 4.5.5. During the examination the user should sit still in front of the camera. Probably, as shown by the analysis of the state of the art, the best results are achieved if a user has no glasses and his face is not covered by any additional elements, however these conditions will be tested and specified later. All of the alert boxes, important for the user, will appear in the center of the screen. Yet they will not interrupt or interfere with the examination. The application will support English language, however to perform tests, understanding it won't be required. All necessary actions will happen automatically and the user will be able to manipulate it by following instructions from pictures.

Ergonomy

The interface will be created in a clear and readable way in order to focus users attention on the performed action instead of distracting them with additional elements. All of the screens will be designed in a friendly manner to make an examination more pleasant and users less nervous of suffering from a disease. Taking into account potential users, described in section 4.3, the application should be useful and functional for patients from all of the age groups. The most important feature of the application for being suitable for all of the users is versatility. Furthermore, the application will be also used by elderly people, so the font and other graphical elements should have an appropriate size, colour and contrast.

Restrictive requirements

A user of the application, will have to have basic skills of manipulating a computer mouse or touching screens. All of the steps and actions needed to take the examination will be easy, intuitive and as automatic as possible. For example the examination will automatically start after logging to the system. In order to use the application the following software and hardware requirements will have to be filled.

4.4.4. Software and hardware requirements

In order to use the application a user will need one of the following:

1. a computer equipped with a web camera and a mouse

2. any mobile device equipped with a web camera

Taking software requirements into consideration the only one requirement points that a web browser will have to be installed on a chosen device and it has to support WebRTC. One can choose from following options:

1. Desktop PC
   (a) Google Chrome 23
   (b) Mozilla Firefox 22
   (c) Opera 18
2. Android
   (a) Google Chrome 28
   (b) Mozilla Firefox 24
   (c) Opera Mobile 12
3. Chrome OS
4. Firefox OS
5. iOS
   (a) Bowser

There are no specified requirements about the operating system installed on the device until it supports required web browser. Moreover in order to use the application a user will have to have access to the Internet.

4.5. Project of a system

This section provides the project of the described solution for supporting remote patient monitoring during facial paralysis. The project was presented by the diagrams employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems [74]. The section presents main functionalities of the application in accordance to 3 categories
:

1. an examination
2. recording and preview of the results

3. a guide

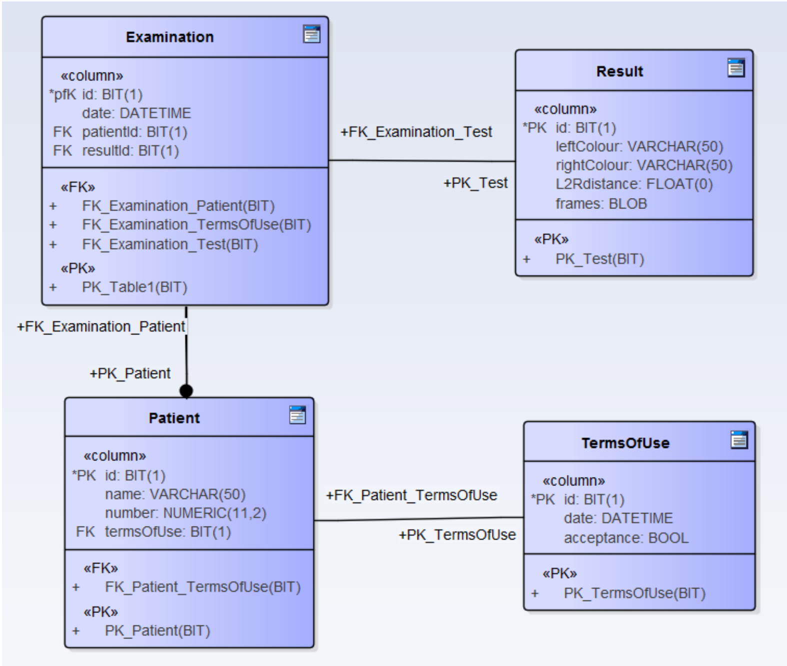


Figure 4.4. Entity-relationship model

4.5.1. Entity–relationship model

An entity–relationship diagram (figure 4.4) was used to present which kind of data will be saved during the examination. The whole examination consists of one test in which geometrical and functional features of the face are saved, including:

- date of the examination
- information about the patient (number, name)
- relation between left and right distance
- colours of the left and right parts of the face
- frames

There is also an entity **terms of use** on the diagram, as accepting it, is necessary to perform tests.

Kinds of tests and facial gestures were chosen based on analysis of existing diagnosis methods and symptoms of the disease. In order to better understand how the application works, the main flows were presented on the UML diagrams.

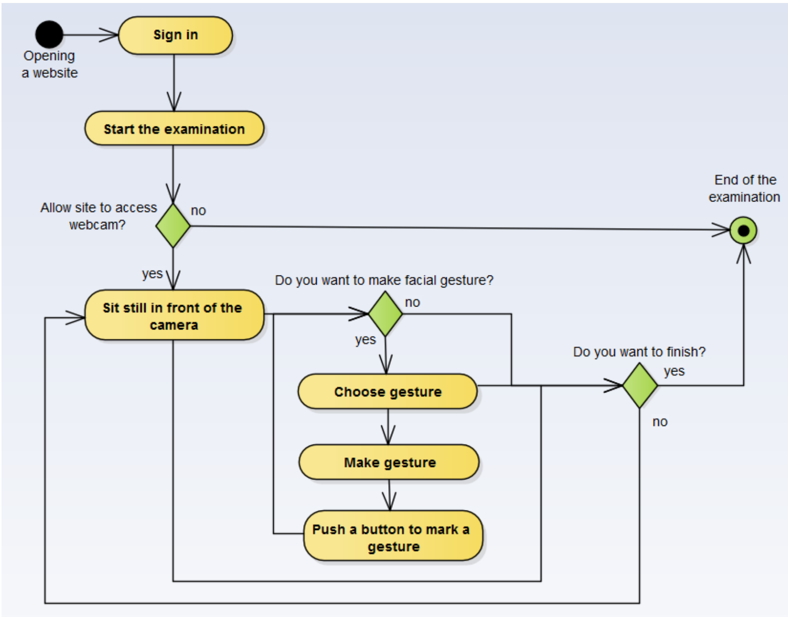


Figure 4.5. Activity diagram of the performing test flow

4.5.2. Activity diagram of the performing test flow

The activity diagram shows the logic of workflows of step-wise actions. Rectangles with rounded corners present sequences of activities. A rhombus represents a decision after which it is possible to choose one of the alternatives. A diagram in the figure 4.5 shows a sequence of actions needed to take an examination. Opening a website is a starting point, closing it is a final point. While performing test, it is possible to repeat it at any time. A user can choose the order of the examination and start with any of the listed facial gestures, he can even skip these steps. During the examination, the user can analyze results. After finishing the test, he can compare actual results with previous ones. All of the results are automatically saved on the server.

4.5.3. Sequence diagram of processing frames

A sequence diagram (figure 4.6) was used to present steps needed to get processed frames. Actions that will be implemented will allow to take picture of the face, send it to a server and get a response with processed frames.

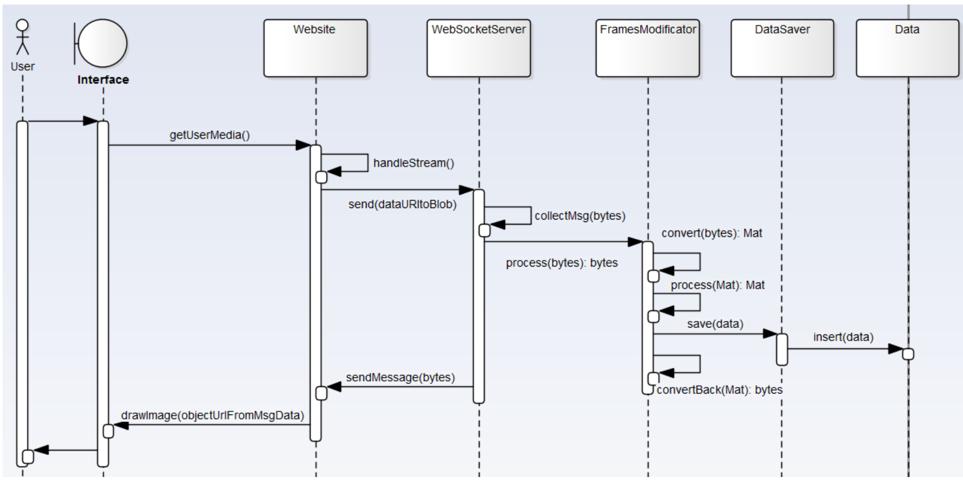


Figure 4.6. Sequence of processing frames

4.5.4. Diagram of classes

A diagram of classes (figure 4.7)is one of the most popular UML diagrams. Rectangles show classes, their attributes and methods. Association connections were used to present relations between entities. Classes communicates with each other by sending messages (by association connections) and as a result they are able to fill tasks presented on the dynamic diagrams, for example use case model 4.1.

In order to present the diagram more clearly it was divided into smaller parts presented in figures 4.8, 4.9, 4.10. The WebsocketServer will be the main class of the application. It will be responsible for creating a new Server listening on the specific host and port, which will create a FaceDetectWebSocket. The FaceDetectWebSocket class will open and close connections and will send and receive messages. After getting bytes of frames, the class will invoke method from the FramesConverter class responsible for processing frames. Then, processed frames will be send back to the website and saved

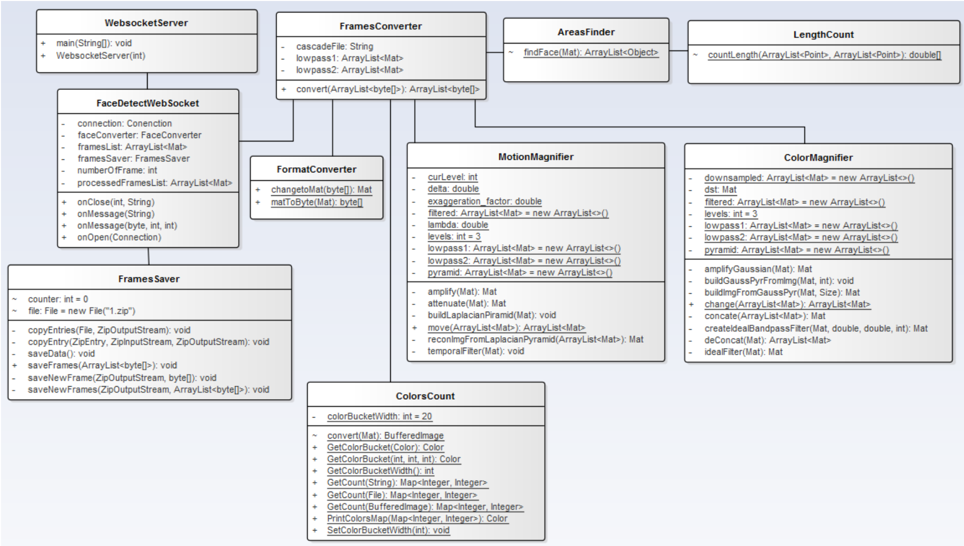


Figure 4.7. Diagram of classes

to zip files in FramesSaver class. At this step it is not necessary to create a movie as all of the features can be examined by analyzing frames. Taking it into consideration any algorithms for video compression and frame rate conversion won't be implemented.

The FramesConverter class will be responsible for all of the operations required to implement described algorithms. This class will invoke methods from other classes, including:

- FormatConverter - convert to Mat from byte (and back)
- AreasFinder - find area of face, mouth and eyes
- MotionMagnifier - magnify motions
- ColorMagnifier - magnify colors
- LengthCount - count length of the left and the right distance from eye center to mouth corner.
- ColorCount - count colors of two halves of the face

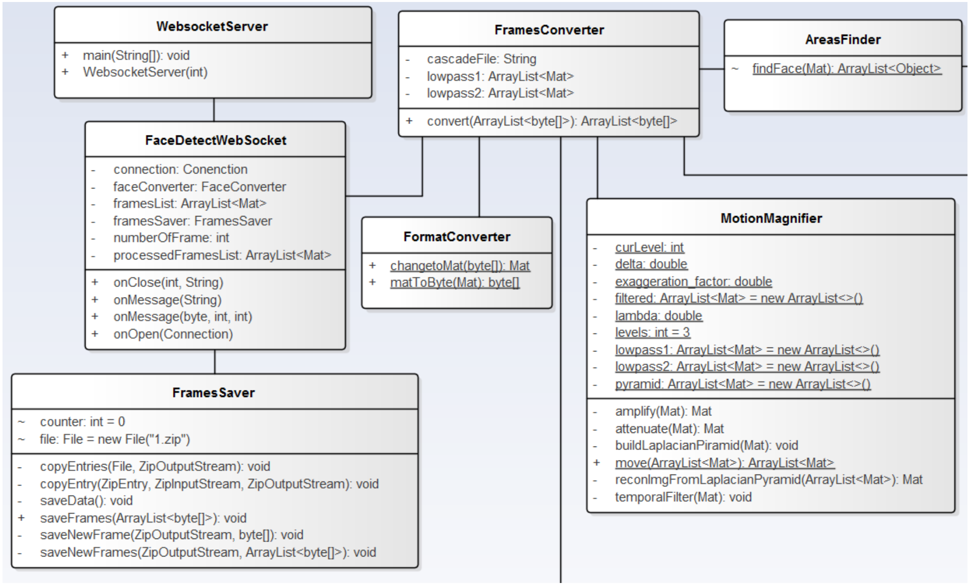


Figure 4.8. Diagram of classes - part 1

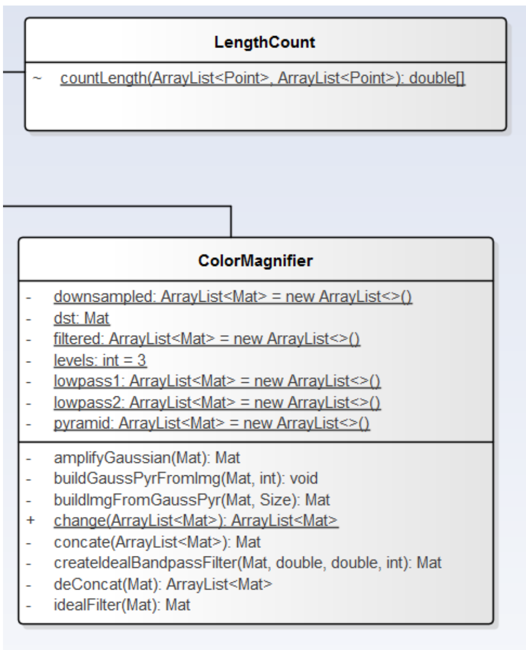


Figure 4.9. Diagram of classes - part 2

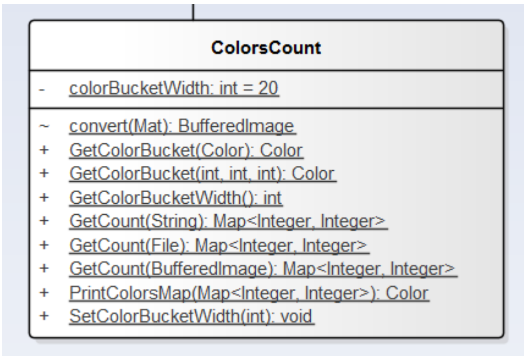


Figure 4.10. Diagram of classes - part 3

The diagram was also attached as an appendix at the end of the document.

4.5.5. Interfaces

Considering the requirements defined in section 4.4.3, the project of the interface was prepared. The application will contain two primary pages: a **sign in** page presented in the figure 4.11 and an **examination** page presented in the figure 4.12.

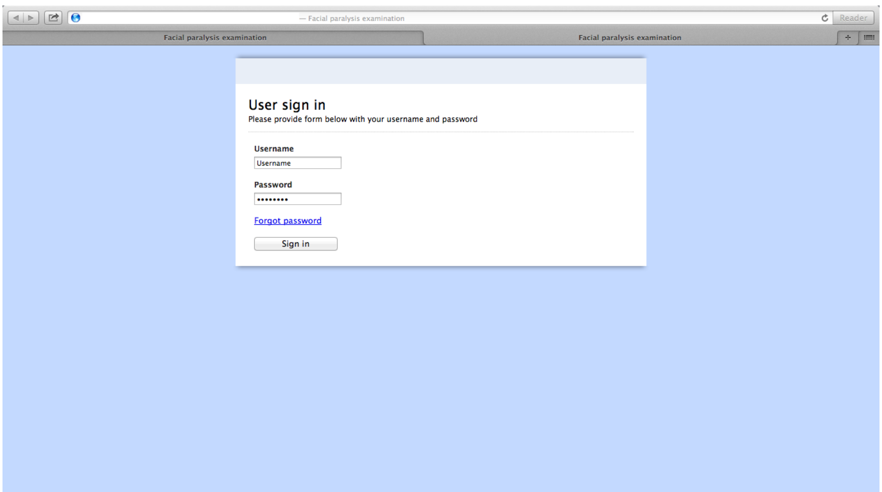


Figure 4.11. Project of sign-in page interface

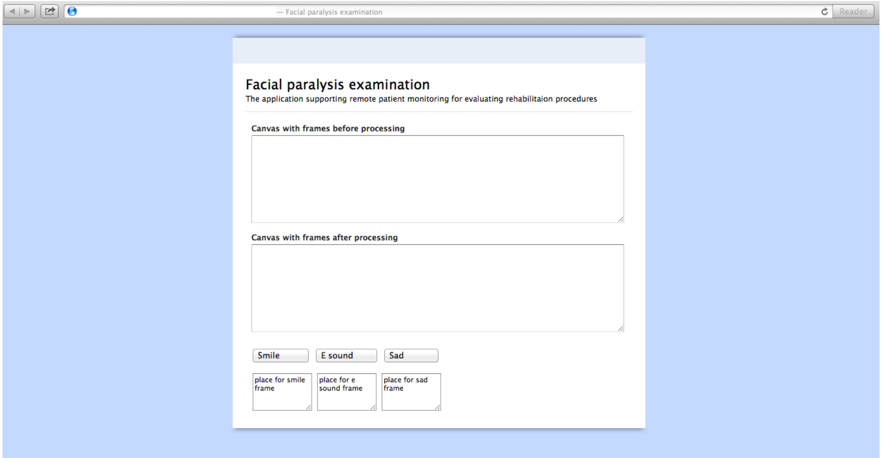


Figure 4.12. Project of the main page interface

4.5.6. Architecture

The whole application will be based on client-server model. It is a distributed application structure that partitions tasks between the providers, called servers, and service requesters, called clients [?]. The main idea of this model is presented in the figure 4.13.

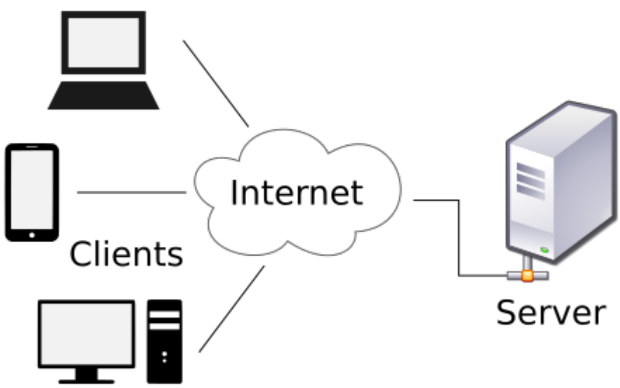


Figure 4.13. Main idea of the client server model [75]

As can be seen in most cases clients and servers communicate over a computer network. The client initiates communication session with the server and requests a service function from a server, which runs a program that shares resources with the client. In described solution a web server serves a web page with an examination interface. The client does not have to be concerned with how the server works and delivers the response, but he has to understand the response as they exchange messages in a request–response messaging pattern. The client sends a request, in this case a binary message with a frame content, the server processes it and sends a response (frames after processing). This exchange of messages is an example of inter-process communication.

4.5.7. Technologies

For a client side following technologies will be used:

1. HTML5 - a markup language of the Internet used to present content for the World Wide Web. This is the fifth revision of the HTML. Its main goal is to improve the language by supporting the latest multimedia while keeping it easy to understand by humans, computers and devices. In HTML5 there is a wide range of new tags, for example: $< video >$, $< audio >$, $< canvas >$. Moreover it provides integration of scalable vector graphics (SVG) content (replacing generic $< object >$ tags), and MathML for mathematical operations.

2. WebRTC - an API proposed by the World Wide Web Consortium (W3C). It supports browser-to-browser applications which are based on voice and video chat without the need to use internal or external plugins. Main components of this standard include:

    - getUserMedia - camera and audio access and possibility to capture media
    - RTCPeerConnection - responsible for setting audio/video calls
    - RTCDataChannel - possibility to share data between peers
    - getStats - possibility to retrieve a set of statistics

    WebRTC allows to communicate in real time and share video, audio and text by utilizing the browsers. Its main advantages include ease of use without any additional applications or plug-ins, providing security mechanisms such as encryption for the media and the signaling and possibility to save costs.

For a server side following technologies will be used:

1. Java - concurrent, class-based, object-oriented and designed to have as few implementation dependencies as possible computer programming language
2. Jetty - a Java HTTP (Web) server and Java Servlet container
3. Maven - build automation tool used mainly for Java projects. Maven addresses two aspects of building software: First, how software is built, and second, which dependencies are needed.
4. OpenCv - an open source computer vision and machine learning software library, which was built to provide a common infrastructure for computer vision applications [76].

4.5.8. Application release

The application will be uploaded and hosted on the Amazon.com AWS (Amazon Web Services) platform, which offers reliable, scalable, and inexpensive cloud computing services [77]. Specifically, Amazon Elastic Compute Cloud (EC2) was used to rent virtual machines to run their own computer applications. EC2 allows scalable deployment of applications by providing a web service. A user can boot an instance containing any software. Releasing the application allows to provide health services to users at a distance.

4.6. Conclusion

This chapter includes a complete description of the methodology used to prepare analysis of the requirements for the created application. The proposed algorithm takes advantage of the possibilities of new technologies to provide medical help at a distance. Magnifying colors and counting both geometrical and colours dependencies is the main scientific part of the solution. The proposed algorithms should be verified in order to confirm their correctness. Moreover the project of the system was also described and presented taking into consideration all of the specified requirements.

Chapter 5

Implementation

5.1. Introduction

This chapter describes implementation of all of the algorithms and functionalities presented in the methodology and design chapter 3. The proposed idea of remote patient monitoring is a fully completed solution.

5.2. Description of the solution

As previously described the application takes advantage of the client-server architecture. The description of the solution was divided into two parts taking it into consideration.

5.3. Client side

The client side is a website implemented in HTML5 markup language. To access the application a user should simply visit a website *http://ec2-52-24-99-202.us-west-2.compute.amazonaws.com:8080/*. The WebRTC API was used for communicating with the server in real time and sending video frames.

In order to capture the video the method **getUserMedia** was used:

```
navigator.getUserMedia = navigator.getUserMedia || navigator.
    webkitGetUserMedia || navigator.mozGetUserMedia || navigator.
    msGetUserMedia || navigator.oGetUserMedia;

if (navigator.getUserMedia) {
```

```
        navigator.getUserMedia({video: true}, handleVideo, videoError);
        }
```

This method is one of the main components responsible for access to a camera and audio. If the action ends with a success, the method **handleVideo** is invoked. This method was responsible for opening connection to the server and exchanging messages with it.

```
function handleVideo(stream) {
                video.src = window.URL.createObjectURL(stream);
                var ws = new WebSocket("ws://127.0.0.1:9999");
                ws.onopen = function() {
                    ...
                };
                ws.onmessage = function(msg) {
                  ...
                };
                timer = setInterval(
                        function() {
                           ...
                        }, 250);
            };
```

The whole communication was defined by WebSocket JavaScript interface, which is a part of HTML5 initiative and enables a full-duplex single socket connection over which information can be exchanged with a remote host. What is interesting about this interface is the fact that it allows to traverse firewalls and proxies. Moreover, the compatibility of the WebSocket protocol with the existing Web infrastructure was ensured.

Using WebSocet interface is very easy. To connect to another point, a new WebSocket instance should be created by providing an URL that represents this point and a port on which it is listening.

```
var ws = new WebSocket("ws://ec2-52-24-99-202.us-west-2.compute.amazonaws
     .com:9999");
```

During the initial handshake between two points the HTTP protocol is upgraded to the WebSocket protocol in order to establish a connection. Each phase of the connection life-cycle can be handled by event listeners. For example opening connection by:

```
ws.onopen = function() {
        console.log("Openened connection to websocket");
                };
```

The operation of exchanging messages is achieved by using **onmessage** and **send** functions from the WebSocket interface.

After getting access to media, frames were drawn on one of the canvases. Then, data was converted to DataURL, after that to a Blob Object and a message was send to a server. All of these operations were invoked every 250 milliseconds using a timer:

```
timer = setInterval(
           function() {
                            ctx.drawImage(video, 0, 0, 320, 240);
                            var data = canvas.toDataURL('image/jpeg',
                                 1.0);
                            newblob = dataURItoBlob(data);
                            ws.send(newblob);
                        }, 250);
```

Conversion to a Blob Object was necessary, because when a WebSocket is created, it is set up to handle this kind of data. The Blob Object represents an object similar to file that consists of immutable, raw data. After processing frames they were sent back to the website and displayed as following:

```
ws.onmessage = function(msg) {
                    msgGlob = msg;
                    loadUrl("target", msg, 320, 240);
                };
                (...)
function loadUrl(targetId, msg, w, h) {
                var target = document.querySelector("#" + targetId);
                var myURL = window.URL || window.webkitURL;
                var url = myURL.createObjectURL(msg.data);
                target.onload = function() {
                    myURL.revokeObjectURL(url);
                };
                target.src = url;
                ctx2.drawImage(url, 0, 0, w, h);
            };
```

The design was created with Cascading Style Sheets - a style sheet language used for describing the design of a document implemented in a markup language.

5.4. Server side

The server side was responsible for receiving messages from the client, processing frames, saving data and sending back processed frames.

5.4.1. Handle communication

The server **WebsocketServer** is a class inheriting from a class Server from the jetty package [78] and it is created by providing a port on which it is listening. After creating it, the server is started and joined. Call of **join** (join() of Jetty's thread pool) guarantees that after it, the server is ready as join blocks other actions until the server has completed its startup. It is similar to Thread.join().

```
WebsocketServer server = new WebsocketServer(9999);
server.start();
server.join();
```

In order to specify host on which the server is created the **SelectChannelConnector** class was used.

```
connector.setHost("ec2-52-24-99-202.us-west-2.compute.amazonaws.com");
connector.setPort(port);
```

As documentation [79] explains this connector uses efficient non-blocking IO buffers with a non blocking threading model. Threads are only allocated to connections with requests.

The communication was defined by using Jetty WebSocket Server API which provides the ability to wire up WebSocket endpoints by upgrading the HTTP protocol to WebSocket protocol and migrating from a HTTP Connection to a WebSocket Connection.

To handle messages the FaceDetectWebSocket class overrides the implementation of methods **onOpen, onClose, onMessage** in the jetty superclass WebSocket.

After getting the message received data were added to the list and passed to **FaceConverter** class responsible for extracting all features from video sequence and implementing magnifying algorithms. Processed frames were passed to **FramesSaver** class responsible for writing them to archive files. These frames were also returned back to the website by invoking **sendmessage** method.

```
public void onMessage(byte[] data, int offset, int length) {
            ByteArrayOutputStream bOut = new ByteArrayOutputStream();
            bOut.write(data, offset, length);
            try {
                framesToProcess.add(bOut.toByteArray());
                if (framesToProcess.size() == processingSize) {
                    framesAfterProcessing = faceconverter.convert(
                         framesToProcess);
```

```
                    framesSaver.saveFrames(framesAfterProcessing);
                    if (framesAfterProcessing.size() > 1) {
                        for (int i = 0; i < framesAfterProcessing.size(); i++) {
                            this.connection.sendMessage(rl.get(i), 0, rl.get(i).length);
                            Thread.sleep(delay);
                        }
                    } else {
                        this.connection.sendMessage("failed");
                    }
                    al.clear();
                }
            } catch (IOException | InterruptedException e) {
                LOG.log(Level.SEVERE, null, e);
            }
        }
```

5.4.2. Saving frames

All frames collected during the examination are saved to the archive files. At first a temporary file is created. Then, frames which already exist in a file for current examination (file name as patient number and date) are copied to this temporary file. After that new frames are also saved to this temporary file and finally the temporary file is moved to the file for current examination with the option **REPLACE EXISTING** which means that the move is performed even when the target file already exists.

The described method looked as follows:

```

public void saveFrames(ArrayList<byte[]> processedFrames) {
        try {
            final File tempFile = File.createTempFile("old", "new");

            try (ZipOutputStream zos = new ZipOutputStream(new FileOutputStream(tempFile))) {
                copyEntries(file, zos);
                saveNewFrames(zos, rl);
                zos.close();
            }

            Files.move(tempFile.toPath(), file.toPath(), StandardCopyOption.REPLACE_EXISTING);
        } catch (IOException ex) {
            LOG.log(Level.SEVERE, null, ex);
        }
    }
```

The method **copyEntries** is responsible for copying already existing frames. It iterates over all the items using the getNextEntry method on the stream (ZipInputStream) in a given archive file. What is important this operation returns only the header data for each entry in turn, not data, which is actually read and copied from the stream separately in the method **copyEntry**. Saving frames is implemented analogically by putting and writing entries to a ZipOutputStream.

```
private void copyEntries(final File file, ZipOutputStream zos) throws
    IOException {
        try (final ZipInputStream zis = new ZipInputStream(new
            FileInputStream(file))) {
            ZipEntry zipEntry = zis.getNextEntry();
            while (zipEntry != null) {
                copyEntry(zipEntry, zis, zos);
                zipEntry = zis.getNextEntry();
            }
        } catch (FileNotFoundException e) {
            LOG.log(Level.SEVERE, null, e);
        }
    }

private void copyEntry(ZipEntry zipEntry, ZipInputStream zis,
    ZipOutputStream zos) throws IOException {
        byte[] buffer = new byte[100];
        int count;

        try (ByteArrayOutputStream baos = new ByteArrayOutputStream()) {
            while ((count = zis.read(buffer, 0, buffer.length)) != -1) {
                baos.write(buffer, 0, count);
            }

            zos.putNextEntry(zipEntry);
            zos.write(baos.toByteArray());
            zos.closeEntry();
        }
    }
```

5.4.3. Converting frames

The **FramesConverter**class manages all the conversion operations. Most of them were implemented using OpenCV library, so a first step was to convert received message from **byte[]** to a format readable by this library.

5.4.4. Format conversion

OpenCV mainly uses **Mat** structure which is a basic image container. Thanks to this format, we no longer need to manually allocate its memory and release it as soon as we do not need it. The Mat consists of the matrix header and a pointer to the matrix containing the pixel values.

```
private Mat changetoMat(byte[] img) {
        Mat image = new Mat();
        try {
            InputStream inputStream = new ByteArrayInputStream(img);
            BufferedImage imBuff = ImageIO.read(inputStream);
            image = new Mat(imBuff.getHeight(), imBuff.getWidth(), CvType.CV_8UC3);
            byte[] pixels = ((DataBufferByte) imBuff.getRaster().getDataBuffer()).getData();
            image.put(0, 0, pixels);
        } catch (IOException ex) {
            Logger.getLogger(FaceDetection.class.getName()).log(Level.SEVERE, null, ex);
        }
        return image;
    }
```

On the other hand, after processing, frames are sent back to the website, but the only format supported by WebSocket is String or byte[], so they had to be converted back from Mat to byte[].

```
private byte[] matToByte(Mat image) {
        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        try {
            MatOfByte bytemat = new MatOfByte();
            Highgui.imencode(".png", image, bytemat);
            byte[] bytes = bytemat.toArray();
            InputStream in = new ByteArrayInputStream(bytes);
            BufferedImage imgb = ImageIO.read(in);
            in.close();
            ImageIO.write(imgb, "png", bout);
        } catch (IOException ex) {
            Logger.getLogger(FaceDetection.class.getName()).log(Level.SEVERE, null, ex);
        }
        return bout.toByteArray();
    }
```

5.4.5. Areas finder

The next step focuses on finding a face area in order to perform magnifying algorithms and extracting important functional facial features. However, as previously described the main aim of the project is to find changes in state of the facial paralysis by obtaining features not only related to face function but also to structure. To achieve this goal other areas have to be also calculated:

- eyes areas
- mouth area

Based on calculated areas eye centres and mouth corners will be computed. The class responsible for this operation is called **AreasFinder**.

Areas are found using Haar Feature-based Cascade Classifiers, described in the section 3.4.1. In OpenCV a classifier is created by providing a path to files contains trained classifiers for detecting objects of a particular type. In the project three classifiers were used. the process of loading them runs as follows:

```
File fileFace = new File("C:\\OpenCV\\data\\haarcascades\\haarcascade_frontalface_alt.xml");
File fileEyes = new File("C:\\OpenCV\\data\\haarcascades\\haarcascade_eye.xml");
File fileMouth = new File("C:\\OpenCV\\data\\haarcascades\\haarcascade_mcs_mouth.xml");

CascadeClassifier faceDetector = new CascadeClassifier(fileFace.getPath());
CascadeClassifier eyeDetector = new CascadeClassifier(fileEyes.getPath());
CascadeClassifier mouthDetector = new CascadeClassifier(fileMouth.getPath());
```

To detect objects of different sizes in the input image, the **detectMultiScale** method can be used. The detected objects are returned as a list of rectangles.

```
faceDetector.detectMultiScale(image, faceDetections);
```

The following method will search for face areas in input matrix **image** using appropriate Cascade Classifier. As a result it will return face areas as a list of rectangles. The rest of areas were detected in a similar manner, however the input matrix was limited by making a submat from it.

```
Mat cropForeyes = face.submat(4, (2 * face.width()) / 3, 0, face.height());
```

After getting a list of rectangles and iterating over it, found areas were drawn on the input image.

```java
for (Rect rect : faceDetections.toArray()) {
                Core.rectangle(image, new Point(rect.x, rect.y), new
                    Point(rect.x + rect.width, rect.y + rect.height),
                    new Scalar(0, 255, 0));
                    ...
                    }
```

5.4.6. Magnifying motions

Motion and colours magnification was implemented based on the Eulerian Video Magnification algorithm proposed by Massachusetts Institute of Technology [80]. At first after finding the area of the face it was processed in order to magnify micro movements. To achieve this, the steps described in the chapter 3 were implemented. Most important ones are described below.

1. **Apply spatial filter: Laplacian Pyramid** The Laplacian Pyramid was created following the algorithm described in the section 1. At this stage of the project, the pyramid with 3 levels was used.

```java
private void buildLaplacianPyramid(Mat img) {
        pyramid.clear();
        Mat currentImg = img;
        for (int l = 0; l < levels; l++) {
            Mat down = new Mat();
            Mat up = new Mat();
            pyrDown(currentImg, down);
            pyrUp(down, up, currentImg.size());
            Mat lap = new Mat();
            Core.subtract(currentImg, up, lap);
            pyramid.add(lap);
            currentImg = down;
        }
        pyramid.add(currentImg);
    }
```

To perform upsampling the function **pyrUp** with 3 arguments was used:

- down - the current image
- up - the destination image
- currentImg.size() - the destination size. In the upsampling operation pyrUp expects a size double than the input image

To perform downsampling the function **pyrDown** with 2 arguments was used:

- currentImg - the current image
- down - the destination image

The destination size was left as default, since it is the operation of up-sampling pyrUp expects a size double than the input image.

2. **Apply temporal filter: IIR bandpass filter** The band-pass filter was used to improve the quality of the image as described in the section 1. The output sample of a 2-D IIR filter can be computed using the input and previously calculated output samples as follows:

```

    private void temporalFilter(Mat src) {
    //experimental values
        Scalar highPassLeft = new Scalar(0.8);
        Scalar lowPassLeft = new Scalar(0.2);
        Scalar lowPassRight = new Scalar(0.05);
        Scalar highPassRight = new Scalar(0.95);
        Core.multiply(src, highPassLeft, t1);
        Core.multiply(src, lowPassRight, t2);
        Core.multiply(lowpassLeft.get(curLevel), lowPassLeft, t11);
        Core.multiply(lowpassRight.get(curLevel), highPassRight, t22);
        Core.add(t1, t11, temp1);
        Core.add(t2, t22, temp2);
        lowpassLeft.set(curLevel, temp1);
        lowpassRight.set(curLevel, temp2);
        Core.subtract(lowpassLeft.get(curLevel), lowpassRight.get(curLevel), dst);
        filtered.add(dst);

    }
```

3. **Amplify filtered matrix** After that the filtered matrix was multiplied by the exaggeration factor to expose changes while ignoring the highest and lowest frequency band.

```
private Mat amplify(Mat src) {
   ...
        currAlpha *= exaggeration_factor;
        if (curLevel == levels || curLevel == 0)
        {
            Core.multiply(src, new Scalar(0), d);
        } else {
            Core.multiply(src, new Scalar(Math.min(2, currAlpha)), dst);
        }
        ...

    }
```

4. **Add magnified image to the original** This operation was performed by using **add** method from the OpenCV Core class.

```
Core.add(magnified, notMagnified, dst);
```

5. **Reconstruct image from Laplacian Pyramid** Reconstructing an image was done in the operation inverse to the previous sampling operation.

```
private Mat reconImgFromLaplacianPyramid(ArrayList<Mat> filtered) {
        Mat currentImg = filtered.get(levels - 1);
        Mat d = new Mat();
        for (int l = levels - 1; l >= 0; l--) {
            Mat up = new Mat();
            pyrUp(currentImg, up);
            ...
        }
        ...
    }
```

5.4.7. Magnifying colours

After finding the area of the face it was also processed in order to magnify local colour changes. To achieve this, the steps described in the chapter 3 were implemented. Most important ones are described below.

1. **Apply spatial filter: Gaussian Pyramid** The Gaussian Pyramid was created following the algorithm described in the section 1. At this stage of the project, the pyramid with 3 levels was used.

```
private void buildGaussPyrFromImg(Mat image, int levels) {
        pyramid.clear();
        Mat currentImg = image;
        for (int l = 0; l < levels; l++) {
            Mat down = new Mat();
            pyrDown(currentImg, down);
            pyramid.add(down);
            currentImg = down;
        }
    }
```

To perform downsampling the function **pyrDown** with 2 arguments was used:

- currentImg - the current image
- down - the destination image

2. **Concat arrays into matrix** In a first step of joining every single frame into a large matrix, a temporary matrix was created. A size of this matrix was the same as a size of the biggest image from a pyramid. CV$_3 2FC3$ means that this is a three channel matrix of 32-bit floats. Then, each of the frame from the pyramid was reshaped and cloned to next lines of the temporary matrix.

```
private Mat concat(ArrayList<Mat> frames) {
    Size frameSize = frames.get(0).size();
    Mat temp = new Mat((int) (frameSize.width * frameSize.height), levels - 1, CV_32FC3);
    for (int i = 0; i < levels - 1; ++i) {
        Mat input = frames.get(i);
        Mat reshaped = input.reshape(3, input.cols() * input.rows()).clone();
        Mat line = temp.col(i);
                    //copy reshaped frame to each column
        reshaped.copyTo(line);
    }
    temp.copyTo(dst);
    return dst;
}
```

3. **Apply temporal filter: ideal band-pass filter** Filter was applied on each channel individually to attenuate all frequencies outside the band-pass defined between low-pass equals 0.05 and high-pass equals 0.8. The frame rate was set as 25. At first the **split** method from Core class was used to split input image into channels. Then, the height of the input image was expanded to optimal size defined by **getOptimalDFTSize**. The performance of a DFT depends on the image size. Generally the best performance is achieved if the size of the image is a multiple of the numbers two, three or five. Therefore, to make the operation fast border values can be pad to the image to get a size with such traits. Zero values were added on the borders by using the function **copyMakeBorder** which copies the source image into the middle of the destination image and the rest of the image area is filled with extrapolated pixels.

   The next step included calculating discrete Fourier Transform of the image. The result of a Fourier Transform is complex, so for each image value we get two image values.

   After that the operation of multiplying the per-elements of two Fourier spectrums was performed. The spectrum of the image was multiplied with the spectrum of the created filter, defined in a next method.

Finally channels were merged back into a single image and the whole image was normalized.

```
private Mat idealFilter(Mat src) {
        Mat dst = new Mat();
        ArrayList<Mat> channels = new ArrayList<>();
        split(src.clone(), channels);
        for (int curChannel = 0; curChannel < channels.size(); ++curChannel) {
            Mat current = channels.get(curChannel);
            Mat tempImg = new Mat();
            int width = current.cols();
            int height = getOptimalDFTSize(current.rows());
            Imgproc.copyMakeBorder(current, tempImg, 0, height - current.rows(), 0, width - current.cols(), Imgproc.BORDER_CONSTANT, Scalar.all(0));
            Core.dft(tempImg, tempImg, Core.DFT_ROWS | Core.DFT_SCALE, Core.DFT_ROWS | Core.DFT_SCALE);
            Mat filter = tempImg.clone();
            filter = createIdealBandpassFilter(filter.clone(), lowpass, highpass, frameRate);
            Core.mulSpectrums(tempImg, filter, tempImg, Core.DFT_ROWS);
            Core.idft(tempImg, tempImg, Core.DFT_ROWS | Core.DFT_SCALE, Core.DFT_ROWS | Core.DFT_SCALE);
            Rect r = new Rect(0, 0, current.cols(), current.rows());
            Mat roi = new Mat(tempImg, r);
            roi.copyTo(channels.get(curChannel));
        }
        merge(channels, dst);
        normalize(dst, dst, 0, 1, NORM_MINMAX);
        return dst;
    }
```

As can bee seen values of the filter matrix for frequencies below and above low cut off and high cut off frequencies were set to 0. After multiplying its spectrum with the image spectrum all frequencies outside the band-pass were attenuated.

```
private Mat createIdealBandpassFilter(Mat filter, double cutoffLo, double cutoffHi, int framerate) {
        float width = filter.cols();
        float height = filter.rows();
                // cut frequencies
        double fl = 2 * cutoffLo * width / framerate;
        double fh = 2 * cutoffHi * width / framerate;
        double response;
                // filtermask with shape of a quarter of a circle
        for (int y = 0; y < height; ++y) {
            for (int x = 0; x < width; ++x) {
                if (x >= fl && x <= fh) {
```

```
                response = 1.0;
            } else {
                response = 0.0;
            }
            filter.put(y, x, response);
        }
    }
    return filter;
}
```

4. **Amplify filtered matrix** The amplifying operation was performed by multiplying the matrix by a specially designed scalar.

```
private Mat amplifyGaussian(Mat im) {
        Scalar s = new Scalar(2);
        Mat m = im.clone();
        Core.multiply(m, s, m);
        return m;
    }
```

5. **Deconcat matrix into arrays** The last step of magnifying colours of the facial area was to split the big matrix back into single matrices. This method is an inverse operation to **concat** method described above.

```
private ArrayList<Mat> deConcat(Mat src) {
        Size frameSize = downsampled.get(0).size();
        ArrayList<Mat> a = new ArrayList<>();
        for (int i = 0; i < levels - 1; ++i) { // get a line if any
            Mat line = src.col(i).clone();
            Mat reshaped = line.reshape(3, (int) frameSize.height).
                 clone();
            a.add(reshaped);
        }
        return a;
    }
```

6. **Reconstruct image from Gaussian Pyramid** Finally the destination image was created by reconstructing it from the pyramid.

```
private Mat buildImgFromGaussPyr(Mat pyr, Size size) {
        Mat r = new Mat();
        Mat currentLevel = pyr.clone();
        for (int i = 0; i < levels; ++i) {
            Mat up = new Mat();
            pyrUp(currentLevel, up);
            currentLevel = up;
        }
        currentLevel.copyTo(r);
        return r;
    }
```

The image with magnified colours and the image with magnified motion were combined in order to form the final image.

5.4.8. Most frequent colour counting

To compare two halves of the face (the paralyzed and the healthy one) the most frequent colour of each of them was calculated by invoking the method **getCount** from the class **ColoursCount**. As the output of this method, the object which maps keys (colour value in the RGB model) to values (number of its occurrences) was returned. Colours were taken from the input image in the RGB model.

```
public static Map<Integer, Integer> GetCount(BufferedImage sourceImage){
        Map<Integer, Integer> s= new HashMap<>();
        int value;
        for(int x=0; x<sourceImage.getWidth(); x++){
            for(int y=0; y<sourceImage.getHeight(); y++){
                  = new (sourceImage.getRGB(x,y));
                 Bucket = sCount.GetBucket();
                if(s.containsKey(Bucket.getRGB())){
                    value = s.get(Bucket.getRGB())+1;
                }
                else{
                    value = 1;
                }
                s.put(Bucket.getRGB(), value);
            }
        }
        return s;
    }
```

5.4.9. Distances counting

To obtain geometrical features of the face the method **countLength** from the class **LengthCount** was invoked with two lists as parameters. One list contained points of eye centered, the other points of mouth corners. The distance between point **a** and **b**was computed as the hypotenuse:

```
Math.sqrt((a.x - b.x) * (a.x - b.x) + (a.y - b.y) * (a.y - b.y));
```

5.5. Final design of the website

Based on defined non functional requirements and taking into account potential users an interface of the application was created to be universal, intuitive and clear as presented in the figure 5.1. The button is highlighted with blue after a mouse event.

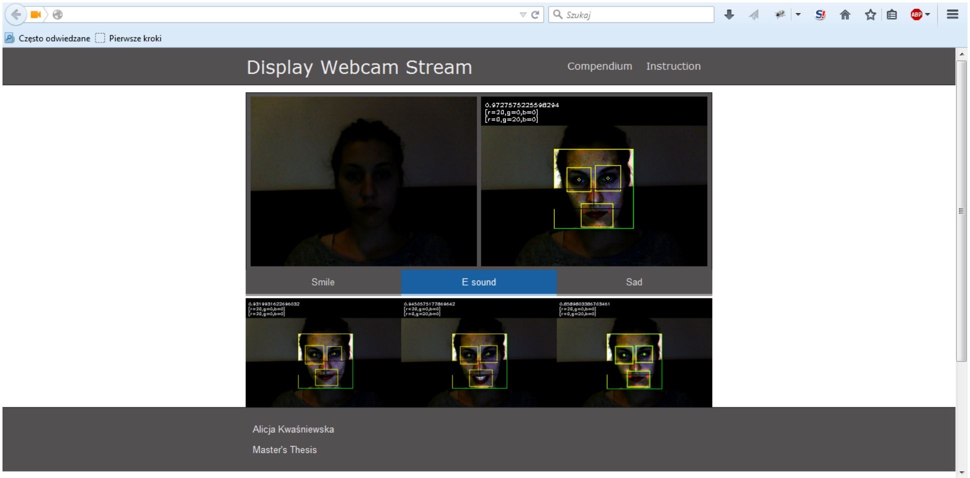


Figure 5.1. Final interface of the application

5.6. Conclusion

The implementation of all functionalities important for analyzing facial paralysis defined in the chapter 3 was done based on the state of the art described in the chapter 2. Some functionalities that did not have influence on the hypothesis were designed theoretically. The designed classes perform actions presented on the UML diagrams 4.5. In this chapter the way of how the application works was described including used and implemented methods, subclasses, superclasses, interfaces and packages.

Chapter 6

Tests, experiments and results

6.1. Introduction

In this chapter performed tests and examinations are briefly described. The application was tested in different measuring conditions on different devices. The chapter includes a description of test scenarios and is focused on defining best measurements conditions. The results of the tests and experiments are presented in next sections of the chapter.

6.2. Tests

Performing tests is an integral process of a software production cycle. The main goal of testing process is to verificate and validate the software in order to ensure its quality. Verification checks whether the software is implemented according to specification and validation whether the software is compatible with the expectations of the user. Software testing can be implemented at any time during software development life cycle.

6.3. Used devices

For all of the tests and experiments a computer with parameters collected in the table 6.1 was used:

Processor	Memory	System	Camera
Intel(R) Core(TM) i3-2350M CPU @ 2.3 GHz	*8 GB*	*Windows 7 64 bit*	*scb-1100n 1.3 MP*

Table 6.1. Parameters of the device used for tests

The server was launched on the Amazon EC2 t2.micro instance with parameters collected in the table 6.2.

vCPU	Memory	Storage (GB)	Physical Processor	Networking performance	Clock Speed (GHz)	Intel AVX†	Intel Turbo
1	*1 GB*	*EBS only*	*Intel Xeon family*	*Low to moderate*	*Up to 3.3*	*Yes*	*Yes*

Table 6.2. Parameters of the Amazon instance

Processor	Memory	System	Camera
Intel Core i5 1,6 GHz	*8 GB 1600 MHz DDR3*	*OS X Yosemite*	*720p FaceTime HD camera*

Table 6.3. Parameters of the second device used for tests

Processor	Memory	System	Camera
Intel Core i5 1,6 GHz	*1GB RAM Snapdragon 400*	*Android OS, v4.4.2 (KitKat)*	*5 MP*

Table 6.4. Parameters of the third device used for tests

6.4. Used methods

6.4.1. A white-box testing

A white-box testing was used as a first testing method. In this kind of tests a tester (in this case the author of the application) had an access to the software source code, observed how different parts of the application behaved and which modules and libraries were used. After confirming that the application is error free by examining any fragile code, next tests were performed.

6.4.2. A black-box testing

A functional testing was used as a second testing method. These tests often called black-box testing rely on the fact that a software is treated as a black box that performs certain tasks. A user does not penetrate it and does not analyze any technical details of the program.

To validate the application many different methods of obtaining information may be used. Each of them is created for other purposes. To choose the right method it is necessary to consider what kind of target group we would like to test, how much time could be spend on researching, what kind of results are expected and whether we need a lot of details about these results or not [81]. The following methods can be distinguished:

1. Observation - is an intentional and systematical process of examining users in their natural environments.
2. Interview - is a kind of a conversation. It can be divided into one-on-one interviews and group interviews.
3. Facilitated session - is a kind of meeting with a larger group gathered for using your product.
4. Survey - is a set of questions prepared for the group of respondents, who answer these questions without the researcher.

In order to examine the quality of the application and possibility to confirm the hypothesis, it was decided to observe patients during performing tests.

In order to check the correctness of the way of how the application works and confirm a positive validation of the created project, user acceptance tests

(UAT) were performed on a testing group which consists of 3 users. UAT includes a process of verifying that an application works for users, so during taking the examination they were asked for their point of view. Based on their observations a design of the interface was improved. The elements were placed in a more optimal way and the contrast between them was increased. These tests allowed also to examine the ease of the application use. Every user managed to perform the examination without any problem.

Based on the theoretical analysis of the problem, functionalities described in the chapter 4 have been implemented. After undertaking User Acceptance Tests the application has been tested in practice by performing following experiments.

6.5. Experiments

6.5.1. An impact of the used device on the results

Firstly, it was checked how a percentage of properly detected areas and a correctness of counted distances change for different devices. Following devices were used:

1. **D1** - Samsung Notebook NP300V5AI with parameters collected in the table 6.1

2. **D2** - Macbook Air with parameters collected in the table 6.3

3. **D3** - Smartphone LG L90 with parameters collected in the table 6.4

On all of the devices the Mozilla Firefox browser in version 39.0 were used. For each of the device following tests were performed:

1. **percentage of properly detected areas** - in this experiment the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentages of all detected areas (area of the face, the eyes and the mouth) were calculated.

2. **correctness of counted distances** - in this experiment the face was being scanned for two minutes and all frames together with calculated distances were saved to the archive file. After that the distances

between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

6.5.2. A correctness of the application algorithms

Then, a correctness of the application algorithms was checked. At first under strictly defined conditions (daylight, scanning distance equals 30 centimeters, hair tied, no eyeglasses, face in front of the camera, not rotated, used device: D1) following parameters were examined:

1. **percentage of properly detected areas** - in this experiment the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentage of all detected areas (area of the face, the eyes and the mouth) were calculated.

2. **correctness of counted distances** - in this experiment the face was being scanned for two minutes and all frames together with calculated distances were saved to the archive file. After that the distances between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

3. **visibility of colour changes after their magnification** - in this experiment the face was being scanned for two minutes and all frames were saved to archive file. After that some of them were combined into one sequence to highlight the difference between colors. Then, the color in a RGB model was taken for a chosen part of the picture and three graphs for each channel were drawn.

6.5.3. Measurement conditions

The next step focused on checking measurement conditions affecting the reliability and repeatability of the results. Following test cases were checked:

1. **different light conditions** (daylight, night, halogen, led, incandescent), other parameters strictly defined (scanning distance equals 30 centimeters, no eyeglasses, face in front of the camera, not rotated, used device: D1):

(a) **percentage of properly detected areas** - in this experiment in each light condition the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentages of all detected areas (area of the face, the eyes and the mouth) were calculated.

(b) **correctness of counted distances** - in this experiment in each light condition the face was being scanned for two minutes and all frames together with calculated distances were saved to the archive file. After that the distances between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

(c) **value of colours difference** - in this experiment in each light condition the face was being scanned for two minutes and all frames together with values of the most frequent colors of two halves of the face were saved to the archive file. Then, for each light source for 5 chosen frames the color difference between these two colors were calculated and plot on the graph.

2. **different scanning distances** (20, 30, 50, 70 centimeters), other parameters strictly defined (daylight, hair tied, no eyeglasses, face in front of the camera, not rotated, used device: D1):

   (a) **percentage of properly detected areas** - in this experiment for each scanning distance the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentages of all detected areas (area of the face, the eyes and the mouth) were calculated.

   (b) **correctness of counted distances** - in this experiment for each scanning distance the face was being scanned for two minutes and all frames together with calculated distances were saved to the archive file. After that the distances between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

   (c) **value of colours difference** - in this experiment for each scanning distance the face was being scanned for two minutes and all frames together with values of the most frequent colors of two halves of the face were saved to the archive file. Then, for each light source for 5 chosen frames the color difference between these two colors were calculated and plot on the graph.

3. **different personal features** (glasses, tied hair, loose hair), other parameters strictly defined (daylight, scanning distance equals 30 centimeters, face in front of the camera, not rotated, used device: D1):

    (a) **percentage of properly detected areas -** in this experiment for each feature the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentages of all detected areas (area of the face, the eyes and the mouth) were calculated.

    (b) **correctness of counted distances -** in this experiment for each feature the face was being scanned for two minutes and all frames together with calculated distances were saved to the archieve file. After that the distances between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

4. **head rotation in horizontal and vertical planes** (positive and negative slope of 30 degree in vertical plane, positive and negative slope of 45 degree in vertical plane), other parameters strictly defined (daylight, scanning distance equals 30 centimeters, hair tied, no eyeglasses, used device: D1):

    (a) **percentage of properly detected areas -** in this experiment for each slope the face was being scanned for two minutes and all frames with marked areas were saved to the archive file. After that the percentages of all detected areas (area of the face, the eyes and the mouth) were calculated.

    (b) **correctness of counted distances -** in this experiment for each slope the face was being scanned for two minutes and all frames together with calculated distances were saved to the archive file. After that the distances between eye centers and mouth corners for 5 random frames were calculated in Gimp application and compared with distances calculated by the application in order to determine whether it works correctly.

(c) **value of colours difference** - in this experiment for each slope the face was being scanned for two minutes and all frames together with values of the most frequent colors of two halves of the face were saved to the archive file. Then, for each light source for 5 chosen frames the color difference between these two colors were calculated and plot on the graph.

6.5.4. Examination

The examinations were made three times at weekly intervals. Patients were asked to perform tests in the constant environmental conditions. The suggested conditions include daylight, the face not rotated and place in front of the camera and the scanning distance no longer than 30 centimeters. They had to perform 3 facial gestures during each examination. The results were saved to files. The aim of this part of the project was to check whether there is a possibility to detect any change in the skin colour difference or any change in the relation between left and right distances from an eye center to a mouth corner for people who:

- suffer from facial paralysis and are rehabilitated in order to detect changes in the facial functions and evaluate therapy's effectiveness

- people who do not suffer from facial paralysis in order to find some new abnormalities

The testing group consists of 5 people with different features collected in the table 6.5. It is a small group of people, however at this step of working on a project it is enough to draw basic conclusions about the application after this number of responses.

id	gender	age	facial paralysis	rehabilitation
P1	*male*	*50-60*	*short episodes/ very high blood pressure*	no
P2	*female*	*8-12*	*no*	-
P3	*male*	*20-30*	*yes / Bell's palsy diagnosed in January*	*yes from January*
P4	*female*	*50-60*	*yes / Bell's palsy diagnosed in April*	*no*
P5	*female*	*20-30*	*no*	-

Table 6.5. Group of tested patients

6.6. Results

6.6.1. An impact of the used device on the results

The aim of this research was to determine whether different devices allow to get similar results and whether these results can be independent on the used software and hardware.

1. **percentage of properly detected areas -** the results were collected in the table 6.6 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

	percentage of frames with recognized area		
area	D1	D2	D3
face	*100 %*	*100 %*	*100 %*
left eye	*100 %*	*100 %*	*100 %*
right eye	*100 %*	*100 %*	*100 %*
mouth	*100 %*	*100 %*	*100 %*

Table 6.6. Percentage of detected areas - different personal features

2. **correctness of counted distances -** the results were collected in the table 6.7 taking into account relations between distances computed

manually (using the Gimp application) and automatically (by the application).

	approximation errors		
frame	D1	D2	D3
1	*0.022*	*0.013*	*0.019*
2	*0.019*	*0.012*	*0.017*
3	*0.080*	*0.022*	*0.022*
4	*0.029*	*0.019*	*0.028*
5	*0.062*	*0.027*	*0.051*

Table 6.7. Calculated approximation errors - different devices

Analyzing above results, one can see that different devices do not affect results very much. For all of the devices the percentage of detected areas is the same. Very similar results were also achieved for the calculated approximation errors. The difference is negligible (parts per hundred) and probably are result of the measurement error caused by the finite precision of measuring tools and imperfection of the person performing the measurement. A standard deviation equals 0 indicates that the data points tend to be close to the mean and in our case the standard deviation has a very small value. For the rest of tests device D1 was used.

6.6.2. A correctness of the application algorithms

The example picture taken during this part of tests is presented in the figure 6.1. As can be seen all areas and values were computed correctly.

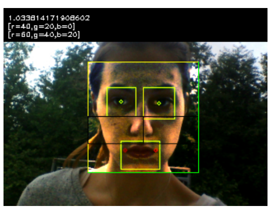


Figure 6.1. Example frame - strictly defined measurement conditions

1. **percentage of properly detected areas -** the results were collected in the table 6.8 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

area	percentage of frames with recognized area
face	*100 %*
left eye	*100 %*
right eye	*100 %*
mouth	*100 %*

Table 6.8. Percentage of detected areas - strictly defined measurement conditions

   Analyzing above results one can assume that measurement conditions were selected correctly. In all cases the percentage of recognized areas equals 100 %.

2. **correctness of counted distances -** the results were collected in the table 6.9 taking into account relations between distances computed manually (using the Gimp application) and automatically (by the application). Detected points were analyzed and if necessary new points were added and new distances were computed. The results are shown in the figure 6.2.

frame	relation computed automatically	relation computed manually	error
1	*0.999*	*0.977*	*0.022*
2	*0.999*	*0.980*	*0.019*
3	*1.038*	*0.958*	*0.08*
4	*0.999*	*1.028*	*0.029*
5	*0.998*	*0.936*	*0.062*

Table 6.9. Calculated distances and their errors - strictly defined measurement conditions

In this test case, similar as for experiment 6.6.1, the approximation error has rather small values and most probably it is the result of finite precision of measuring tools and imperfection of the person performing the measurement. Taking it into account one can assume that under strictly defined conditions it is possible to achieve the reliable and repeatable results.

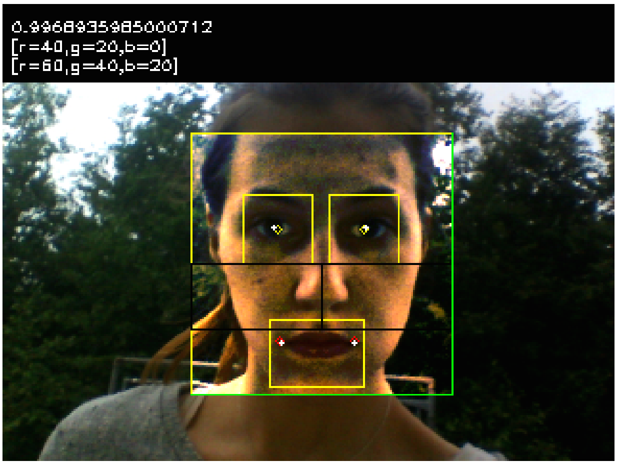


Figure 6.2. Example frame - new points added manually

3. **visibility of color changes after their magnification** - in this test the color of the specific point in the center of the forehead was picked from the pictures. The chosen results are collected in the table 6.10

frame	R	G	B	Color
1	*71*	*61*	*37*	
2	*82*	65	40	
3	*82*	66	29	
4	*69*	58	41	
5	*80*	77	46	

Table 6.10. Picked colors - strictly defined measurement conditions

The values of the colors from each channel were plotted on the graph presented in the figure 6.3.

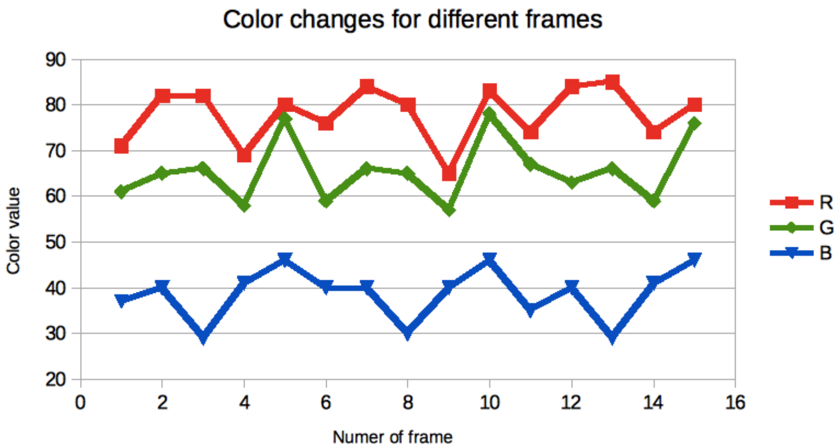


Figure 6.3. Color changes in sequence of frames

As can be seen channels (mainly the green one) changes periodically. The heart rate may be calculated based on these changes, however it is the subject of other studies and should be considered separately. Magnifying colours and motions changes allowed to strengthen slight variations in brightness produced by the flow of blood through blood vessels in the face, what may be useful in analysing facial paralysis.

6.6.3. Measurement conditions

1. **different light conditions** (daylight, night, halogen, led, incandescent) - a picture taken in different light conditions is presented in the figure 6.4.

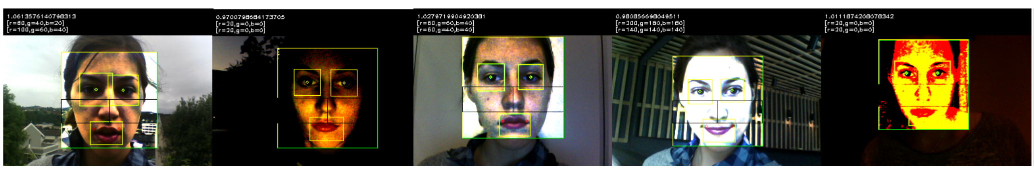


Figure 6.4. A picture taken during examination in different light conditions, from the left: daylight, night, halogen, led, incandescent

(a) **percentage of properly detected areas** - the results were collected in the table 6.11 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

	percentage of frames with recognized area				
area	daylight	night	halogen	led	incandescent
face	*100 %*	*100 %*	*100 %*	*100 %*	*100 %*
left eye	*100 %*	*40 %*	*100 %*	*100 %*	*100 %*
right eye	*100 %*	*47 %*	*100 %*	*100 %*	*100 %*
mouth	*100 %*	*96 %*	*34 %*	*55 %*	*100 %*

Table 6.11. Percentage of detected areas - different light conditions

The results collected in the table 6.11 indicate that light conditions have influence on the percentage of the recognized areas. The best effects are achieved for daylight and incandescent light. In halogen and led light there are some problems with detecting mouth area, however eyes are found in 100 % of situations. What is interesting during the night when the light conditions are very poor, the mouth are recognized very good.

(b) **correctness of counted distances** - the approximation errors caused by incorrect detection of characteristic points and as a result miscalculation of the distances were collected in the table 6.12

If the areas are recognized correctly, the approximation errors are small. Taking it into account one can assume that light conditions do not have influence on the correctness of the counted distances but have influence on finding characteristic points and areas.

(c) **value of colours differences** - the values of color differences calculated as CIE 1976 3.4.3 were collected in the table 6.14. The left and the right most frequent color were also added to the cells below the difference values.

	approximation errors				
frame	daylight	night	halogen	led	incandescent
1	*0.086*	*0.060*	*error*	*error*	*0.160*
2	*0.031*	*0.050*	*error*	*error*	*0.150*
3	*0.020*	*0.010*	*0.04*	*0.020*	*0.090*
4	*0.033*	*0.019*	*0.05*	*0.040*	*0.080*
5	*0.014*	*0.040*	*error*	*0.080*	*0.140*

Table 6.12. Approximation errors caused by incorrect detection of characteristic points - different light conditions

Table 6.13. My caption

	CIE1976 value				
frame	daylight	night	halogen	led	incandescent
1	*7.3273*	*0*	*25.9235*	*7.6177*	0
2	*7.3273*	*0*	*26.2602*	*7.6177*	0
3	*8.7745*	*0*	17.0487	*19.3631*	0
4	*0*	*0*	17.0487	19.3631	0
5	*8.7745*	*0*	*17.0487*	*19.3631*	0

Table 6.14. CIE1976 values - different light conditions

Analyzing results collected in the table 6.14, one can see that in a daylight the most frequent colors are detected correctly (colours similar to skin colour). Moreover the difference is almost constant, so the results may be reliable. However they are only preliminary and it is necessary to perform more tests. At night and in incandescent light it is not possible to detect colours difference. In halogen and led light colour differences change much between frames.

2. **different scanning distances** (20, 30, 50, 70 centimeters) - a picture taken at different scanning distances is presented in the figure 6.5.

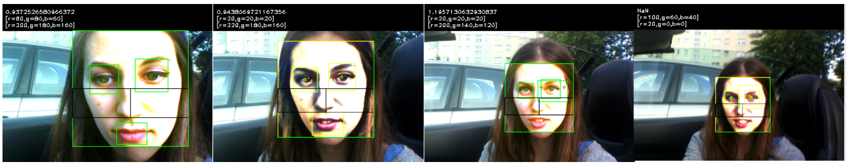


Figure 6.5. A picture taken at different scanning distances, from the left: 20, 30, 50, 70 cm

(a) **percentage of properly detected areas** - the results were collected in the table 6.15 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

	percentage of frames with recognized area			
area	20 cm	30 cm	50 cm	70 cm
face	*100 %*	*100 %*	*100 %*	*100 %*
left eye	*96 %*	*100 %*	*58 %*	*0 %*
right eye	*37 %*	*100 %*	*54 %*	*0 %*
mouth	*46 %*	*100 %*	*12 %*	*0 %*

Table 6.15. Percentage of detected areas - different scanning distances

The results collected in the table 6.15 indicate that scanning distances have influence on the percentage of the recognized areas. The facial area was recognized for all of the checked distances. The best effects are achieved for distance of 30 centimeters. For longer distances the percentage of recognized eyes and mouth areas was rapidly decreasing. For distances longer that 70 centimeters these areas were invisible. For distances smaller than 30 centimeters the results were also worse.

(b) **correctness of counted distances** - the approximation errors caused by incorrect detection of characteristic points and as a result miscalculation of the distances were collected in the table 6.16

	approximation errors			
frame	20	30	50	70
1	*0.030*	*0.020*	*error*	*error*
2	*0.031*	*0.023*	*error*	*error*
3	*error*	*0.016*	*error*	*error*
4	*error*	*0.009*	*error*	*error*
5	*error*	*0.012*	*error*	*error*

Table 6.16. Approximation errors caused by incorrect detection of characteristic points - different scanning distances

If the areas are recognized correctly, the approximation errors are small. Taking it into account one can assume that scanning distances do not have influence on the correctness of counting distances but have influence on finding characteristic points and areas.

(c) **values of colours differences** - the values of color differences calculated as CIE 1976 3.4.3 were collected in the table 6.17. The left and the right most frequent color were also added to the cells below the difference values.

Analyzing results collected in the table 6.17, one can see that scanning distances do not have influence on the repeatability of the colours difference. In all of test cases the difference is almost constant, so the results may be reliable. However they are only preliminary and it is necessary to perform more tests.

3. **different personal features** (glasses, tied hair, loose hair) - a picture taken for different personal features is presented in the figure 6.6.

	CIE1976 value			
frame	20 cm	30 cm	50 cm	70 cm
1	*22.7584*	*20.8904*	26.6542	33.2575
2	*22.7584*	*20.8904*	26.6542	33.2575
3	*22.7584*	*20.8904*	26.6542	33.2575
4	*22.7584*	*20.8904*	26.6542	33.2575
5	*50.3127*	*20.8904*	26.6542	33.2575

Table 6.17. CIE1976 values - different scanning disctances

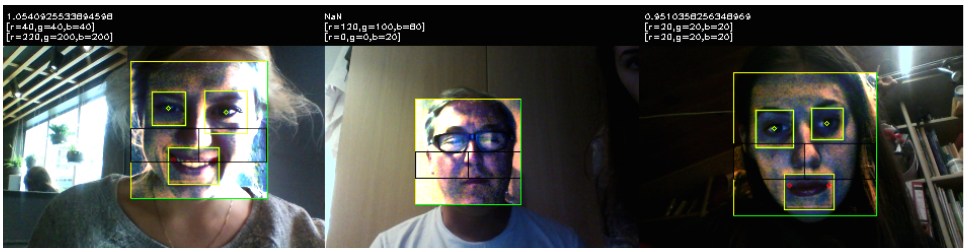


Figure 6.6. A picture taken for different personal features

(a) **percentage of properly detected areas** - the results were collected in the table 6.18 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

Following conclusions can be made. It is not possible to detect eyes areas if a person wear eyeglasses. The mouth area is found less often if a person has loose hair. The best results are achieved for tied hair.

(b) **correctness of counted distances** - the approximation errors

	percentage of frames with recognized area		
area	glasses	tied hair	loose hair
face	*100 %*	*100 %*	*100 %*
left eye	*0 %*	*100 %*	*0 %*
right eye	*0 %*	*100 %*	*25 %*
mouth	*100 %*	*100 %*	*15 %*

Table 6.18. Percentage of detected areas - different personal features

caused by incorrect detection of characteristic points and as a result miscalculation of the distances were collected in the table 6.19

	approximation errors		
frame	tied hair	loose hair	glasses
1	*0.020*	*0.030*	*error*
2	*0.023*	*0.020*	*error*
3	*0.016*	*0.015*	*error*
4	*0.009*	*0.033*	*error*
5	*0.012*	*0.040*	*error*

Table 6.19. Approximation errors caused by incorrect detection of characteristic points - different personal features

If the areas are recognized correctly, the approximation errors are small. Taking it into account one can assume that personal features do not have influence on the correctness of counting distances but have influence on finding characteristic points and areas.

4. **head rotation in horizontal and vertical planes** (positive and negative slope of 30 degree in vertical plane, positive and negative slope of 45 degree in vertical plane) - a picture taken for different head positions is presented in the figure 6.7.

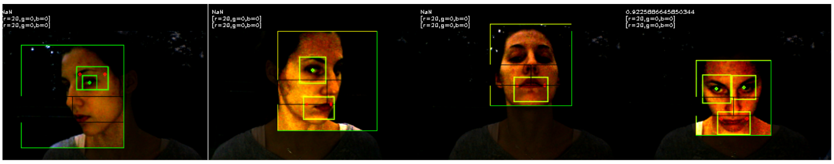


Figure 6.7. A picture taken for different head positions, from left: + 45 horizontal, -45 horizontal, +30 vertical, -30 vertical

(a) **percentage of properly detected areas** - the results were collected in the table 6.20 taking into account the percentage of frames with detected face area, left eye area, right eye area and mouth area.

	percentage of frames with recognized area			
area	+45 horizontal	-45 horizontal	+30 vertical	-30 vertical
face	*100 %*	*100 %*	*100 %*	*100 %*
left eye	*58 %*	*0 %*	*42 %*	*0 %*
right eye	*0 %*	*100 %*	*75 %*	*0 %*
mouth	*0 %*	*50 %*	*46 %*	*100 %*

Table 6.20. Percentage of detected areas - head rotation

Obviously after rotating head in a horizontal plane, only one eye may be found. However taking into account the results from the table 6.20 one can realize that better results are achieved if a person moves the head upwards than the downwards.To achieve reliable results the head should be placed in front of the camera.

(b) **correctness of counted distances** - the approximation errors caused by incorrect detection of characteristic points and as a result miscalculation of the distances were collected in the table 6.21

If the areas are recognized correctly, the approximation errors are small. Taking it into account one can assume that head rotation do not have influence on the correctness of counting distances but have influence on finding characteristic points and areas.

	approximation errors			
frame	+45 horizontal	-45 horizontal	+30 vertical	-30 vertical
1	*error*	*error*	*error*	*error*
2	*error*	*error*	*0.091*	*error*
3	*error*	*error*	*0.045*	*error*
4	*error*	*error*	*0.031*	*error*
5	*error*	*error*	*error*	*error*

Table 6.21. Approximation errors caused by incorrect detection of characteristic points - different head rotations

(c) **values of colours differences** - the values of color differences calculated as CIE 1976 3.4.3 were collected in the table 6.22. The left and the right most frequent color were also added to the cells below the difference values.

Moving the head upwards or downwards does not have influence on colours difference - the values are constant.

6.6.4. Examination

After performing theoretical tests, the practical examinations were taken by a group of people (6.5). They were performing the test once a week for three times. During every examination they had to make three facial gestures (smile, sad, e sound). Then, for each of the patient:

1. a color difference between most frequent colors from two halves of face were calculated and plotted
2. the discrepancy between values A and B was calculated and plotted, where:
   - A - an exact value of the relation between left and right distance from eye center to mouth corner (in perfect situation equals 1)
   - B - the calculated value of the relation between left and right distance from eye center to mouth corner calculated in the application

	CIE1976 value			
frame	+45 horizontal	-45 horizontal	+30 vertical	-30 vertical
1	*one half hidden*	*one half hidden*	*32.7842*	*32.7842*
2	*one half hidden*	*one half hidden*	*32.7842*	*32.7842*
3	*one half hidden*	*one half hidden*	*32.7842*	*32.7842*
4	*one half hidden*	*one half hidden*	*32.7842*	*32.7842*
5	*one half hidden*	*one half hidden*	*32.7842*	*32.7842*

Table 6.22. CIE1976 values - different head rotations

Then, for each of the line between calculated points, its slope was computed in order to decide whether there is a change in the patient's state of the health. The results are presented in the next section.

Results of the examination

Patient 1

Patient 1 had a very high blood pressure what sometimes leaded to short episodes of the facial paralysis. During the examination he didn't suffer from this disease, what can be seen in a zero value of colour difference. Moreover over 3 weeks the patient was making exercises for facial muscles what can be observed in the reduced value of approximation errors of calculated relation between left and right distance during making smile and e sound gesture. The experiments for this patients should last longer in order to notice the direction of the trend line.

Patient 2

Patient 2 did not have a facial paralysis. Beside some exceptions the colour difference was rather constant and small what may confirm the lack of the facial paralysis. The patient was also making the exercises during the examination process. Maybe that is why the approximation errors of calculated relation between left and right distance during making sad and e sound gesture were decreasing. On the other hand the approximation errors for smile gesture were increasing, so the diagnose can be clearly defined. The experiments for this patients should last longer in order to notice the direction of the trend line.

Patient 3

Patient 3 had a Bell's palsy diagnosed in January, what can be seen at his left eye, which is closed more than the right. He has had the rehabilitation since then. Moreover over 3 weeks the patient was making exercises for facial muscles. The achieved results shows that this therapy may not be effective, because the approximation errors are increasing rapidly for smile and sad gestures. On the other hand for e sound gesture this error is decreasing. So the experiments for this patients should last longer in order to notice the direction of the trend line. However during the experiments the color difference decreased rapidly. At first it was very high so one can assume that Bell's palsy may be diagnosed by analyzing colours differences. Then it was decreased what may be the result of the exercises. Yet the results are only preliminary and they should be confirmed.

Patient 4

Patient 4 had a Bell's palsy diagnosed in April, what can be seen at her left mouth corner, which is lower than the right. She has not had the rehabilitation since then. However over 3 weeks the patient was making exercises for facial muscles, what resulted in the decreased values of approximation errors over time. The colour differences were high but rather constant, so the therapy did not have influence on this parameter.

Patient 5

Patient 5 did not have a facial paralysis. However the colours differences were very high and the approximation errors were increasing rapidly what can mean that some anomalies may appear. The patient should be observed for a longer period of time and have a consultation with the specialist.

The pictures, graphs and calculated values are presented below. **Patient 1**

Examination	1	2	3
CIE 1976 smile	0	0	0
CIE 1976 sad	0	11.4383	0
CIE 1976 e sound	0	0	0

Table 6.23. Patient 1 CIE 1976 values

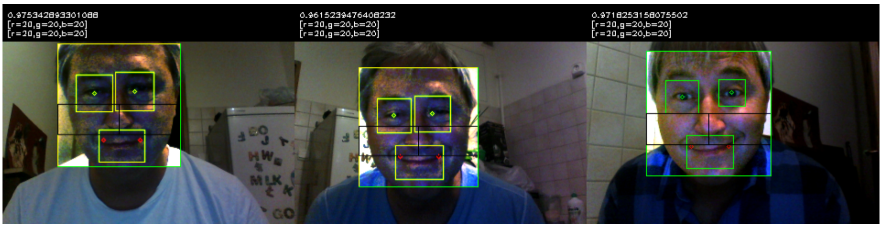


Figure 6.8. Patient 1 - smile

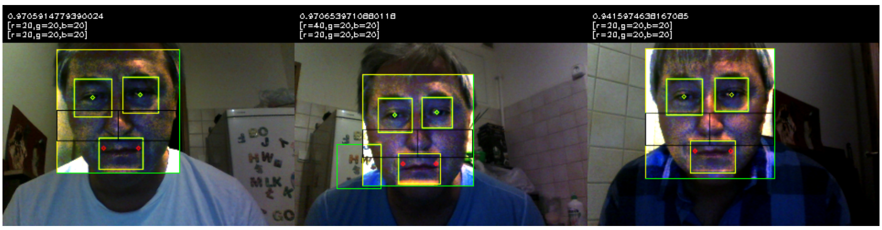


Figure 6.9. Patient 1 - sad

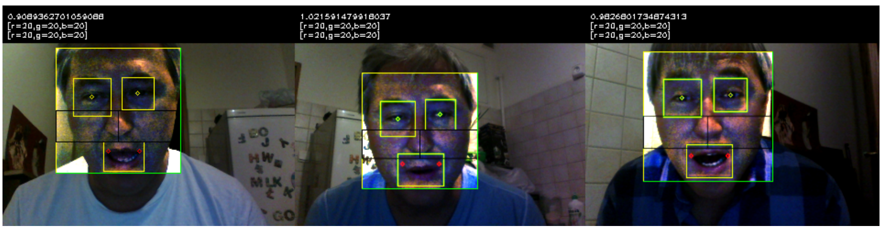


Figure 6.10. Patient 1 - e sound

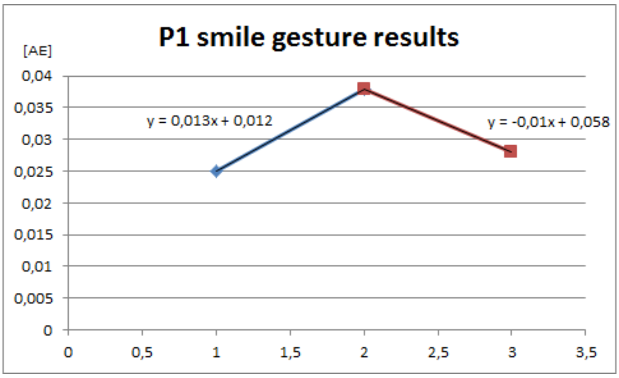


Figure 6.11. Patient 1 - smile - graph

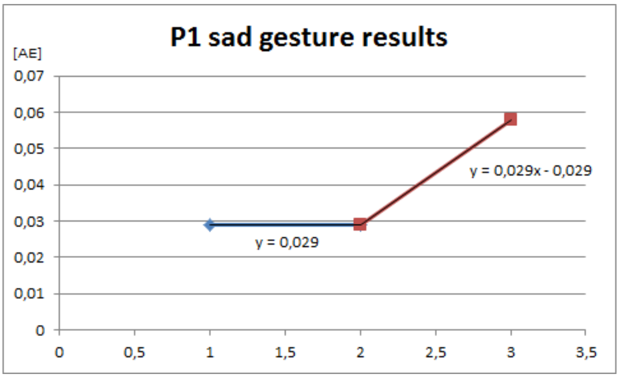


Figure 6.12. Patient 1 - sad - graph

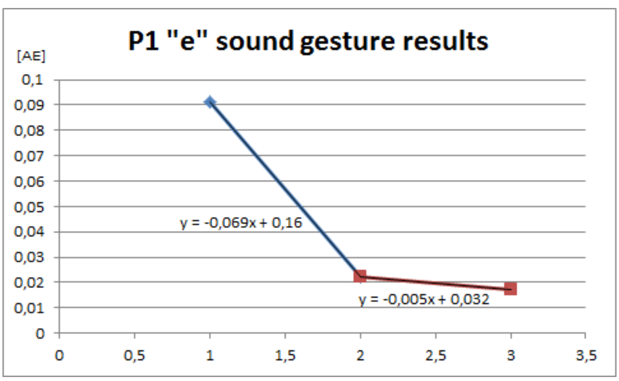


Figure 6.13. Patient 1 - e sound - graph

Patient 2

Examination	1	2	3
CIE 1976 smile	9.7948	9.7948	9.7948
CIE 1976 sad	8.967	9.7948	20.737
CIE 1976 e sound	14.6442	9.7948	9.7954

Table 6.24. Patient 2 - CIE 1976 values

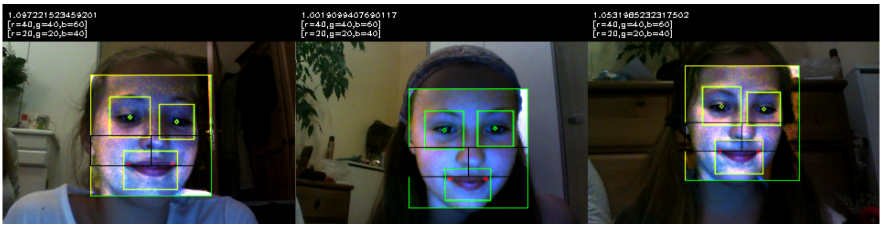


Figure 6.14. Patient 2 - smile

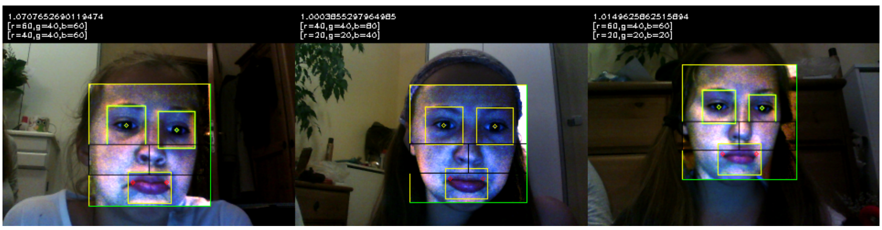


Figure 6.15. Patient 2 - sad

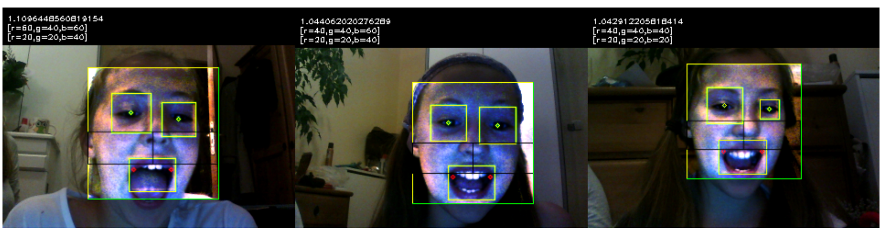


Figure 6.16. Patient 2 - e sound

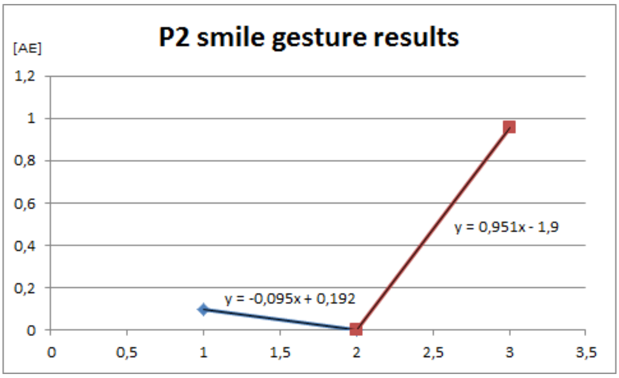


Figure 6.17. Patient 2 - smile - graph

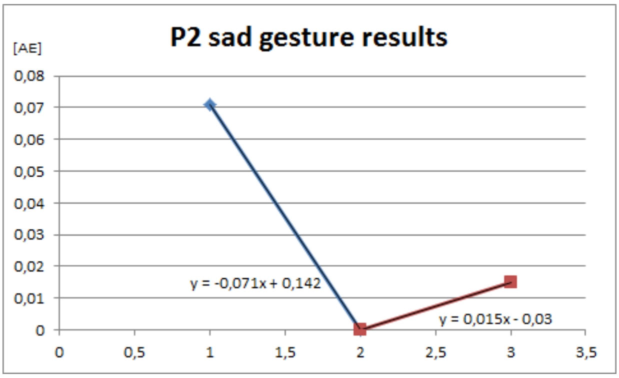


Figure 6.18. Patient 2 - sad - graph

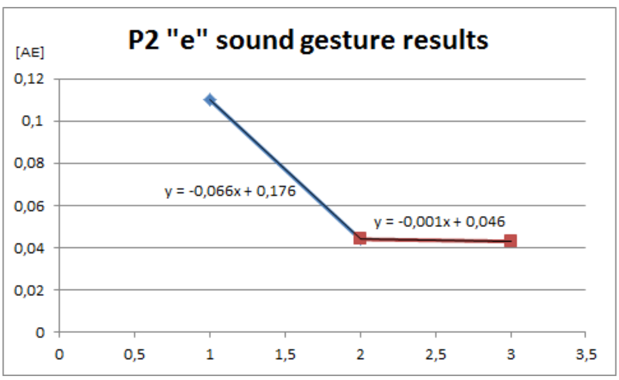


Figure 6.19. Patient 2 - e sound graph

Patient 3

Examination	1	2	3
CIE 1976 smile	64.6381	18.9978	0
CIE 1976 sad	17.9997	18.9978	0
CIE 1976 e sound	57.5997	23.1538	0

Table 6.25. Patient 3 - CIE 1976 values

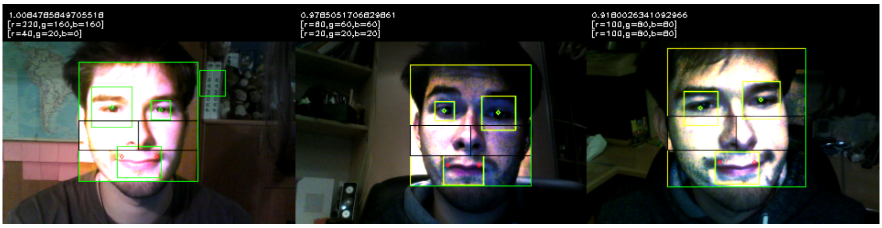


Figure 6.20. Patient 3 - smile

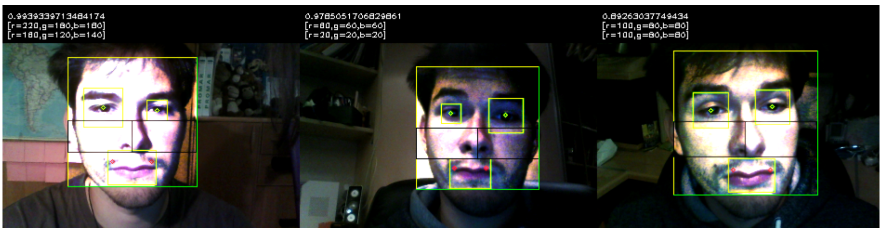


Figure 6.21. Patient 3 - sad

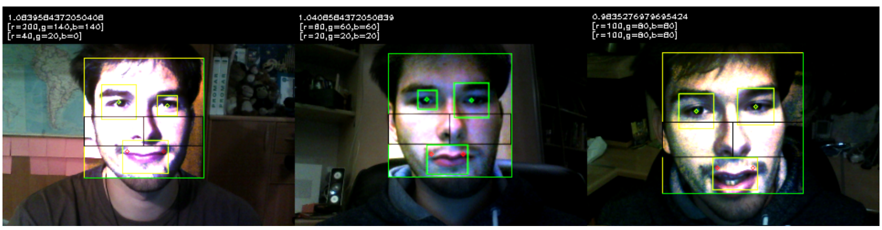


Figure 6.22. Patient 3 - e sound

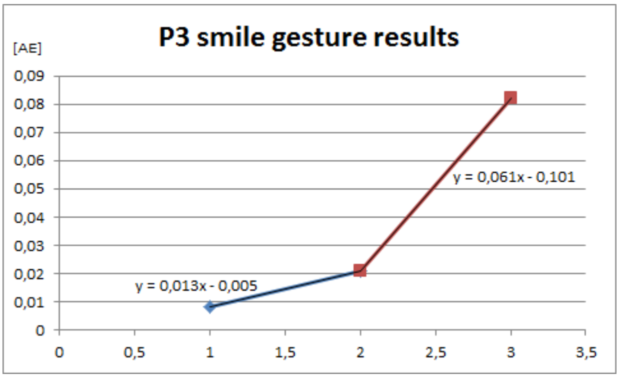


Figure 6.23. Patient 3 - smile - graph

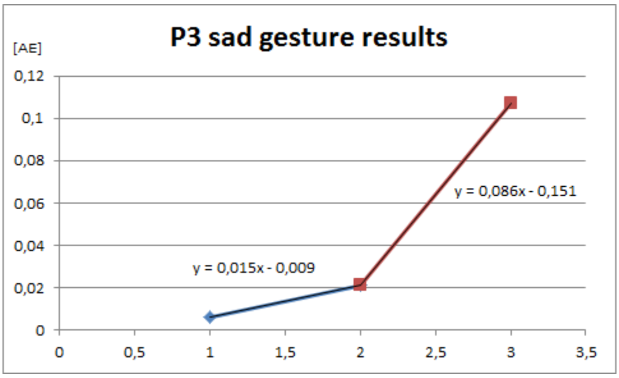


Figure 6.24. Patient 3 - sad - graph

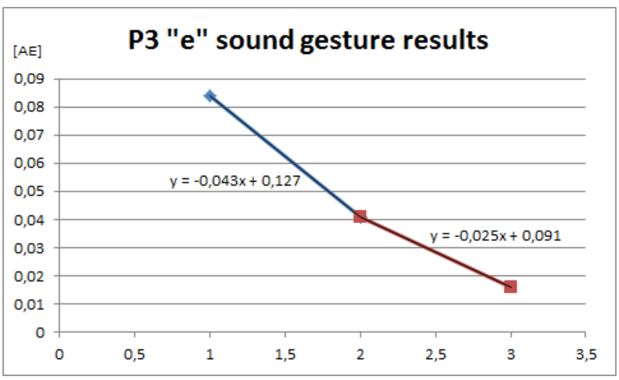


Figure 6.25. Patient 3 - e sound - graph

Patient 4

Examination	1	2	3
CIE 1976 smile	10.9874	8.3758	21.351
CIE 1976 sad	8.9958	8.3758	22.5839
CIE 1976	8.9958	10.0654	22.5839

Table 6.26. Patient 4 - CIE 1976 values

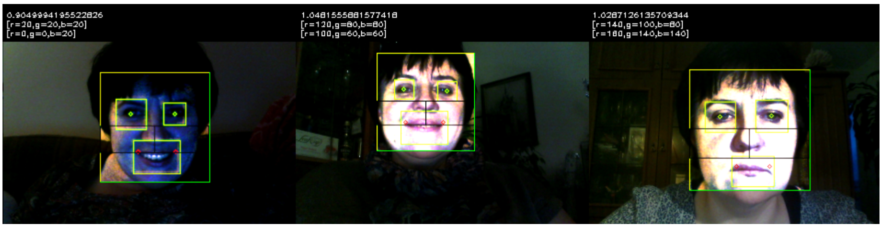


Figure 6.26. Patient 4 - smile

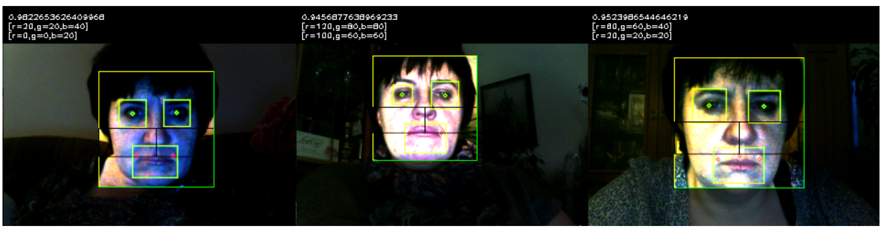


Figure 6.27. Patient 4 - sad

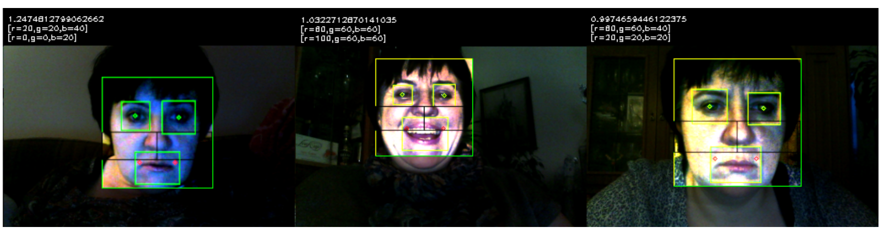


Figure 6.28. Patient 4 - e sound

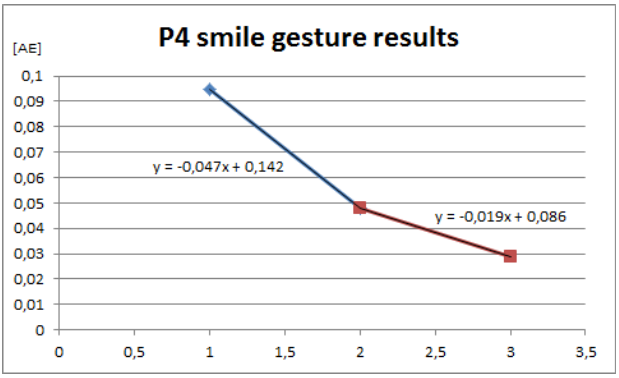


Figure 6.29. Patient 4 - smile - graph

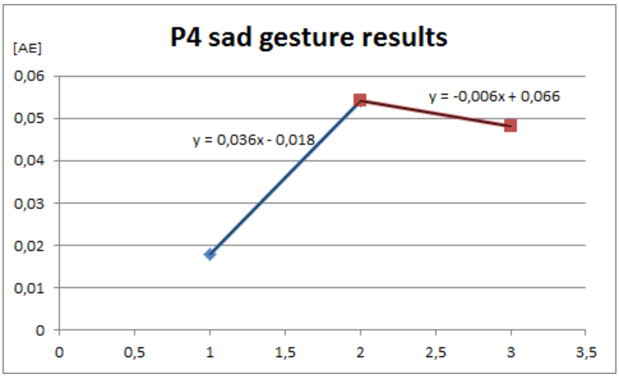


Figure 6.30. Patient 4 - sad -graph

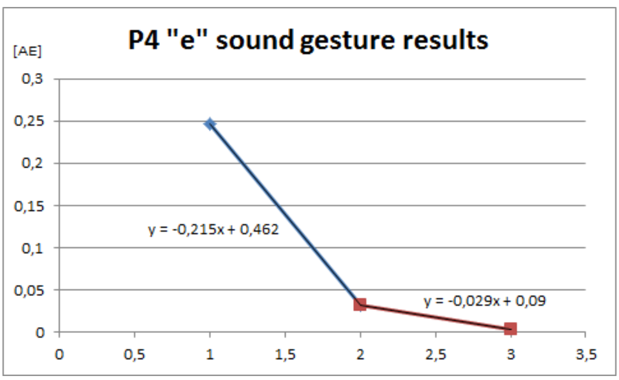


Figure 6.31. Patient 4 - e sound - graph

Patient 5

Examination	1	2	3
CIE 1976 smile	22.5839	0	68.338
CIE 1976 sad	22.5839	7.1235	68.338
CIE 1976	28.7722	0	68.338

Table 6.27. Patient 5 - CIE 1976 values

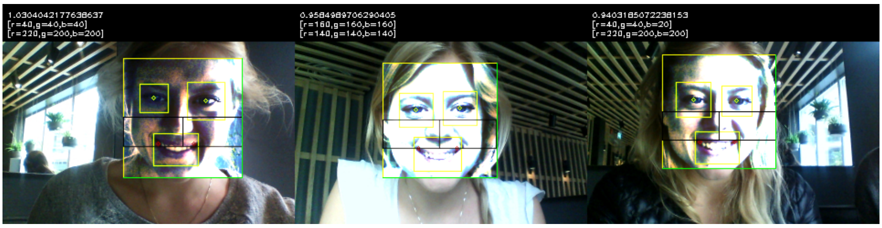


Figure 6.32. Patient 5 - smile

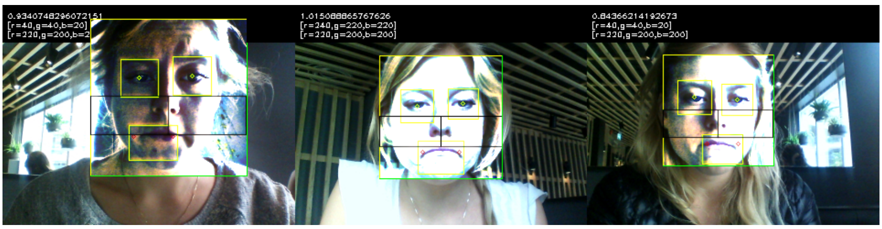


Figure 6.33. Patient 5 - sad

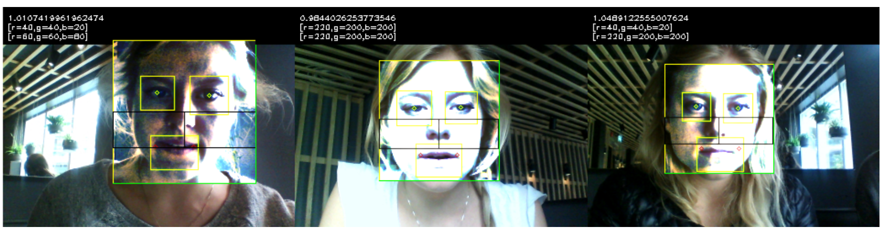


Figure 6.34. Patient 5 - e sound

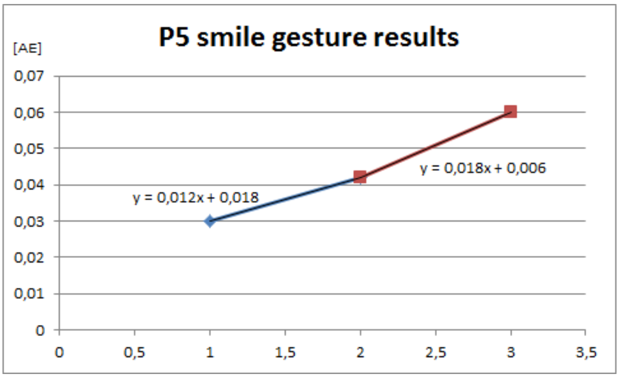


Figure 6.35. Patient 5 - smile - graph

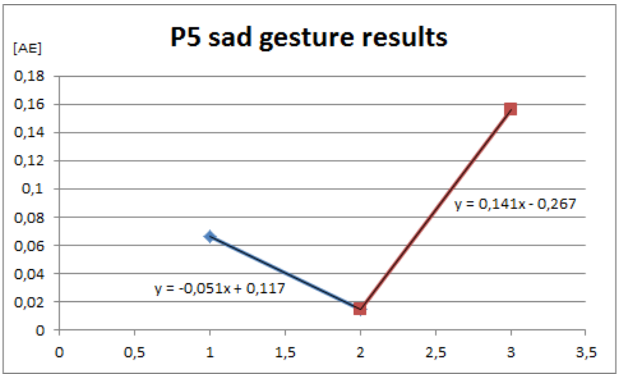


Figure 6.36. Patient 5 - sad - graph

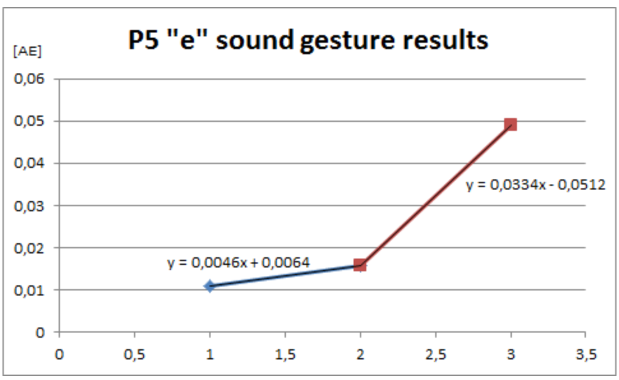


Figure 6.37. Patient 5 - e sound - graph

Id	age	gender	paralysis	rehabil-itation	slope_e1	slope_e2	CIE_e_1-2	CIE_e_2-3
P1	50-60	m	s	n	-	-	0	0
P2	8-12	f	n	n	-	-	-	+
P3	20-30	m	y	y	-	-	-	-
P4	50-60	f	y	y	-	-	+	+
P5	20-30	f	n	n	+	+	-	+

Table 6.28. Analysed dataset

Analyzing collected data

The Apriori algorithm was used to define some association rules in a data set for a chosen gesture (e sound). This algorithm helps to find which variables can depend on others. Association rule learning is a popular method for discovering interesting relations between variables. For example, the rule can look as follows

$$\{bread, cheese\}ß\{butter\}$$

It means that if someone buys bread and cheese, he buys also butter. Such information can be used as the basis for decisions about marketing activities. The calculations were made in the Weka program.

The analyzed data set was collected in the table 6.28.

Where:

1. **Id**: patient number
2. **age**: patient age (1 →8-20 years, 2 →20-30 years, 3 →50-60 years)
3. **gender**: m →male, f →female
4. **paralysis**: y →yes, n →no, s →sometimes short episodes
5. **rehabilitaton**: y →yes, n →no
6. **slope_e1** - slope of the line between points of approximation errors of calculated distances relations (between examination 1 and 2); p →positive, m →negative, c →constant

7. **slope_e2** - slope of the line between points of approximation errors of calculated distances relations (between examination 2 and 3); p →positive, m →negative, c →constant

8. **CIE_e_1-2** - slope of the line between points of colours differences calculated as CIE1976 (between examination 1 and 2); p →positive, m →negative, c →constant

9. **CIE_e_2-3** - slope of the line between points of colours differences calculated as CIE1976 (between examination 2 and 3); p →positive, m →negative, c →constant

The results of Apriori algorithm are presented below. **Conf:** confidence is a conditional probability that the consequent will occur given the occurrence of the antecedent.

1. *slope_e2=m 4 →slope_e1=m 4 conf:(1) -*
   In every case where the slope_e2 is negative, the slope_e1 is also negative.

2. *slope_e1=m 4 →slope_e2=m 4 conf:(1) -*
   In every case where the slope_e1 is negative, the slope_e2 is also negative.

3. *CIE_e_2-3=p 3 →gender=f 3 conf:(1) -*
   If the CIE_e_2-3 is positive, it is sure that the woman is examined.

4. *gender=f 3 →CIE_e_2-3=p 3 conf:(1) -*
   If the woman is examined, she always has the positive value of the CIE_e_2-3.

5. *age=2 2 →CIE_e_1-2=m 2 conf:(1) -*
   If the person is in a second age group (20-30), he gets a negative value of the CIE_e_1-2.

6. *age=3 2 →slope_e1=m 2 conf:(1) -*
   If the person is in a third age group (50-60), he gets a negative value of the slope_e1.

7. *age=3 2 →slope_e2=m 2 conf:(1) -*
   If the person is in a third age group (50-60), he gets a negative value of the slope_e2.

8. *paralysis=n 2 →gender=f 2 conf:(1)* -
   If the person does not have a paralysis, it is a woman.

9. *gender=m 2 →slope_e1=m 2 conf:(1)* -
   If the person is a man, he always gets the negative value of the slope_e1.

10. *gender=m 2 →slope_2=m 2 conf:(1)* -
    If the person is a man, he always gets the negative value of the slope_e2.

6.7. Discussion

Performed tests indicate that the used device does not affect the results. Secondly, under strictly defined conditions the achieved results are very good. For the experiments which were run in daylight, at 30 centimeters from the camera, for a person who does not wear eyeglasses and sit in front of the camera, 100 % of the areas were detected correctly and approximation errors of calculated relations between left and right distances were small. Moreover magnifying colours and motions changes allowed to strengthen slight variations in brightness produced by the flow of blood through blood vessels in the face, what may be useful in analyzing facial paralysis.

Taking it into consideration one may assume that the algorithms were implemented correctly.

Furthermore, light conditions, scanning distances and head rotation have influence on percentage of recognized area but do not affect correctness of counting distances.

The most frequent colors are detected correctly (colours similar to skin colour) in a daylight. Scanning distances do not have influence on the repeatability of the colours difference, but to achieve reliable results the head should be placed in front of the camera and it should not be rotated.

It is not possible to detect eyes areas if a person wear eyeglasses. The mouth area is found less often if a person has loose hair. The best results are achieved for tied hair.

The results of the experiments confirm the hypothesis. By analysing changes in distances between characteristic points on the face it is possible to detect changes in the facial functions caused by face paralysis in order to evaluate therapy's effectiveness or find some new abnormalities. In the case of measuring changes in colours differences, it is much more difficult to formulate a clear diagnose as this parameter is influenced by measurement conditions such as light conditions or scanning distances. However some relationships may be noticed. For example the colour difference is high for patients who suffer from facial paralysis. Secondly the colour difference increased for a patient with Bell's palsy, who did not have a rehabilitation. One should remember that as pointed in the chapter 2, good therapy has influence one the possibility to recover, so this result may confirm this statement. On the other hand it is very probable that the patient who make the exercises will recover, so the decreasing value of colours difference may be a proof of this. Yet the results are only preliminary and it is required to conduct more surveys and allow patients to use the application for a longer time. Moreover the algorithms implemented in the application should be consulted with the specialist as they are scientific and innovative. Certainly combining this application with more professional diagnosis methods will result in a high quality product. Adding possibility for thermal imaging is a good idea because this technique has been found valuable in diagnosing paralysis of the face. There are more open issues that should be considered in the future.

6.8. Conclusion

In this chapter a series of in-depth analysis, examinations and researches on possibilities to detect changes in facial functions and geometry were conducted. Recognition of characteristic points and colours under different conditions and situations were also investigated. I believe that there are a lot of possibilities of using web browsers and HTML5 technology for telemedicine purposes if special requirements of measurement conditions are fulfilled.

Chapter 7

Conclusion and future work

7.1. Introduction

This chapter contains quick overview of the work, pointing out most important information. Moreover one can find below some ideas for future work and improvement.

7.2. Project overview

The aim of this project was to check whether it is possible to detect changes in the facial functions caused by the facial paralysis by providing algorithms for detecting geometrical and colours differences between two halves of the face during making sequence of special facial gestures in order to evaluate therapy effectiveness or find some new abnormalities. To achieve this goal, some new scientific methods were proposed. These algorithms allows to obtaining multi modal features of the face by analyzing structural and functional properties of the face. At first areas of the face, the eyes and the mouth were detected using Haar Feature-based Cascade Classifiers. Then, the most frequent colours of two halves of the face were calculated to find relationship between the paralysis and the colour of the face. In order to improve quality of proposed methods and make them work faster, the algorithms for magnifying motions and colours based on the last research from

Massachusetts Institute of Technology were applied. This combination uncover new dependencies between data set and lead to conclusions that were previously undiscovered. The first step of the project (chapter 2) was to find some similar solutions that take advantage of the telemedicine, because the aim of the project was to prepare a product for remote patient monitoring using web browsers. Moreover, this chapter includes also an in-depth analysis of the facial functions, signs and symptoms of the paralysis and ideas for facial motion and features recognition. The research shows that it is possible to measure the facial paralysis by performing a sequence of specific movements and assigning a grade of palsy based on clinical observation and subjective evaluation. Moreover it is also said that physical therapy and rehabilitation can primarily protect from permanent damage by strengthening muscles and there is a need to find an automatic tool which will allow to evaluate therapy effectiveness. After that some requirements based on gained knowledge were stated. Functional requirements were presented on use case diagrams to show the possible interaction between the application's actors and each modules (section 4.1). All of the use cases were described in a table with initial, alternative and final conditions to present each function under certain states. Furthermore non functional and system requirements were also defined. Taking into consideration requirements connected with user experience design (for example interfaces) the design of the system was prepared and described in the chapter 4. Next, based on created project, all of described functionalities were implemented and final interfaces were prepared. The chapter 5 consists the description of the implementation. At this step of the project the functionality of registering users was only shown as a design, because it was not important for analyzing facial paralysis defined in the chapter 3. The project was launched on the Amazon Web Services in order to access a wide range of users in the future. When the application was finished, a series of experiments were conducted both in a simulated environment and on a group of potential users. Unfortunately because of the medical data protection rules, which says that using personal data must only take place within a robust ethical framework and clear governance rules to mitigate privacy risks, only a few of patients agreed to take part in the experiment. However it was still possible to formulate some conclusions. Performed tests indicate that the used device does not affect the results very much. Secondly, one may achieve very good results if the conditions are strictly defined. Moreover magnifying colours and motions changes allowed to strengthen slight variations in brightness produced by the flow of blood through blood vessels in the face, what may be useful in analyzing facial paralysis. Moreover, some measurement conditions, for example light conditions, scanning distances and head rotation have influence on percentage of recognized areas but do not affect correctness of the counted distances. Furthermore, based on the results it can be assumed that by analyzing changes in distances between characteristic points on the face it is possible to detect changes in the facial functions caused by face paralysis in order to evaluate therapy's effectiveness or find some new abnormalities. For colours differences it is much more difficult to formulate a clear diagnose as this parameter is influenced by measurement conditions. However some relationships may be noticed. For example the colour difference is high for patients who suffer from facial paralysis. Yet these results are only preliminary and it is required to perform more tests.

7.3. Future work

Despite positive results, there is no hundred percent certainty that they are true. It is still required to perform more tests and allow patients to use the application for a longer time. Furthermore to confirm the quality of the implemented algorithms it is necessary to discuss them with the specialist. Moreover the application performance is quite poor and in future the implementaton should be improved in this way. What is more, new functionalities may be added and the application can be extended to a bigger module of the whole platform for remote patient monitoring. Last but not least, combining this application with professional diagnosis methods for example thermography will result in a high quality product.

7.4. Conclusion

While working on the project, all of the tasks mentioned in the chapter 1 were completed. Furthermore the aim of the project was achieved and some conclusions based on performed tests were formulated. Because of the positive results I believe that it is possible to monitor patients remotely via web browsers. I think that this project is a good start and first step for future work on developing an innovative tool for monitoring facial paralysis. Moreover the application has been launched on the internet for free, because the main idea of this product is to help people, not to earn money on it.

Bibliography

[1] A. Moghadas, M. Jamshidi, M. Shaderam, ”Telemedicine in healthcare system,” automation congress, Wac 2008. World , vol., no., pp.1,6,

[2] D. H. Gustafson, P.F Brenna, R.P. Hawkins ”Investing in e-health. What it takes to sustain consumer”, Health Informatics, Foreword by S.M. Shortell, phd., m.p.h.

[3] ISO/TR 18307 Health informatics - ”Key characteristics for interoperability and compatibility in messaging and communications standards”, 2002

[4] G. Nadler, ”Work design, homewood”, 1963 il: Irwin Publishers.

[5] W.E. Deming, ”Out of the crisis. Center for advanced engineering study”, 1986, Massachusetts institute of technology.

[6] R. M. Saleem, A. Muhammad, A. M. Martinez-Enriquez, ”Remote patient monitoring and healthcare management using multi-agent based architecture,” Artificial Intelligence (micai), 2010 ninth mexican international conference on , vol., no., pp.118,123, 2010

[7] J. K. Taitsman, m.d., j.d., C. M. Grimm, m.p.a., and S. Agrawal, m.d.”Protecting patient privacy and data security”, N engl j med 2013; 368:977-979 2013

[8] Owasp testing guide 2008 v3.0

[9] Health On the Net Foundation (HON), http://www.hon.ch, access 01.05.2015

[10] C. Boyer, M. Selby, J. R. Scherrer et al, ’The health on the net code of conduct for medical and health websites’, 1998, Comput Biol Med 28(5):603–610

[11] S. H. Fouladi, R. Chavez-Santiago, P. A. Floor, I. Balasingham, T.A. Ramstad, "Sensing, signal processing, and communication for wbans", 2014 ZTE Communications. vol. 12 (3).

[12] R. Chavez-Santiago, I. Balasingham, "Ultrawideband signals in medicine". 2014 IEEE signal processing magazine (print). Vol. 31 (6).

[13] C. Chakraborty, B. Gupta, S.K. Ghosh, "A review on telemedicine-based wban framework for patient monitoring", Telemed j e health, 2013 19(8): 619–626. Doi: 10.1089/tmj.2012.0215

[14] C. Uasong, V. Leung, C. Chow, H. Chan "Consumer communications and networking enabling technologies for wireless body area networks: a survey and outlook", University of British Columbia, The Hong Kong Polytechnic University

[15] Y. Jamil, M. Khan, R. Yuce, and K. Farbood, "Performance evaluation of a wireless body area sensor network for remote patient monitoring" engg. In medicine and biology society, 2008. Embs 2008. 30th annual international conference of the ieee.

[16] J. Kvedar, M. J. Coye, W. Everett "A Review Of Technologies And Strategies To Improve Patient Care With Telemedicine And Telehealth", Connected Health, Aff February 2014 33:2194-199; doi:10.1377/hlthaff.2013.0992

[17] Partners http://www.partners.org/, access 20.04.2015

[18] VA http://www.va.gov/health/, access 20.04.2015

[19] Kaiser Permanente https://healthy.kaiserpermanente.org/ html/kaiser/index.shtml, access 15.04.2015

[20] Memorial Health Care http://www.umassmemorialhealthcare.org/umass-memorial-medical-center/services-treatments/critical-care/services-we-provide/eicu, access 17.04.2015

[21] Connected Health http://connectedhealth.partners.org/, 27.04.2015

[22] "Anatomy of the Face and Head Underlying Facial Expression" http://www.face-and-emotion.com/dataface/anatomy/anatomy.jsp, access 01.04.2015

[23] Braus, Hermann, "Anatomie des Menschen: ein Lehrbuch für Studierende und Ärzte.", 1921 p. 777.

[24] M. Pantic, L.J.M. Rothkrantz, "Expert system for automatic analysis of facial expressions", Faculty of Information Technology and Systems, Department of Knowledge Based Systems, Delft University of Technology, P.O. Box 356, 2600 AJ Delft, The Netherlands

[25] L. K. Moore, A. F. Dalley, A. M. R. Agur, "Moore's clinical anatomy. United States of America: Lippincott Williams & Wilkins", 2010, pp. 843–980.

[26] K. Najarian, R. Splinter, "Biomedical Signal and Image Processing", Taylor & Francis Group, 2006

[27] Patrick J. Lynch, "Cranial nerve VII", permission: Creative Commons Attribution 2.5 License 2006

[28] "Facial (VII) nerve disorders", http://neuromuscular.wustl.edu/nanatomy/vii.htm, access 20.03.2015

[29] National Institute of Neurological Disorders and Stroke http://www.ninds.nih.gov/disorders/bells/detail_bells.htm, access 02.04.2015

[30] "Causes of facial paralysis. (n.d.).", http://www.nyee.edu/facialparalysis-causes.html, The New York Eye and Ear Infirmary. access 29.01.2015

[31] J. Dong, Q. Wang, S. Wang, L. Liu, "Evaluation of the Facial Paralysis Degree", College of Information Science and Engineering, Ocean University of China, Qingdao Haiser Medical Centre China

[32] J. House & D. Brackman, "Facial nerve grading system. Otolaryngol Head Neck Surgery", 1985, pp. 146-147

[33] Tessa A. Hadlock, MD; Luke S. Urban, MS, "Toward a Universal, Automated Facial Measurement Tool in Facial Reanimation", American Medical Association, 2012

[34] A. S. Fauci, T. R. Harrison, "Harrison's principles of internal medicine (17th ed.)", New York: McGraw-Hill Medical, 2008

[35] C. H. G. Beurskens, P. G. Heymans, "Positive Effects of Mime Therapy on Sequelae of Facial Paralysis: Stiffness, Lip Mobility, and Social and Physical Aspects of Facial Disability", Otology & Neurotology, Vol.24, No.4, July 2003, pp. 677-681

[36] C. B. Novak, "Rehabilitation Strategies for Facial Nerve Injuries. Seminars in Plastic Surgery.", 2004, 18(1):47-52. doi:10.1055/s-2004-823123.

[37] Bell's Palsy Information http://www.bellspalsy.ws/, access 02.05.2015

[38] Tian, Y.L., Kanade, T., Cohn, J.F.: Facial expression analysis. In: Li, S.Z., Jain, A.K. (eds.) "Handbook of Facial Recognition", Springer, Heidelberg (2004)

[39] M. Pantic, Student Member, IEEE, and Leon J.M. Rothkrantz, "Automatic Analysis of Facial Expressions: The State of the Art", IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 22, NO. 12, DECEMBER 2000

[40] M. Yoneyama, Y. Iwano, A. Ohtake, and K. Shirai, "Facial Expressions Recognition Using Discrete Hopfield Neural Net- works" Proc. Int'l Conf. Information Processing, vol. 3, pp. 117-120, 1997

[41] H. Kobayashi and F. Hara, "Facial Interaction between Animated 3D Face Robot and Human Beings" Proc. Int'l Conf. Systems, Man, Cybernetics,, pp. 3,732-3,737, 1997

[42] P. Menezes, J. C. Barreto, J. Dias, "Face tracking based on haar-like features and eigenfaces", IFAC/EURON Symposium on Intelligent Autonomous, 2004

[43] J. Barreto, P. Menezes, J. Dias, "Human-robot interaction based on Haar-like features and eigenfaces", In:International Conference on Robotics and Automation, 2004

[44] C. L. Huang and Y. M. Huang, "Facial Expression Recognition Using Model-Based Feature Extraction and Action Parameters Classification", J. Visual Comm. and Image Representation, vol. 8, no. 3, pp. 278-290, 1997

[45] M. Pantic and L. J. M. Rothkrantz, "Expert System for Automatic Analysis of Facial Expression" Image and Vision Computing J., vol. 18, no. 11, pp. 881-905, 2000.

[46] B. Knight, A. Johnston, "The role of movemen in face recognition", Visual Cognition 4, pp. 265-267

[47] T. F. Cootes, G. J. Edwards, C. J. Taylor, "Active Appearance Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(6), pp. 680-682, 2001

[48] P. Ekman, W. V. Friesen, "Facial Action Coding System (FACS): Manual", Consulting Psychologists Press, Palo Alto, 1978

[49] V. Bruce, "Recognizing Faces", Hove, East Sussex: Lawrence Erlbaum Assoc., 1986.

[50] H. Naver, Ch. Blomstrand, S. Ekholm, Ch. Jensen, T. Karlsson, G. Wallin, "Autonomic and thermal sensory symptoms and dysfunction after stroke", 1995, pp. 1379-1385

[51] Guan, Ling et al. "Infrared Thermography and Meridian-Effect Evidence and Explanation in Bell's Palsy Patients Treated by Moxibustion at the Hegu (LI4) Acupoint: Overall Regulation or a Specific Target?" Neural Regeneration Research 7.9 (2012): 680–685. PMC. Web. 19 May 2015

[52] Pedro Boloto Chambino, 'Android-based implementation of Eulerian Video Magnification for vital signs monitoring', FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO, 2013, pp. 21

[53] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Fredo Durand, William Freeman, 'Eulerian Video Magnification for Revealing Subtle Changes in the World', MIT CSAIL Quanta Research Cambridge, Inc, pp. 1-2

[54] P. C. Johnson, H. Brow, W. M. Kuzon, "Simultaneous quantification of facial movements: The maximal static response assay of facial nerve function", Annals of Plastic Surgery 5, (1994), pp.171-174,

[55] R. J. van Vark, L. J. M. Rothkrantz, E. J. H. Kerckhoffs, "Prototypes of multimedial stress assessment", in: MediaComm 95, SCS International, Ghent, 1995, pp. 108–112.

[56] D. Terzopoulos, K. Waters, "Analysis and synthesis of facial image sequences using physical and anatomical models", IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (6) (1993) 569–579.

[57] S. Morishima, F. Kawakami, H. Yamada, H. Harashima, "A Modelling of Facial Expression and Emotion for Recognition and Synthesis", Symbiosis of Human and Artifact, Elsevier, Amsterdam, 1995 (pp. 251 – 256).

[58] J. Zhao, G. Kearney, "Classifying facial emotions by backpropagation neural networks with fuzzy inputs", in: International Conference on Neural Information Processing, vol. 1, 1996, pp. 454–457.

[59] I. A. Essa, A. P. Pentland, "Coding analysis interpretation and recognition of facial expressions", IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 757–763.

[60] P. Ekman, W. V. Friesen, "Unmasking the Face", Prentice Hall, New Jersey, 1975.

[61] M. Lewandowska, J. Ruminski, T. Kocejko, J. Nowak, "Measuring pulse rate witha webcam - a non-contact method for evaluating cardiac activity", Proc. of Federated Conference on Computer Science and Information Systems (IEEE), 2011

[62] Dr. M. Anto Bennet, Professor, M. Manimaraboopathy, R. Srinath, P. Maragathavalli, S. Mekala, Assistant Professors, 'Frame Rate Up-Conversion using Trilateral Filtering For Video Processing', Department of ECE, Veltech, Chennai-600062

[63] Fredrik Vestermark, 'Implementation of a Frame Rate Up-Conversion Filter', May 24, 2013

[64] Tanaphol Thaipanich, Student Member, IEEE, Ping-Hao Wu, Student Member, IEEE and C.-C. Jay Kuo Fellow, IEEE, 'Low Complexity Algorithm for Robust Video Frame Rate Up-Conversion (FRUC) Technique'

[65] Natan Jacobson, Yen Lin Lee, Vijay Mahadevan, Nuno Vasconcelos, Truong Q. Nguyen, 'Motion Vector Redinement for FRUC using saliency and segmentation', ECE Department, University of California, San Diego, La Jolla, CA 920930407

[66] David Hargreaves, B.a.sc., 'Interpolation and motion compensation of Interlaced video', Simon Fraser University, 1992

[67] J. Cohn, A.J. Zlochower, J.J. Lien, T. Kanade, "Feature-point tracking by optical flow discriminates subtle differences in facial expression", in: Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998, pp. 396–401.

[68] G.D. Kearney, S. McKenzie, "Machine interpretation of emotion: design of a memory-based expert system for interpreting facial expressions in terms of signalled emotions (JANUS)", Cognitive Science 17 (4) (1993) 589–622.

[69] M. Kato, I. So, Y. Hishinuma, O. Nakamura, T. Minami, "Description and synthesis of facial expressions based on isodensity maps", in: T.L. Kunii (Ed.), Visual Computing, Springer, Tokyo, 1991, pp. 39–56.

[70] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features," Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on , vol.1, no., pp.I-511,I-518 vol.1, 2001 doi: 10.1109/CVPR.2001.990517

[71] http://sepwww.stanford.edu/data/media/public/sep// morgan/texturematch/paper_html/node3.html, access: 01.06.2015

[72] Steven Tanimoto (University of Washington), The Elements of Artificial Intelligence, http://www.ru.lv/ peter/zinatne/buki/b02.pdf, access 02.06.2015

[73] International Commission on Illumination http://www.cie.co.at/, access 15.05.2015

[74] M. Fowler, UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition) 3rd Edition

[75] https://upload.wikimedia.org/wikipedia/commons/thumb/c/c9/Client-server-model.svg/2000px-Client-server-model.svg.png access: 11.06.2015

[76] OpenCv Documentation http://docs.opencv.org/index.html, access 01.03-1.07.2015

[77] Amazon Web Services http://aws.amazon.com, access 15.05.2015

[78] Jetty documentation http://www.eclipse.org/jetty/ documentation/current/index.html, access 10.06.2015

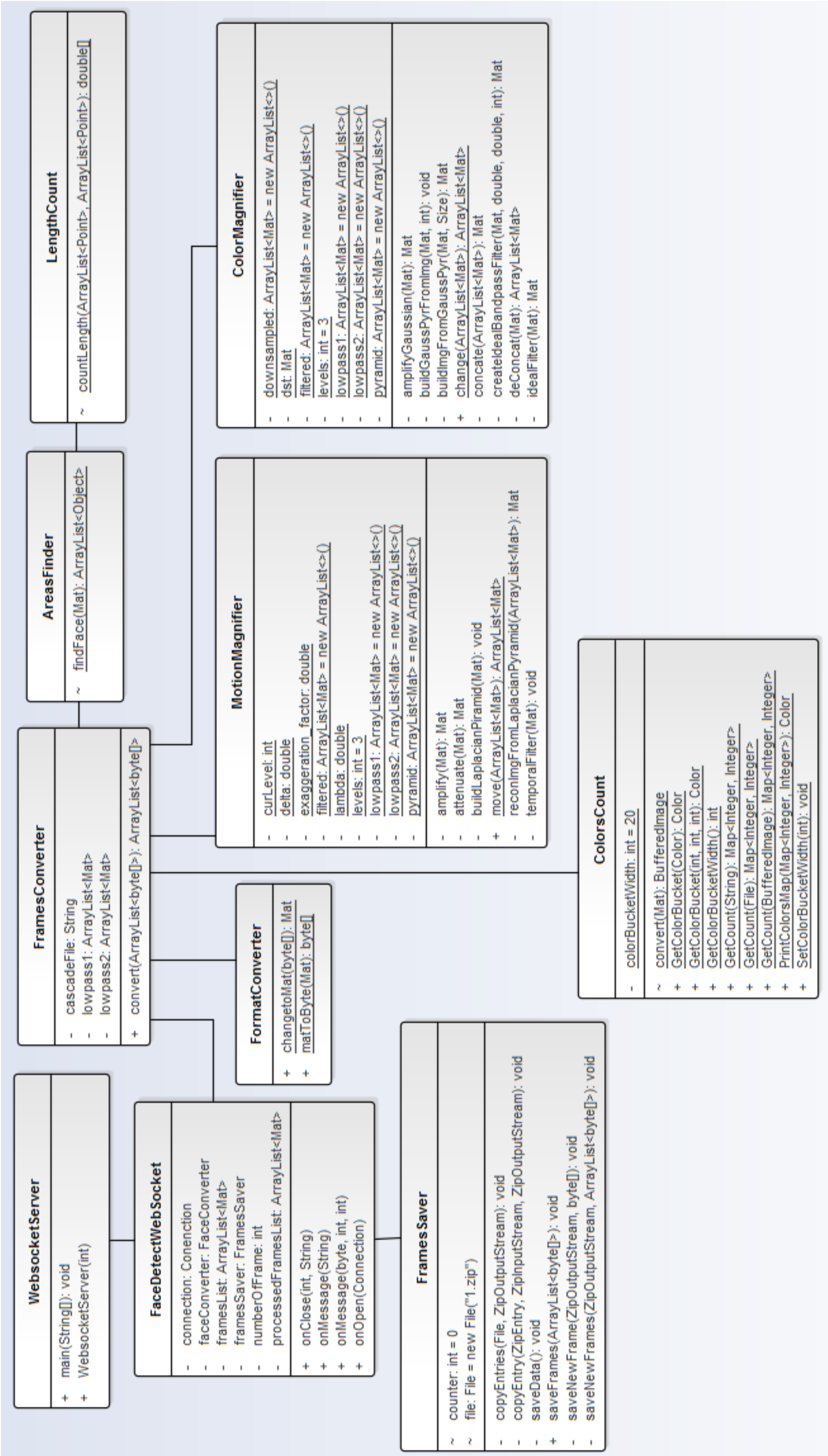
[79] Jetty Web Server documentation http://api.dpml.net/org/mortbay/jetty/6.1.3/overview-summary.html, access 10.06.2015

[80] Eulerian Video Magnification for Revealing Subtle Changes in the World - Matlab sources http://people.csail.mit.edu/mrub/evm/#code, access: 15.05.2015

[81] A.Kwasniewska, P.Otulak, E.Graban SPECIALIZATION PROJECT "Gamification in learning and training", Trondheim, December 11, 2014

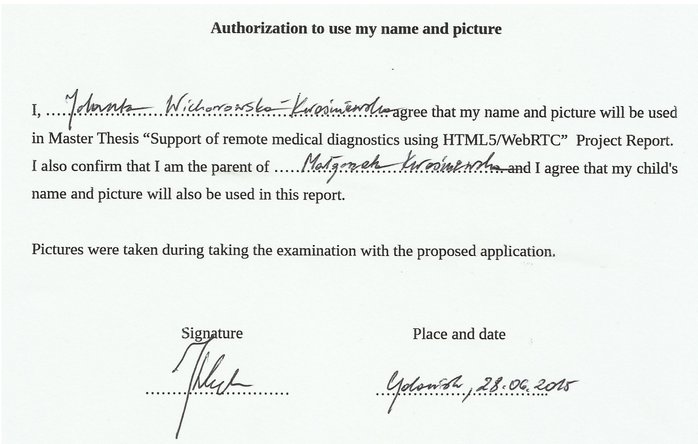
Appendix A

Class diagram


LengthCount
~ countLength(ArrayList<Point>, ArrayList<Point>): double[]
AreasFinder
~ findFace(Mat): ArrayList<Object>
ColorMagnifier
- downsampled: ArrayList<Mat> = new ArrayList<>()
- dst: Mat
- filtered: ArrayList<Mat> = new ArrayList<>()
- levels: int = 3
- lowpass1: ArrayList<Mat> = new ArrayList<>()
- lowpass2: ArrayList<Mat> = new ArrayList<>()
- pyramid: ArrayList<Mat> = new ArrayList<>()
- amplifyGaussian(Mat): Mat
- buildGaussPyrFromImg(Mat, int): void
- buildImgFromGaussPyr(Mat, Size): Mat
+ change(ArrayList<Mat>): ArrayList<Mat>
- concate(ArrayList<Mat>): Mat
- createIdealBandpassFilter(Mat, double, double, int): Mat
- deConcat(Mat): ArrayList<Mat>
- idealFilter(Mat): Mat
FramesConverter
- cascadeFile: String
- lowpass1: ArrayList<Mat>
- lowpass2: ArrayList<Mat>
+ convert(ArrayList<byte[]>): ArrayList<byte[]>
MotionMagnifier
- curLevel: int
- delta: double
- exaggeration_factor: double
- filtered: ArrayList<Mat> = new ArrayList<>()
- lambda: double
- levels: int = 3
- lowpass1: ArrayList<Mat> = new ArrayList<>()
- lowpass2: ArrayList<Mat> = new ArrayList<>()
- pyramid: ArrayList<Mat> = new ArrayList<>()
- amplify(Mat): Mat
- attenuate(Mat): Mat
- buildLaplacianPiramid(Mat): void
+ move(ArrayList<Mat>): ArrayList<Mat>
- reconImgFromLaplacianPyramid(ArrayList<Mat>): Mat
- temporalFilter(Mat): void
WebsocketServer
+ main(String[]): void
+ WebsocketServer(int)
FormatConverter
+ changetoMat(byte[]): Mat
+ matToByte(Mat): byte[]
ColorsCount
- colorBucketWidth: int = 20
~ convert(Mat): BufferedImage
+ GetColorBucket(Color): Color
+ GetColorBucket(int, int, int): Color
+ GetColorBucketWidth(): int
+ GetCount(String): Map<Integer, Integer>
+ GetCount(File): Map<Integer, Integer>
+ GetCount(BufferedImage): Map<Integer, Integer>
+ PrintColorsMap(Map<Integer, Integer>): Color
+ SetColorBucketWidth(int): void
FaceDetectWebSocket
- connection: Conenction
- faceConverter: FaceConverter
- framesList: ArrayList<Mat>
- framesSaver: FramesSaver
- numberOfFrame: int
- processedFramesList: ArrayList<Mat>
+ onClose(int, String)
+ onMessage(String)
+ onMessage(byte, int, int)
+ onOpen(Connection)
FramesSaver
~ counter: int = 0
~ file: File = new File("1.zip")
- copyEntries(File, ZipOutputStream): void
- copyEntry(ZipEntry, ZipInputStream, ZipOutputStream): void
- saveData(): void
+ saveFrames(ArrayList<byte[]>): void
- saveNewFrame(ZipOutputStream, byte[]): void
- saveNewFrames(ZipOutputStream, ArrayList<byte[]>): void

Appendix B

Authorizations



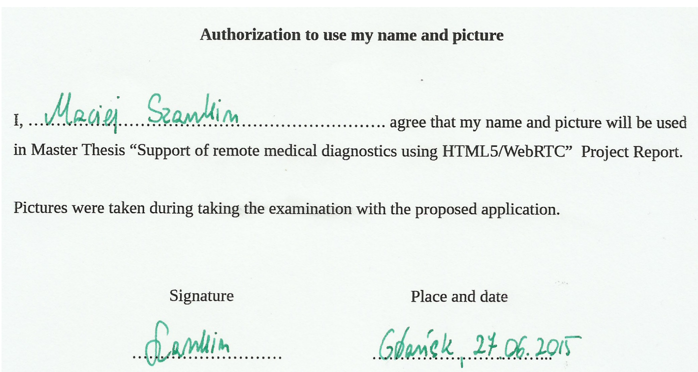
Authorization to use my name and picture

I, ...Jolanta Wichorowska-Krasiewska... agree that my name and picture will be used in Master Thesis “Support of remote medical diagnostics using HTML5/WebRTC” Project Report. I also confirm that I am the parent of ...Małgorzata Krasiewska... and I agree that my child's name and picture will also be used in this report.

Pictures were taken during taking the examination with the proposed application.

Signature ....

Place and date ...Gdańsk, 28.06.2015...



Authorization to use my name and picture

I, ...Maciej Szankin... agree that my name and picture will be used in Master Thesis “Support of remote medical diagnostics using HTML5/WebRTC” Project Report.

Pictures were taken during taking the examination with the proposed application.

Signature ...Szankin...

Place and date ...Gdańsk, 27.06.2015...

Authorization to use my name and picture

I, Mariana Kwaśniewska……………… agree that my name and picture will be used in Master Thesis "Support of remote medical diagnostics using HTML5/WebRTC" Project Report.

Pictures were taken during taking the examination with the proposed application.

Signature: [illegible]

Place and date: Gdańsk, 28.06.2015

Authorization to use my name and picture

I, EMILIA GRABAN……………… agree that my name and picture will be used in Master Thesis "Support of remote medical diagnostics using HTML5/WebRTC" Project Report. I also confirm that I am the parent of …………………………………… and I agree that my child's name and picture will also be used in this report.

Pictures were taken during taking the examination with the proposed application.

Signature: Emilia Graban

Place and date: TRONDHEIM, 12.06.2015

Appendix C

Users manual

1. Open a website specified by a given url.
2. Register a new account if you are not registered by accepting terms of use and providing username, password and e-mail address (not available in a prototype).
3. Log in if you already have an account (not available in a prototype).
4. Start the examination by allowing a website to access a webcam.
5. Sit still in front of the camera, if you want to save specific gesture frames, make this gesture and push the appropriate button (**smile, sad or e sound**). Suggested conditions include daylight, scanning distance equals 30 centimeters, no eyeglasses.
6. You can finish the examination at any time you want by closing the website (you will be automatically logged out).

Appendix D

Running the program

The program may be run by double clicking on the jar file added to a Master Thesis archive. On the Windows the appropriate OpenCV dll library should be placed in the same directory as the jar file. On the Linux the appropriate OpenCV so library should be placed in a directory pointed by the java library path.