Anexa 3

"Politehnica" University of Bucharest

The Faculty of Electronics, Telecommunications and Information Technology

FACIAL ANALYSIS METHOD FOR RECOGNIZING THE EXPRESION OF SURPRISE

Bachelor's Degree project

presented as a partial requirement for obtaining the title of

Engineer in the field of *Electronic Engineering and Telecommunications*

License Study Program *Applied Electronics*

Scientific leader

Conf.Dr.Ing. Laura-Maria Florea

Graduate

Marius-Cristian Poparascu

2019

University "Politehnica" of Bucharest
Faculty of Electronics, Telecommunications and Information Technology
Department **EAII**

Anexa 1

DIPLOMA THESIS

of student **POPĂRĂSCU C.M. Marius-Cristian , 441F**

1. Thesis title: Facial analysis method for recognizing the expression of surprise.

2. The student's original contribution will consist of (not including the documentation part) and design specifications:
A method for facial analisys based on Local Binary Patterns (LBP) and Random Forests (RF) will be implemented in order to detect the expression of surprise on human faces. The method will start with localizing the face and the most important points on the face. In the areas of those points, LBP features will be computed. The features will then be fed to a RF classifier. The method will be tested on a public database annotated with different expressions. The implementation will be done in Python using OpenCV and DLIB libraries.

3. Pre-existent materials and resources used for the project's development:
Python, OpenCV and DLIB libraries, Cohn-Kanade database

4. The project is based on knowledge mainly from the following 3-4 courses:
Data Bases,Computer Programming, Data Structures and Algorithms

5. The Intellectual Property upon the project belongs to: U.P.B.

6. Thesis registration date: 2018-11-28 21:28:04

Thesis advisor(s),
Conf. dr. ing. Laura Maria FLOREA
signature: ....

Student,
signature: ....

Departament director,
Prof. dr. ing Sever PAŞCA
signature: ....

Dean,
Prof. dr. ing. Cristian NEGRESCU
signature: ....

Validation code: **5e5873206c**

Statement of originality

DECLARAȚIE DE ORIGINALITATE A PROIECTULUI DE DIPLOMĂ/LUCRĂRII DE DISERTAȚIE[1)]

Subsemnatul Poparascu Marius-Cristian[2)], posesor al B.I./C.I./pașaport seria [redacted], având CNP [redacted],

am întocmit proiectul de absolvire/diplomă/lucrarea de disertație cu tema:[3)] Metodă de analiză facială pentru recunoașterea expresiei de surpriză,

în vederea susținerii examenului de finalizare a studiilor universitare de licență/masterat, organizat de către Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Departamentul ETTI, din cadrul Universității POLITEHNICA din București, sesiunea Iulie, anul universitar 2018-2019.

Luând în considerare conținutul art. 143 din Legea Educației Naționale nr. 1/2011, al. (4) „*Îndrumătorii proiectelor de diplomă/lucrărilor de disertație răspund în solidar cu autorii acestora de asigurarea originalității conținutului acestora*" și al. (5) „*Este interzisă comercializarea de lucrări științifice în vederea facilitării falsificării de către cumpărător a calității de autor al unui/unei proiect de diplomă/lucrări de disertație*", declar pe propria răspundere, că acest/această proiect/lucrare este original(ă), fiind rezultatul propriei activități intelectuale, nu conține porțiuni plagiate, iar sursele bibliografice au fost folosite cu respectarea legislației în vigoare.

Cunosc faptul că plagiatul sau prezentarea unui/unei proiect/lucrări, elaborat(ă) de alt absolvent sau preluată de pe internet, din manuale și cărți, fără precizarea sursei constituie infracțiune (furt intelectual și nerespectarea dreptului de autor și a proprietății intelectuale) și atrage după sine anularea examenului de diplomă/disertație, precum și răspunderea penală.

Data: 19.06.2019

Semnătura: ____________
(în original)

[1)] Declarația se completează „*de mână*" și reprezintă un document care se depune la înscrierea pentru susținerea examenului de finalizare a studiilor. Ea nu se include în proiect – vezi Anexa 5 în acest scop.

[2)] Numele, inițiala prenumelui tatălui și prenumele, cu majuscule.

[3)] Denumirea proiectului de diplomă/lucrării de disertație, cu majuscule.

Table of Content

List of figures

List of tables

Achronime list

AI	Artificial intelligence
AU	Action Units
AVRS	Audio Video Recovery Systems
CAPS	Chinese Affective Affective System
FACS	Facial Action Coding System
HCI	Human-Computer Interaction
IAPS	International affective image system
LBP	Local Binary Patterns
MAUI	Multimodal Affective User Interface
ML	Machine Learning
SAM	Self-evaluation dummy
RF	Random Forest
VR	Virtual reality

Introduction

Machine Learning (ML) offers systems the ability to learn and improve experience without being explicitly programmed, accessing data and learning from their own experience.

Artificial intelligence (AI) refers to systems or machines that mimic human intelligence, to perform various activities and can improve iterative based on the information they collect.Thus, the system can take decisions with minimal human interactions.

The motivation: The emotions are important because, they reflect the status of a subject and they allow you to analyze it's reaction to certain stimuli. Emotions constantly characterizes us and detecting the types of emotion gives us the opportunity to observe the condition of others.

The Possible applications of my license project would be : Implementing the software in offices so that, the employer can detect the reaction of the employee when being at work or when receiving a task thus by analyzing it's reactions , measures can be taken to improve the working atmosphere.

Another suitable application would be to implement the software in marketing so that the employer can observe the reaction of a customer when looking at a product or when looking at a price tag.

It could be used in the automation domain so that, for example :a car, can read the emotion of the driver and , when the driver is angry, to make the acceleration and the brake harder, to avoid the sudden acceleration or braking of the driver, to prevent a disaster.

The Purpose of the license work is to correctly identify the emotion of surprise of a person based on their facial expression mimics.

Contribution: The work implies the detection of the emotion of surprise in a photo taken from the video camera. This method implies the following mindset : Detect the face and the facial landmarks, crop the face, resize the face image, make the face image gray, split the face image in 49 cells, apply a feature extractor to get the details from the face image, create the histogram for the feature extractor, concatenate all the 49 cells for that we have extracted the features to remake the original photo, train the classifier and test the classifier in order to get a prediction. It is intended to implement a high accuracy recognition algorithm of surprise emotion based on facial analysis algorithms using machine learning and AI.

I have tested the algoritm for 78 subject, the conclusion being that, it scored 90.04% Accuracy

Working on the License project , I have learned the concept of machine learning and artificial intelligence algorithms, how to work with them and I have improved my programming skills.

Facial detection and recognition

Facial detection is a technology used in a variety of applications used to identify faces in digital images or frames from a video camera.Facial detection can be seen as a particular case of object detection. Facial detection algorithms focus on images with faces that look forward, because those are more easily detectable.

What I am doing in my license work is only a part of the facial detection thus,my algorithm can be implemented further with the detection of all the emotions and in the last,the recognition of the faces.

Facial recognition systems are applications capable of identifying and verifying the identity of a person in a digital image or video frames.The facial recognition method used in this work is to compare certain facial features between a test image and a picture stored in a data base.

In chapter 1 will be made a theoretical presentation of the concept and theoretical perspectives of emotions.

In Chapter 2 the Local Binary Pattern algorithm that is used to extract the features from an image.

In Chaper 3 the Random Forest classifier will be presented.

In Chapter 4 will be explained the steps that need to be followed in order to deploy the algorithm, the database that was used to implement the algorithm and the experimental results.

In Chapter 5 will be listed the conclusions that are resulting from the testing of the algorithm.

1.EMOTIONS AND FACIAL EXPRESSION

1.1 Introduction

Nothing more elementary and yet, nothing more complex, difficult to explain than emotion.

Emotions are specific and intense psychological reactions to a particular event. Emotions, often called feelings, include manifestations such as love, hate, anger, trust, panic, fear, pain. There are specific reactions to a particular event, usually short-lived.

The expression of basic emotions is the same regardless of the degree of civilization and the geographical position, similarities involving identical psychophysiological "mechanisms". The issue of the nature and function of emotion has not yet found a unanimously accepted solution[Leonard Gavriliu].

Emotions are complex and have different physical and mental components and they are composed of: subjective feelings, physiological responses and expressive behavior.

The first scientific attempts in the analysis of facial expressions are found in Charles Darwin's work, published in 1872, "Expression of emotions in humans and animals" . Emotions are adaptive behaviors of a stage far exceeded on the path of evolution .Thus, he is one of the first to try to formulate the theory of origin of emotions and their role during the survival cycle of the human species [Charles Darwin].

In the middle of the 20th century, the study emotion has captured the researchers attention, becoming a topic of interest and ,because it is new, we still have unresolved issues that are reflected in Human-Computer Interaction (HCI).

HCI starts from the specifications of the theories from psychology, which refer to emotions, in the desire to design intelligent emotional systems. It seeks consistency and scalability in definition, induction and description of emotions, regardless of race, gender or age.

Human-Computer Interaction community is showing increasing interest in the integration of affective computing in their technology. Particular attention is being paid to research on emotion recognition, since computer systems should be able to recognize human emotions in order to interact with humans in a more adaptive and natural, human-centered way [Dolores Cañamero].

The fundamental question is to design a system for detecting and analyzing emotions, an intelligent system that uses unitary emotional description models. When describing and designing an emotional system, we must keep in mind the context in which it is used. Emotional states are subjective, and different people react differently in identical situations.

Reaction is influenced by the personality of each person and the context in which comes from. It is hard to induce the same emotional state, with the same intensity, to two different people. All these aspects need to be considered together alongside with the goal of the final application.

1.2 The theory of emotions

1.2.1 Theoretical perspectives

What is an emotion?

Are the emotions inherited or learned?

Theare are different perceptions about and defining and interpreting the emotion concept.

Zajonc [Zajonc] believes that the perception of a new stimulus or threatening may occur immediately without involving a conscious or explicit processing and, evaluation of stimulus may occur after emotional response was activated. In a paradigm contrary, Lazarus [Lazarus] argues that the assessment of a situation resulting in emotional and cognitive response to the emotion temporarily prevails. Thus, we can say that there is some validity to both theories.

Cole [Cole PM] defines emotions as dynamic ,powerful and elusive processes, which have the ability to adjust other processes.

Four major theoretical perspectives on emotion are described in psychology. Examples of the ways where research on emotion and speech use aspects of perspectives are presented .

Thus, over the years,we attempted to define emotions, but none of the proposed options is not generally valid. To understand the concept, I listed below the four approaches that summarizes all efforts made in the literature to define the emotional reaction.

The Darwinian Perspective

The central idea of the Darwinian point of view is that the emotions are provided in the human DNA .The notion of "emotion" is a evolved phenomena with crucial survival functions that have been choosen for because they have resolved certain problems that we have faced as species. By its very nature, we should see the same emotions, more or less, in all human beings. In addition, considering that humans share an evolutionary past with other vertebrate, we shall expect to see similarities in the emotions of closely-related species.

The Darwinian mindset and its associated culture of research had their inception in [Charles Darwin].

Here is an explanation of emotions, from the point of view of the impulse of action [Goleman]:

•**Anger** causes an increased flow of blood through the arms and hands
•**Fear** causes blood flow through targeting large skeletal muscles.
•**Happiness** makes one of the main biological changes , it consists in an increase in brain activity center that inhibits negative feelings and houses and boosts energy;
•**Love**, tenderness and sexual satisfaction generates excitement parasympathetic, which induces a general state of calm and contentment, which facilitates cooperation;
•**Surprise** allows you to retrieve a stream of visual information,leading to a greater knowledge;
•**Disgust** is something that is repulsive in taste or smell;
•**Sadness** generates a drop in energy and enthusiasm for daily activities, and induces a state of depression reflected in metabolism;

This model is opposed to the constructivist one and claims that this model description of emotions is universal, regardless of background, social class or cultural background of the individ.

The Jamesian Perspective

Hypothesis and research in the Jamesian theory was motivated by William James on emotions, specifically, his 1884 article [William James].

Almost since the day of its production, clinicians of different sorts have discussed the truth of James renowned condition of feelings with the view of real changes: emotions appear like as a result of changes in human metabolism. Thus, an organism incapable of sensation,perceptions or changes, will also be deprived of emotion.

James demanded that it would be difficult to have feelings without substantial changes. The design of an emotional analysis and detection system based on this model will require intrusive devices, such as sensors attached to the skin, to measure individual physiological activity.

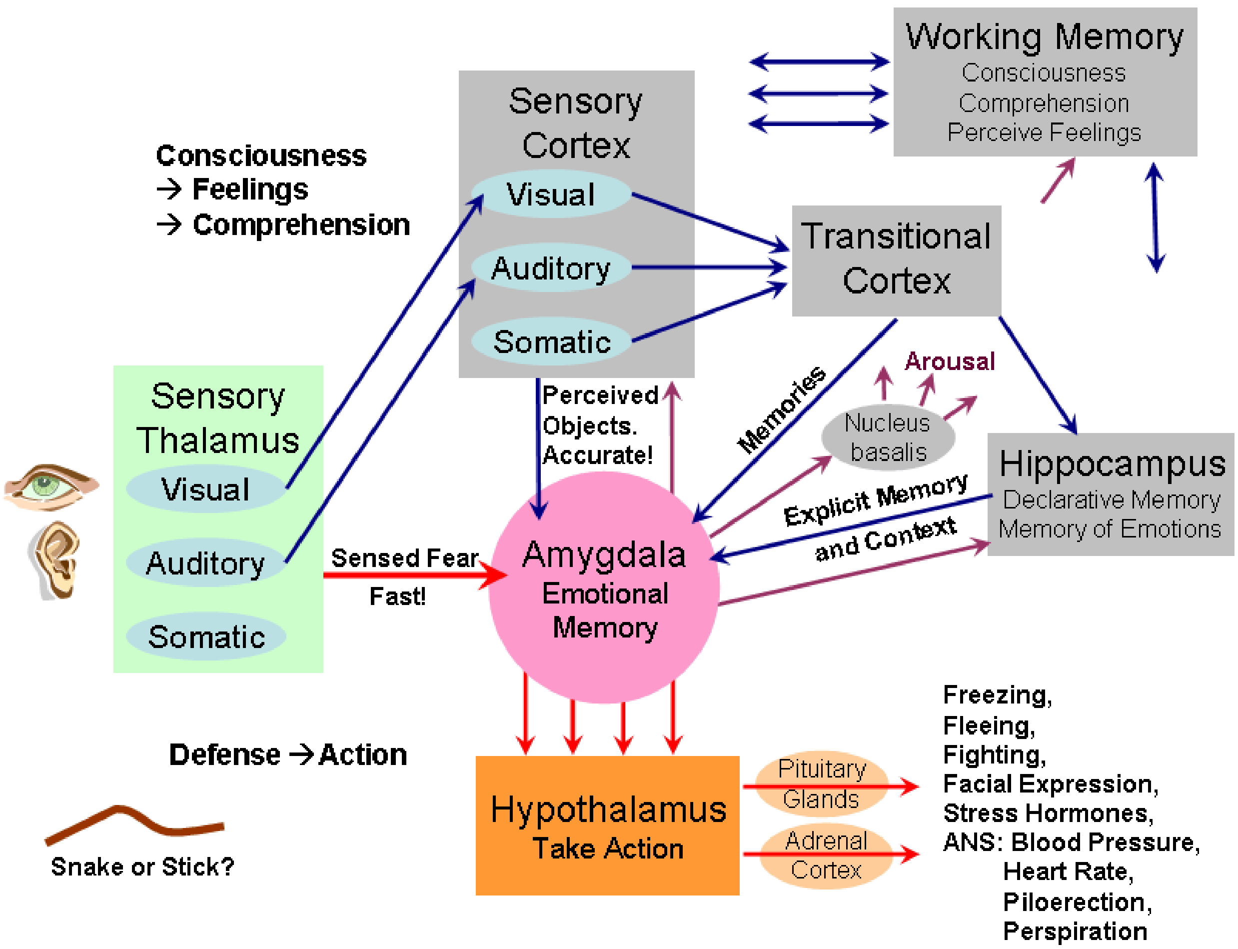


Figure 1.1: The paths of the emotions take through our brain[Damasio]

The Cognitive Perspective

The Cognitive psychology says that emotions come from a cognitive process directly. The central assumption of the cognitive perspective and its associated tradition of research is that thought and emotion are inseparable. To analyze and classify emotions, it is necessary to focus on the brain structure. Following the analysis of the brain signals,we can trigger the emotional processes and their mechanisms.

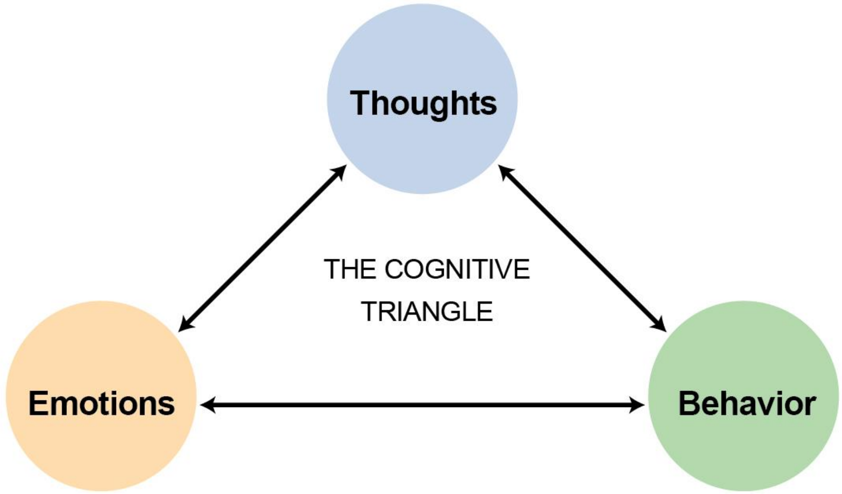


Figure 1.2: Cognitive Triangle

More respectively, all emotions are observed within this perspective as being reliant on what Arnold [Arnold] called appraisal: the mechanism by which events in the environment are distinguished as good or bad for us.

The Social Constructivist Perspective

The most diverse, the youngest and certainly, the most controversial of the four theoretical perspectives is the social constructivist.

From a social perspective, emotions are described as resulting from social interactions, taking into account the cultural rules related [Cornelius]. Although this definition is quite tough, the researchers have concluded that social and cultural rules shape the emotional state of each individual.

Thus, this theory emphasizes the importance of emotions in the social framework and suggest that an emotionally intelligent system must be designed according to the cultural and social context in which it will be implemented.

1.2.2 Emotion Models

The theory of emotions says that no relationship can be defined without a logical frame. Any possible disharmony in the description of experiences can be ignored only by an appropriate widening of the conceptual framework.This theory has been imposed by the development of physics in a way that has a bearing on many other fields of human knowledge and sympathy in which we meet with related position in the analysis and synthesis of experience.

There are a diversity of reasons for the confusion about the nature of emotions and the analysis of what a definition of emotion requires is often faulty.

For the recognition of emotions, the emotions must be defined and accessed quantitatively. The definition of basic emotions was first proposed a few decades ago and the psychologists tend to model emotions in two different ways. One is to divide emotions into discrete categories. The other is to use multiple dimensions for emotion labeling. To excite emotion, subjects receive a series of emotional emotional materials to induce some emotion. The pictures, music and films stimulations are the most used materials. Computer games or reminiscence are used to generate emotion. Among these methods, affective Virtual Reality has captured more and more attentions.

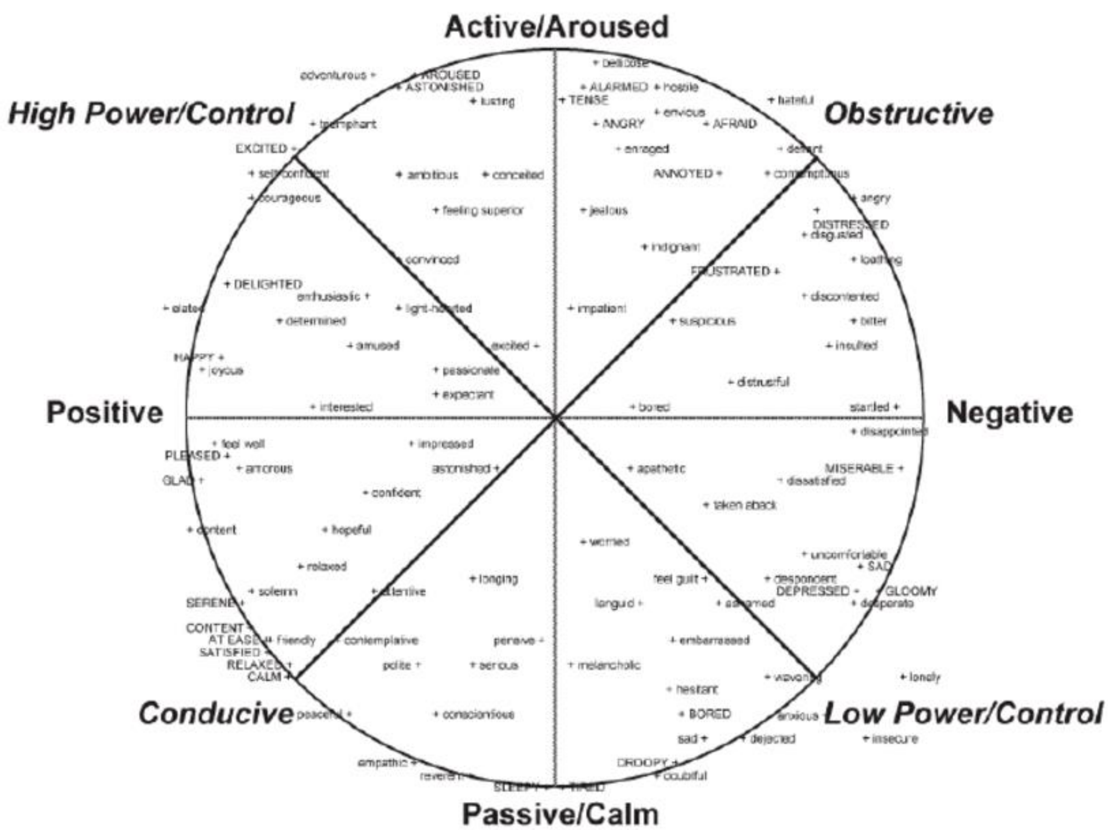


Figure 1.3: Emotion Wheel[Scherer]

1.2.2.1 Discrete Emotion Models

Ekman [Ekman P.] believes that emotions are discrete, measurable and physiologically related. He proposed a series of characteristics regarding the basic emotions:

-People are born with emotions that are not learned; People have the same emotions in the same situation;

-People express these emotions in a similar way;

-People have similar physical models when expressing the same movements.

According to his characteristics, he summed up six fundamental emotions : anger, happiness, surprise, disgust, sadness, and fear regarded other emotions as producing reactions and combinations of these emotions.

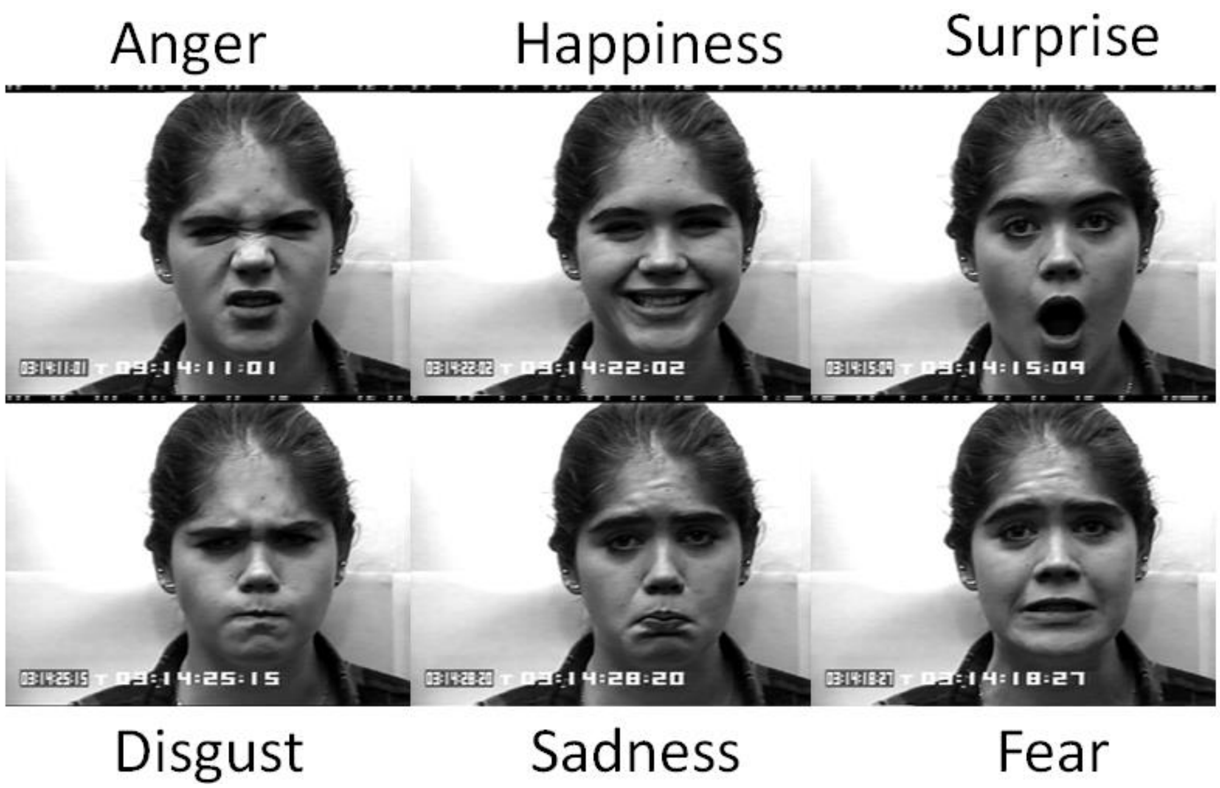


Figure 1.4: Fundamental emotions

In 1980, Plutchik [Plutchik R.] proposed a wheel model that includes eight fundamental emotions : joy, confidence, fear, surprise, sadness, disgust, anger and anticipation, as shown in Figure 1.5.

This emotional model describes emotions according to intensity. Stronger emotions are in the center, while weaker emotions are at the flower blooms. Just like color, basic emotions can be mixed to form complex emotions. They describe that: Basic emotions have formed in the course of human evolution;

Every basic emotion corresponds to a simple circuit and there is no complex cognitive component involved. Then he presented the basic emotions: interest, joy, surprise, sadness, fear, shyness, guilt, anger,disgust and contempt.

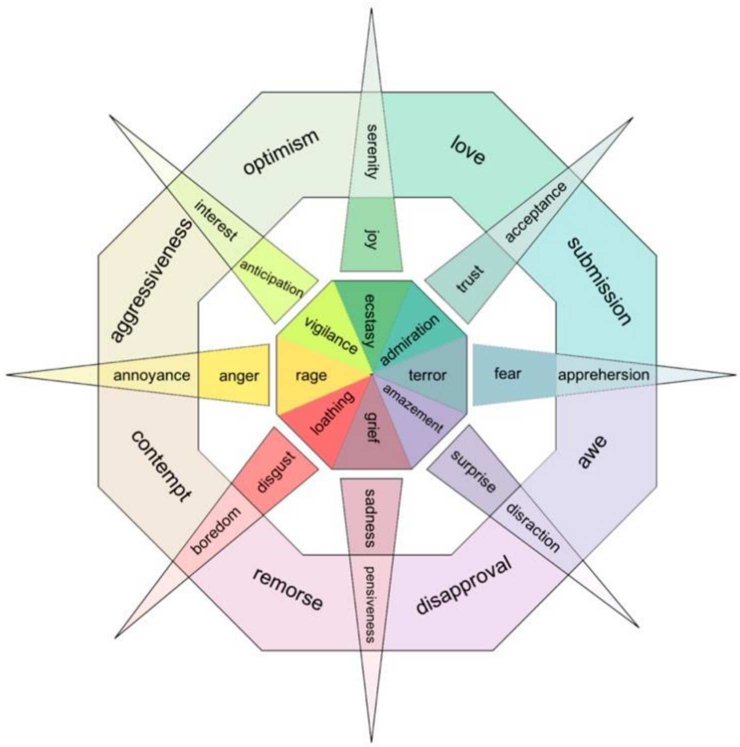


Figure 1.5: Plutchik's Wheel of Emotions[S.Kozielski].

Discrete emotional models used word descriptions for emotions, instead of quantitative analysis. It is therefore difficult to analyze complex emotions, such as mixed emotions that are difficult to express in words and need to be studied quantitatively.

1.2.2.2 ***Multi-Dimensional Emotion Space Model***

With the deepening of the research, psychologists have found that there is a certain correlation between different emotions, such as hatred and hate, pleasure and liking, which represented a certain level of specific emotional level. On the other hand, emotions that have the same descriptions might have different intensities. For example, happy could be described as a little bit happy or very happy. Therefore, psychologists have tried to build multidimensional models of emotional space. Lang [Lang P.J.] investigated that emotions can be classified into a 2D space through valence and arousal. In his theory, valence varies from unpleasant (negative) to pleasant (positive), and excitation ranges from passive (low) to active (high), which indicates how powerful mankind is. Different effects can be represented in 2D space. For example, anger has a negative valence and low arousal.

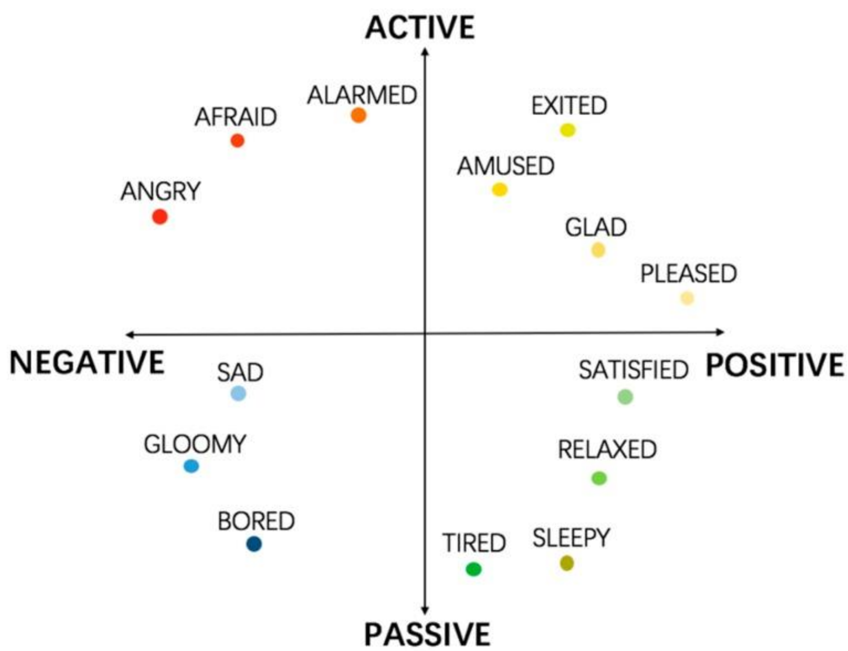


Figure 1.6: 2D emotion space model[Yangzhou Du].

Mehrabian [Mehrabian A.] considered that the 2D model was limited and he extended the emotion model from 2D to 3D (Figure 1.7).The dimensional axis added is called dominance, ranging from submissive to dominant, which reflects the ability of controlling man in a certain emotion. In this dimension, anger and fear can easily be identified as raging on the dominant axis, while fear is on the underlying axis.

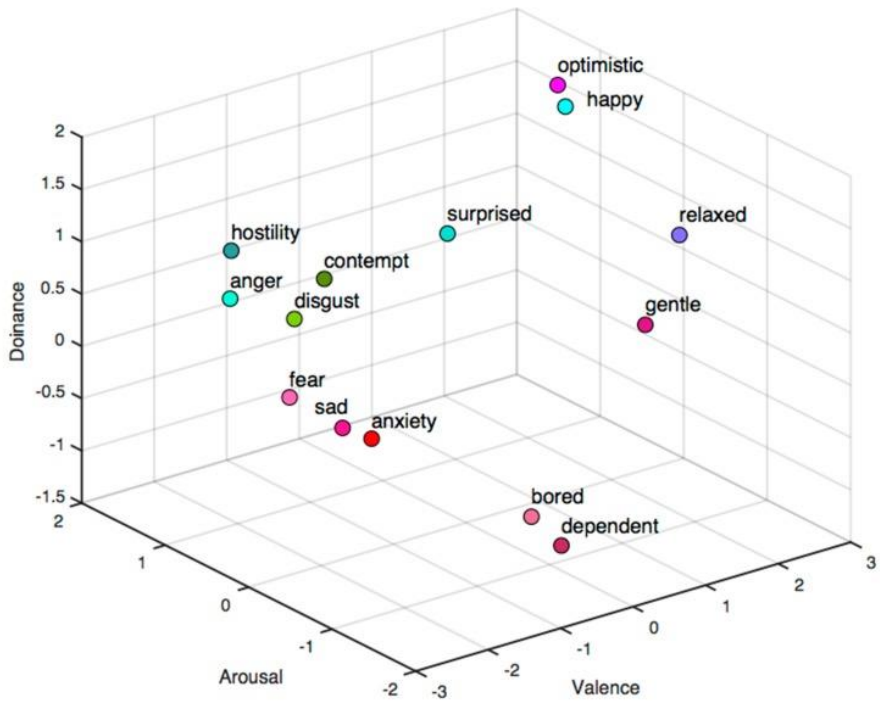


Figure 1.7: 3D emotion space model[C. Breazeal].

1.2.2.3 Emotion Stimulation Tools

The National Institute of Mental Health [Lang P.J.] proposed in 1997 the well-known international affective image system (IAPS), which provided a series of standardized, emotionally-evocative photos that can be accessed by everyone. In addition, in 2005, the Chinese Affective Affective System (CAPS) was proposed [Bai L], an important tool for internal researchers.

Combining visual and auditory senses, film stimulation has much progress. In the beginning [Nasoz F.], the authors built a multimodal affective user interface (Figure 1.8a) to help gather emotion-related data of their users and emotions. After studying a film-maker pilot panel to determine high-quality films, the authors finally chose 21 videos to anger, sadness, fun, disgust, fear and surprise. The name of the film was given in this work. The music video also plays an important role in stimulating emotions.

Zhang proposed a new emotional evocation system called the Virtual Affective Reality System (AVRS, Figure 1.8b), which was composed of eight emotional VR scenarios (Figure

1.8c) and the three-dimensional emotional index that was evaluated by 100 subjects using the self-evaluation dummy (SAM). Colors, sound, and other features have been extracted to create affective VR scenes.

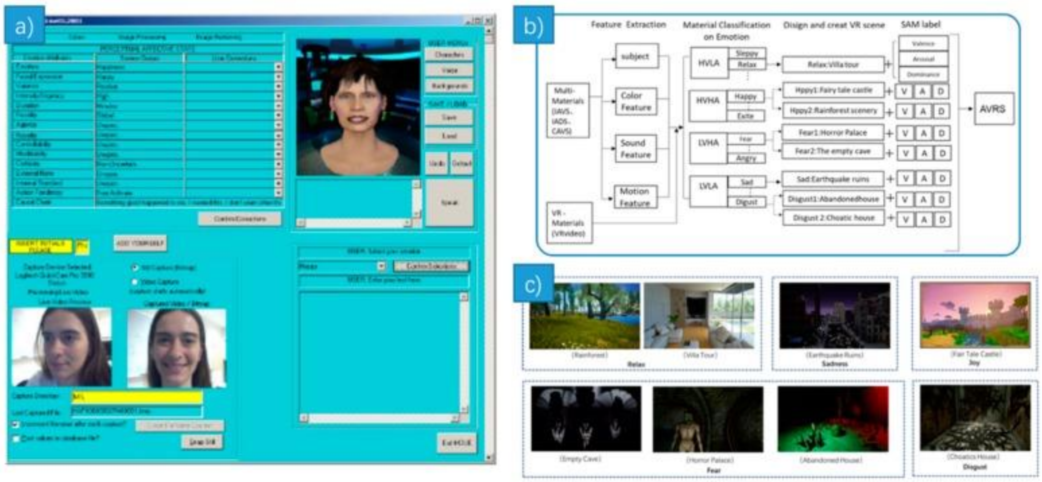


Figure 1.8:(**a**) MAUI—Multimodal Affective User Interface; (**b**) Framework of AVRS; (**c**) VR scenes cut show[Kaye Alvarez].

1.3 Facial expressions

Emotions are transmitted through various ways: face, voice, posture, gestures, muscle tension, skin temperature and others but the emotions are frequently manifested on the face. The face displays the emotions, while the body tells us how the person manages those emotions.

There are about 43 muscles in the face.However, the "universal muscles" (the basic ones) the ones that contribute to display the universal emotions on the face, vary insignificantly among individuals. On the other hand, "non-essential" muscles, those who are not involved in producing universal expressions, vary from one individual to another. The universal emotions that I talk about are: disgust, contempt, happiness, fear, anger, sadness and surprise. We call them universal emotions because all individuals of the human species, irrespective of race, culture, age, sex, etc., manifest them in the same way on the face.

The face transmits three types of signals: static (eg: skin color), slow (eg: permanent wrinkles) and rapid (eg eyebrow lifting). Fast signals are displayed on the face for only a few seconds or even fractions of a second [Ekman, P. and Friesen].

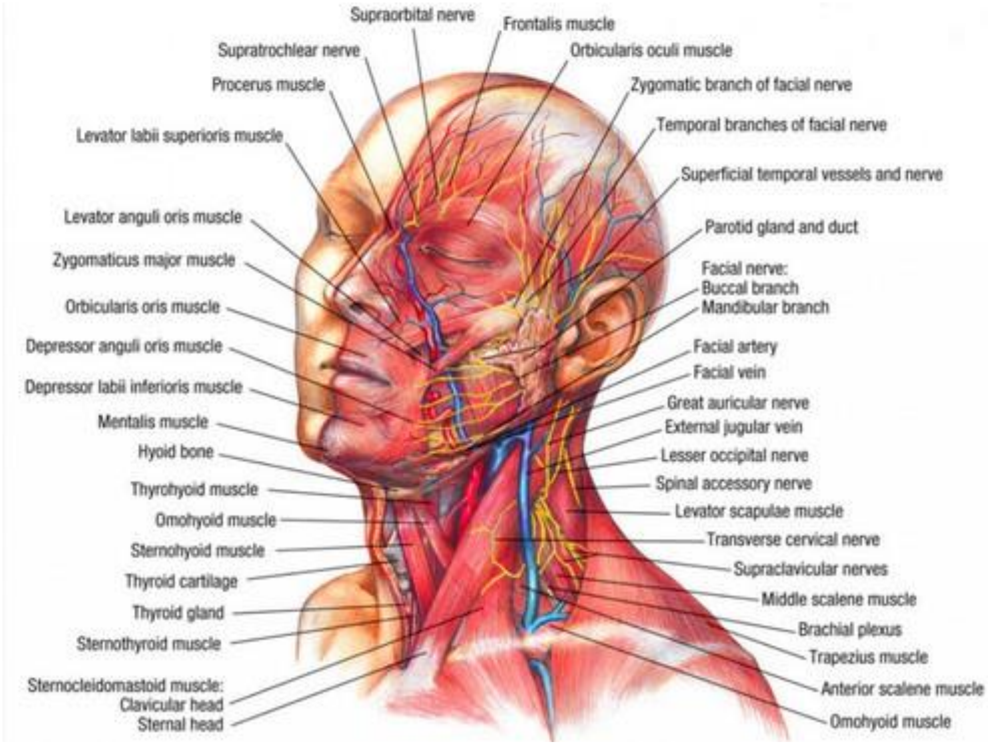


Figure 1.9: Facial muscular system[Patrick B].

1.3.1 Facial expressions measuring

Facial expressions are generated by contractions of facial muscles, which temporarily deform the face's features.

To describe and measure facial expressions, we need to define a certain terminology such as the location of facial features (eyelids, eyebrows, nose, mouth), the severity of deformation and their dynamics.

The intensity of facial expressions decreases, causing geometric distortions of facial expressions or measuring the density of wrinkles. Facial phrases are described by three temporal aspects: **onset**, the moment the expression begins ,**apex** represents the emotion at maximum intensity and **offset**, the moment of facial relaxation following the expression [Chung-Hsien Wu].

Figure 1.10: The three temporal aspects: Onset, Apex and Offset In Intensity and Time domain[C.-H. Wu, J.-C. Lin].

1.3.2 Methods based on signal interpretation

Methods based on signal interpretation (sign-vehicle) and facial deformation are encoded in visual classes. Facial actions are coded and described by their location and intensity. So a complete description frame must contain all possible perceptible changes that may appear on the human face.

1.3.3 FACS

Facial Action Coding System is a tool developed by Paul Ekman and Wallace Friesen [P. Ekman and W] for the analysis of facial muscles, each muscle, in order to identify the real emotions experienced by a person at certain times. FACS is a tool used for objective analysis of the face that describes each movement of the facial muscles in the form of **Action Units (AUs)**.

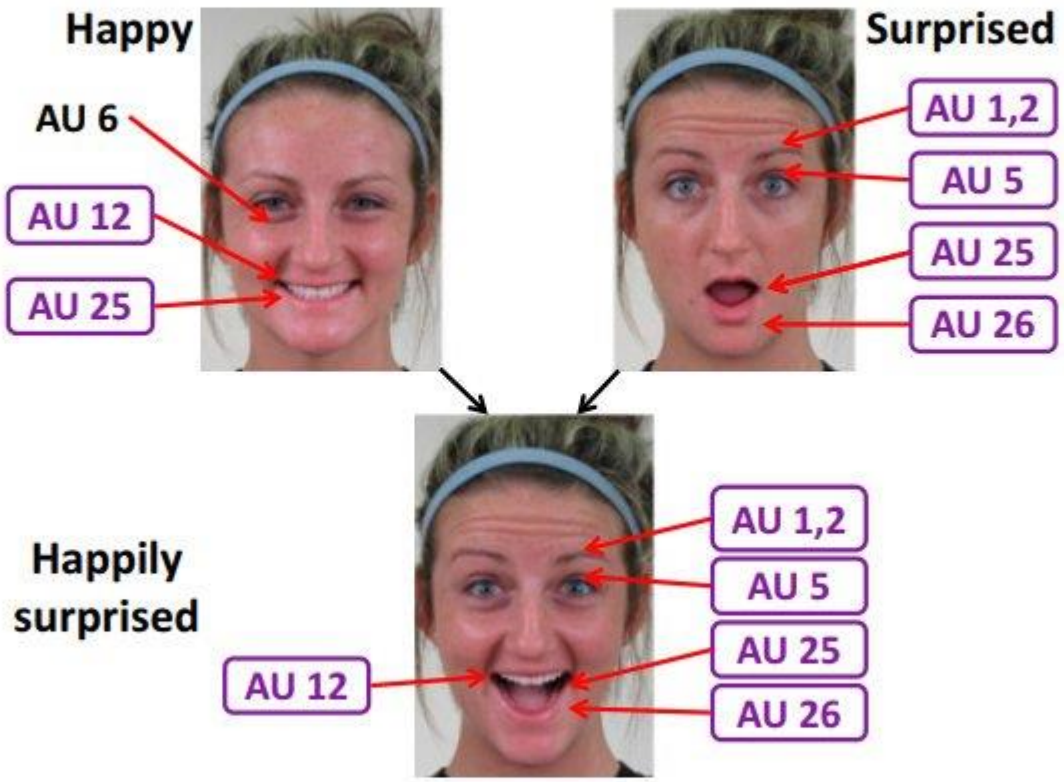


Figure 1.11: Example of decomposition of facial action according to FACS in the form of Action Units [Ekman P, Friesen WV].

Ekman and Friesen developed the FACS for the first time in 1970 [Ekman, P.], tracking the contraction of each facial muscle (alone, but also in combination with other muscles).

FACS uses 28+58 AUs that describe elemental facial movements, specifying location, intensity and magnitude [Ekman, Friesen & Hager]:

- Facial actions at the top of the face: AU 1, 2, 4, 5, 6, 7, 43, 45, 46, 70, 71;

-Head Positions : AU 51, 52, 53, 54, 55, 56, 57, 58;

-Eye Positions: AU 61, 62, 63, 64, 65, 66;

- Lip partition and jaw opening: AU 25, 26, 27;

- Facial actions at the bottom of the face: AU 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 20, 22, 23, 24, 28, 72;

- Facial Combined Actions: AU 8, 19, 21, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39;

For each universal emotion, several groups of AUs have been identified, such as: Happiness = 6 + 12, Sadness = 1 + 4 + 15, Surprise = 1 + 2 + 5B + 26, 7 + 20 + 26, Furie = 4 + 5 + 7 + 23, Disgust 9 + 15 + 16.

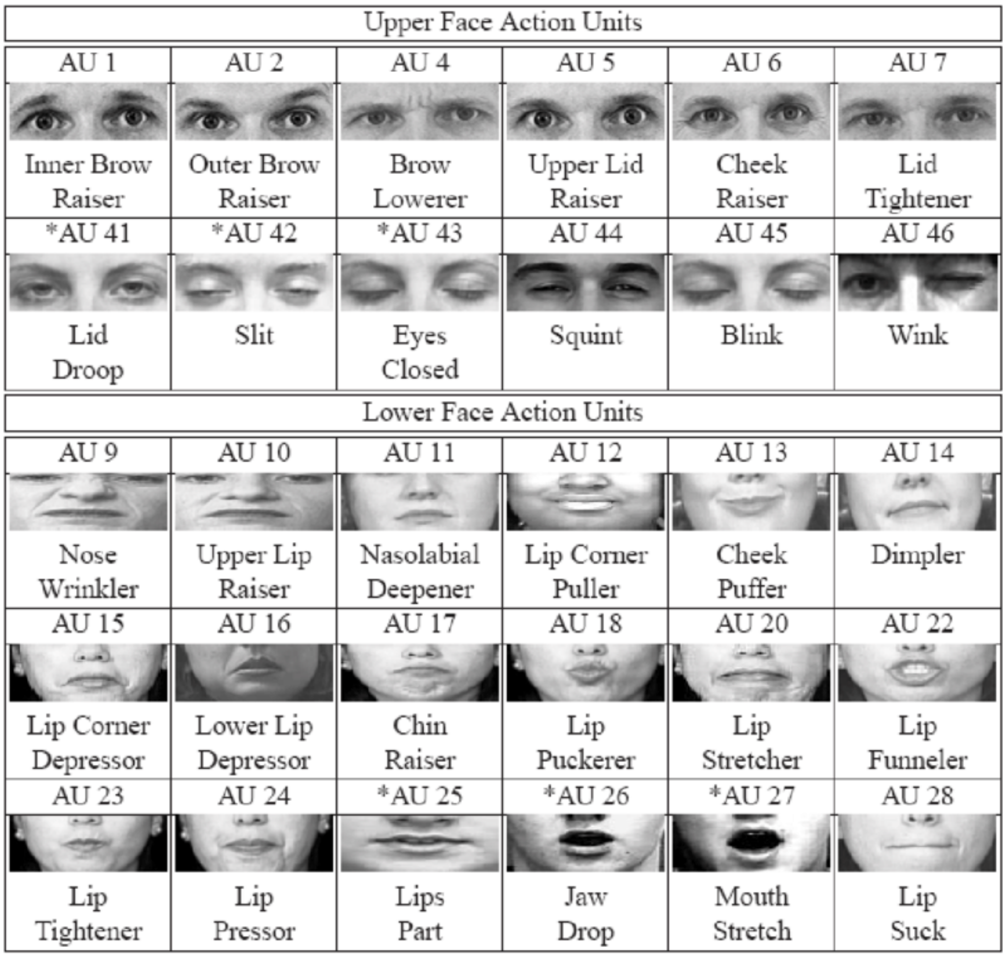


Figure 1.12: AUs from the upper and lower parts of the face [Ekman P, Friesen WV].

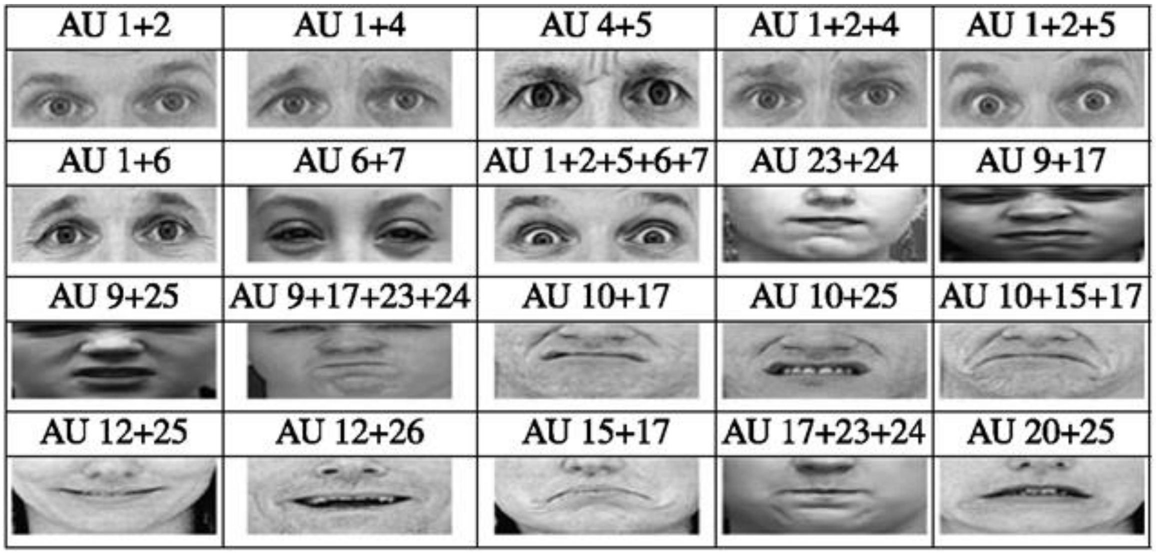


Figure 1.13: Examples of combinations of AUs [Ekman P, Friesen WV].

Facial actions can be encoded by identifying the degree of intensity and variations in the intensity of the Facial Action, and in this respect the following degrees of intensity are proposed according to the FACS [Ekman, P., Friesen, W. V., & Hager].

2. LOCAL BINARY PATTERN(LBP)

What are Local Binary Patterns?

Local Binary Patterns, or LBPs, are a texture descriptor made famous by Ojala in [T. Ojala, M. Pietikainen and T. Maenpaa] ,even the approach of LBPs were established in 1993.

LBPs compute representation of texture that is local and is build up by comparing each pixel with its surrounding neighborhood of pixels.

The first thing that needs to be done in composing the LBP texture descriptor is converting the image to grayscale. For every pixel that is in the grayscale image, we select a neighborhood of size r surrounding the center pixel.For each pixel from the input image, the LBP value is computed and is stored in an output array that has the same characteristics as the input one.

For example, let's take a look at the original LBP descriptor that operates on a 3 x 3 matrix of pixels just like this:

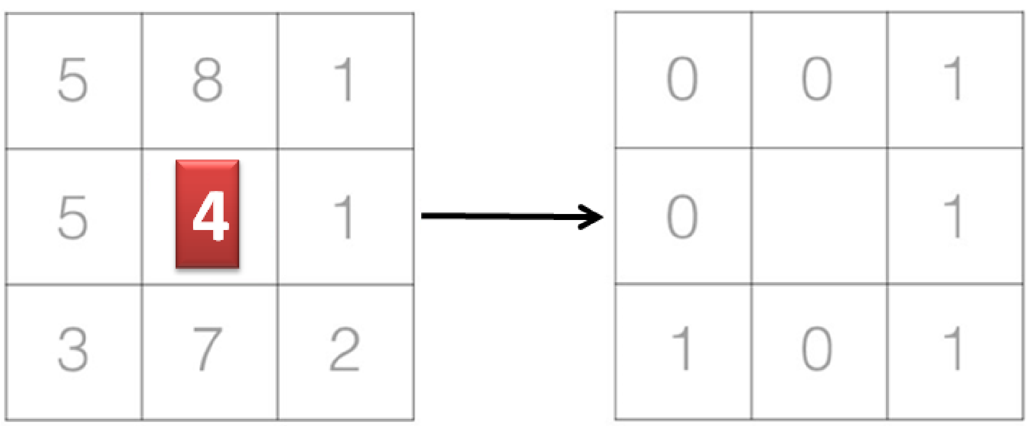


Figure 2.1: To construct the LBP,the 8 pixels that surround the center pixel have to be taken into account.

The first thing that needs to be done in constructing a LBP is to take the surrounding 8 pixels of the central one and compare each pixels's value. If the value is greater or equal to the one in cause, we write in the output array , the value 0 , otherwise, we write 1.

Once we get there, we have to compute the LBP value for the center pixel. We can start from any pixel that is in the neighborhood and work our way clockwise or counter-clockwise, but the order that we choose must be kept constant for all pixels in our image and all images in our dataset. Given a *3 x 3* neighborhood, we thus have 8 neighbors that we have to perform a binary test on. The results of this binary test are being stored in an 8-bit array, which we then convert to decimal, like in the Figure 6.1:

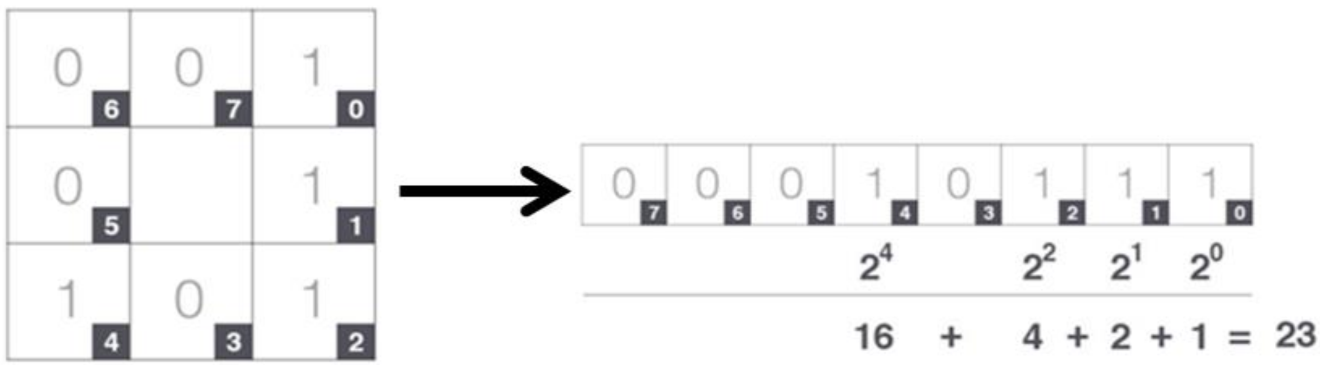


Figure 2.2: We take the 8-bit binary neighborhood of the center pixel and we convert it into a decimal representation.

This process of thresholding, taking the input pixel's value, and storing the output decimal value in the LBP array is then repeated for each pixel in the input image.

An example of visualizing and computing a full LBP 2D array:

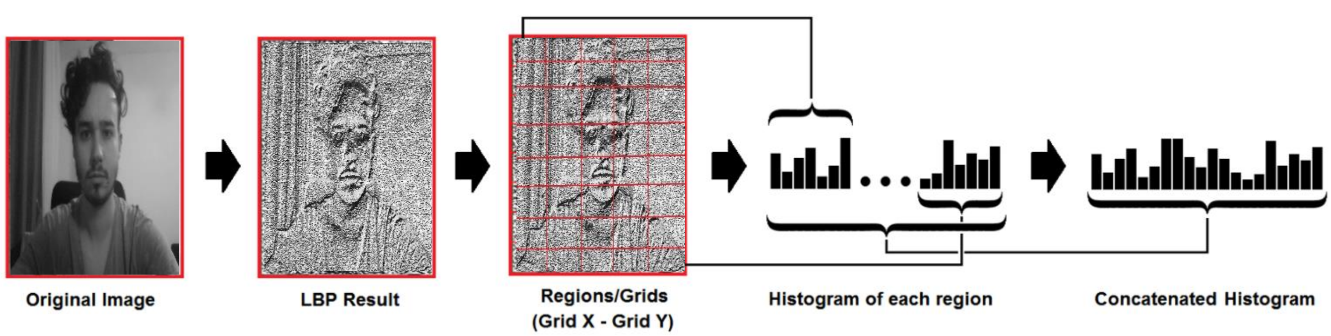


Figure 2.3: An example of calculating the LBP representation from the original image.

The last step that needs to be done is computing a histogram over the output LBP array is to create a histogram. Thus, since a vector has 8 neighbours, it has *256* possible solutions so that ,a LBP array has a value that may vary between 0 and 256.

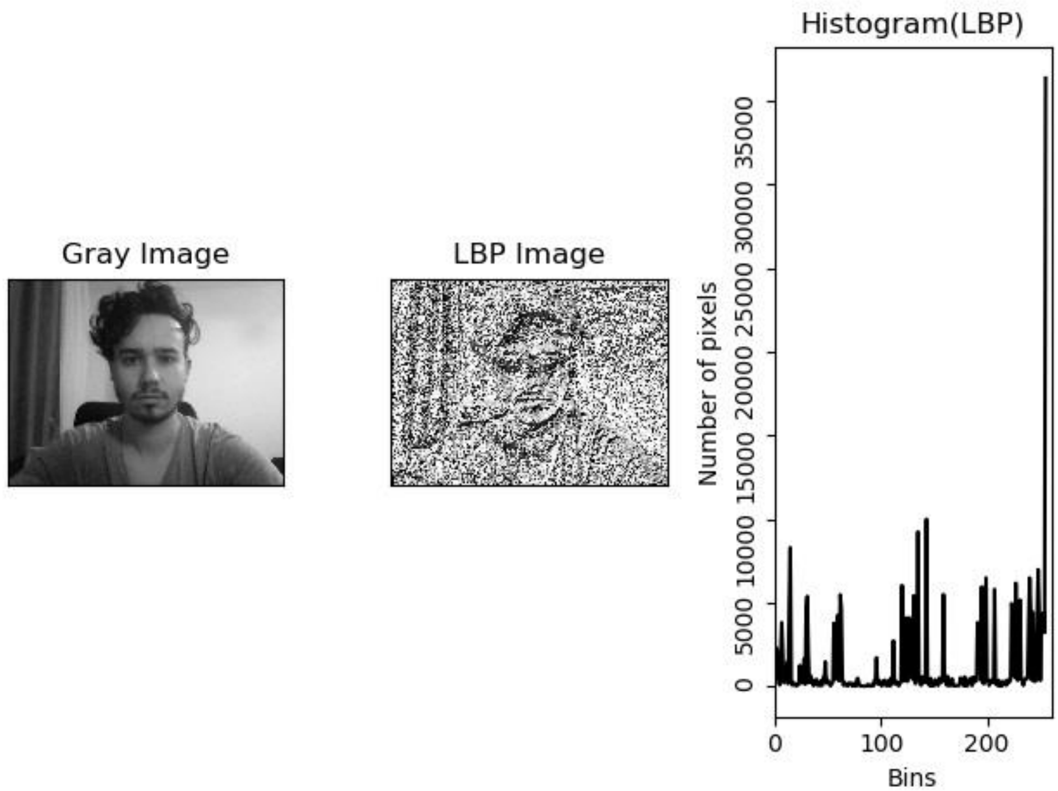


Figure 2.4: We calculate a histogram that counts the number of times each LBP pattern appears so that,we are going to use this histogram as our vector.

The most important benefit of this original LBP implementation is that we can record much more details in the image. However, the drawback of the algorithm is that we are able to capture details ,only at a 3x3 scale and not at a variable scale.

Because of this drawback, an extension to the original LBP implementation was proposed in [Ojala, T., Pietikäinen, M.] and, the two parameters were introduced:

1. The number of points **p** in a symmetric circularly neighborhood to avoid the square one and to maximize the range.

2. The radius of the circle **r**, which allows us to account for different scales.

In the figure 2.5 we have a visual representation of these parameters:

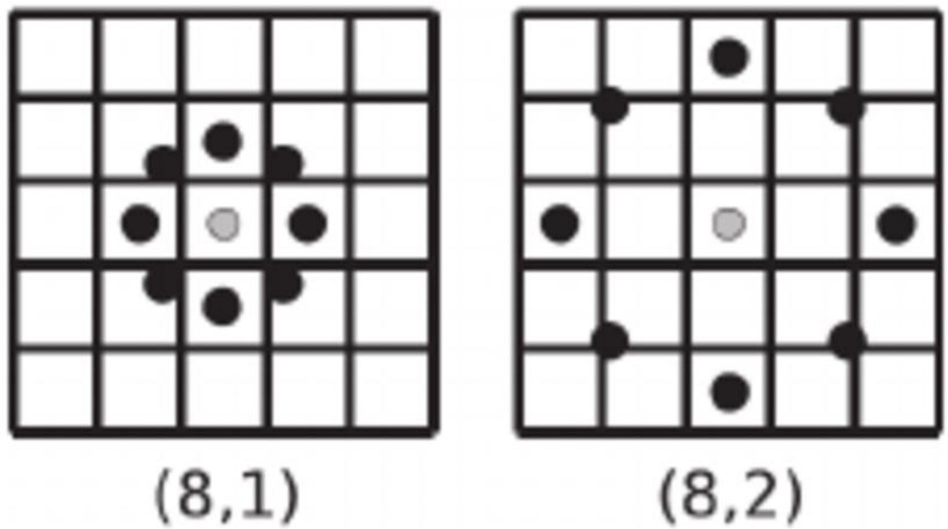


Figure 2.5: Three examples of neighborhood with varying *p* and *r* that were used to construct Local Binary Patterns.[Oravec]

Finally, it's important to take into consideration the concept of LBP ***uniformity***. A LBP is designed to be uniform if it has ***at most*** two *0-1* or *1-0* transitions. For example, the pattern 00001000 -2 transitions and 10000000 -1 transition are both considered to be **uniform patterns** because they have at most two *0-1* and *1-0* transitions. The pattern 01010010 on the other hand is *not* a uniform pattern since it has six *0-1* or *1-0* transitions.

The LBP patterns are so interesting because they add an extra level of *grayscale ,rotation and invariance*, hence they are used when extracting LBP feature vectors from images.

3. RANDOM FOREST

Random forest is a flexible machine learning algorithm. It is one of the most popular algorithms, because it can be used for both regression and classification tasks. The Random Forest algorithm is a classification method based on assemblies that operate with the notion of decision tree.

The methods based on assemblies work on the principle that a group of "weak classifiers" can create, alongside with a "strong classifier".In the case of Random Forest algorithm, the weak classifier is a decision tree, randomly generated. A singular decision tree is constructed from the entire dataset,the decisions being symbolised by the tree nodes that are generated based on the attributes in the dataset. Branches related to a node are the attribute values corresponding to that node.

Within Random Forest, several decision trees are generated from the subsets of the original dataset. Each tree thus generated shall be applied to the instance that has to be classified, resulting in a class for each tree. The final result of the classification is given by the majority class. The principle of the training stage of a Random Forest is illustrated in Fig 3. 1.

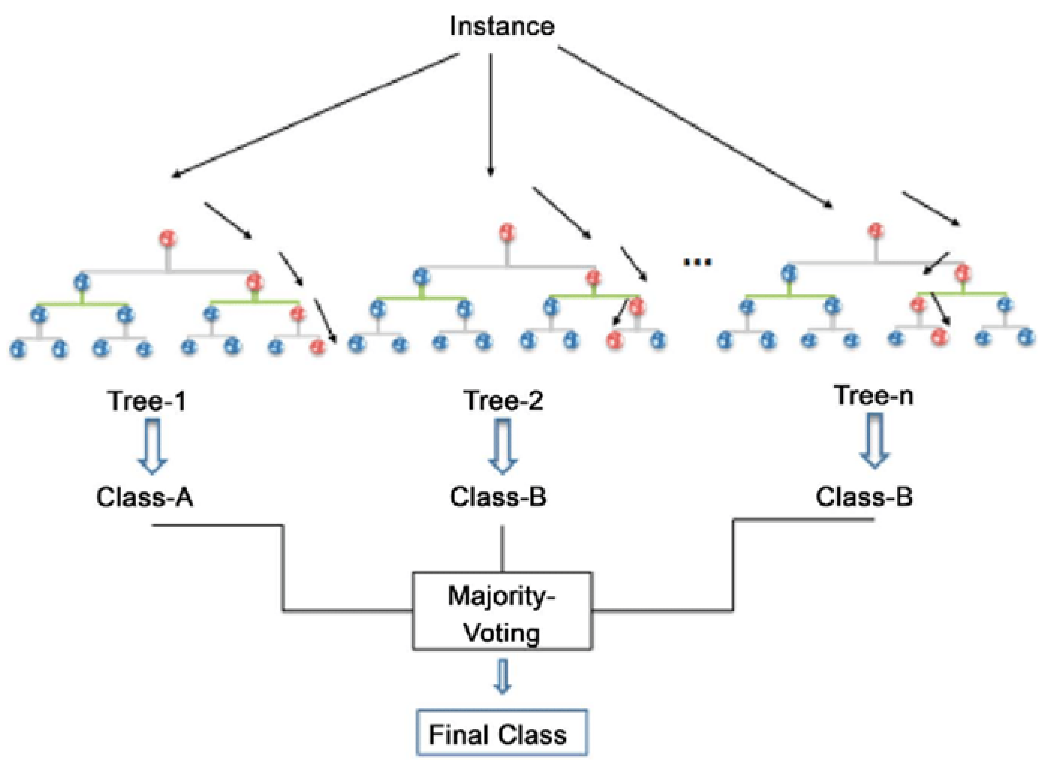


Fig 3.1: Demonstration of the Random Forest methodology[Fu, Yijie.]

The steps in generating a Random Forest algorithm are the following:

-Choose n subsets of K elements from the original dataset, K and N are arbitrary values, determined by the user

-For each sublot, a decision tree is generated, the function attributes of which the tree nodes are built are randomly chosen.

-An unclassified instance is generated (with a set of attribute values different from the training lot)

-The new instance is classified using each of the previously generated trees.Every tree goes through by following the branches that correspond to the values of the new instance attributes,reaching a leaf node containing the class in which the new instance falls.

-The majority class, the most frequently resulting from the application of the previously generated trees, shall be determined.

4. The Proposed Method

4.1 Block scheme

Figure 4.1.1 presents the Logic Scheme that was implemented in order to be able to create a program that detects the emotion of surprise.

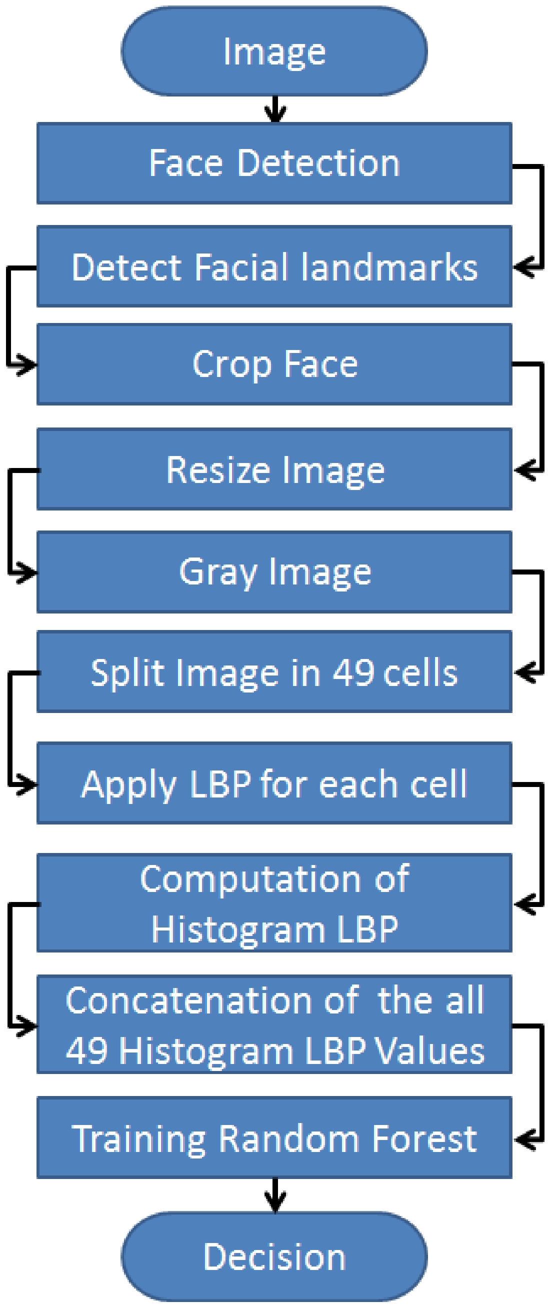


Figure 4.1.1: Block Scheme

In the Figure 4.1.2 a visual representation of the Blocks from the Fig 4.1.1 is presented.

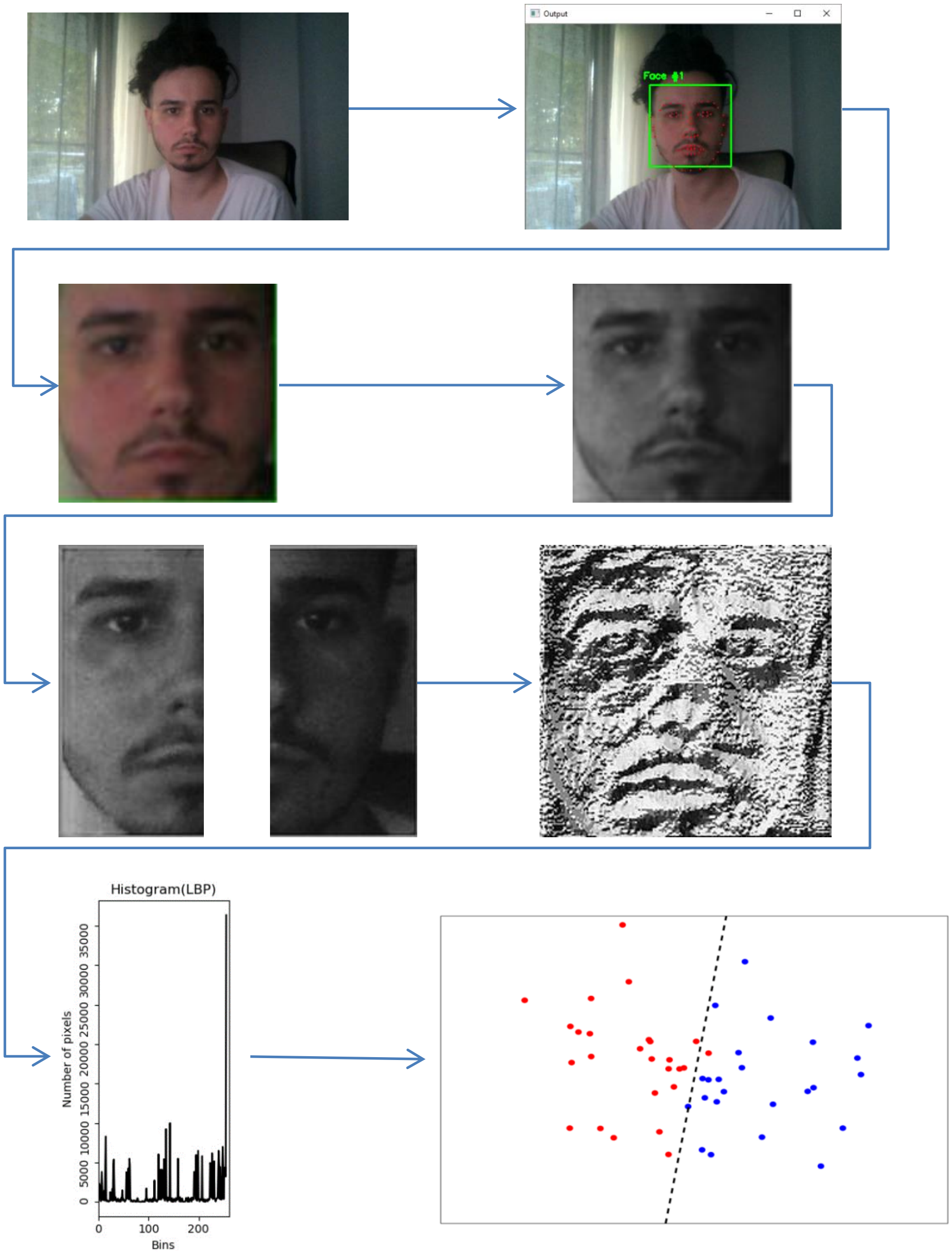


Fig 4.1.2: Exemplification of the block scheme elements

DLIB

Dlib is library software that was designed in 2002 in the programming language C++, it is capable of handling threads, graphical user interface, machine learning, image processing , data mining and so on .

The DLIB library was used in order to detect the face on the input image thus, by pre-processing the image with the face detection functions that it contains, the x and y coordinates of the face from the photo could be extracted. Because a square is defined by 4 points and not by 2 coordinates, the points that indicate the corners of the square had to be found. By finding the points that bound the face, we can use this bounding limits to crop the face in order to process the photo. Furthermore, finding the face, allows us the possibility to detect the visual facial landmarks on the face, presented in fig 4.1.3.

Fig 4.1.3: The 68 facial landmark points mark-up [Huber, Patrik.].

The individual Blocks that make up the Scheme

Face detection Block

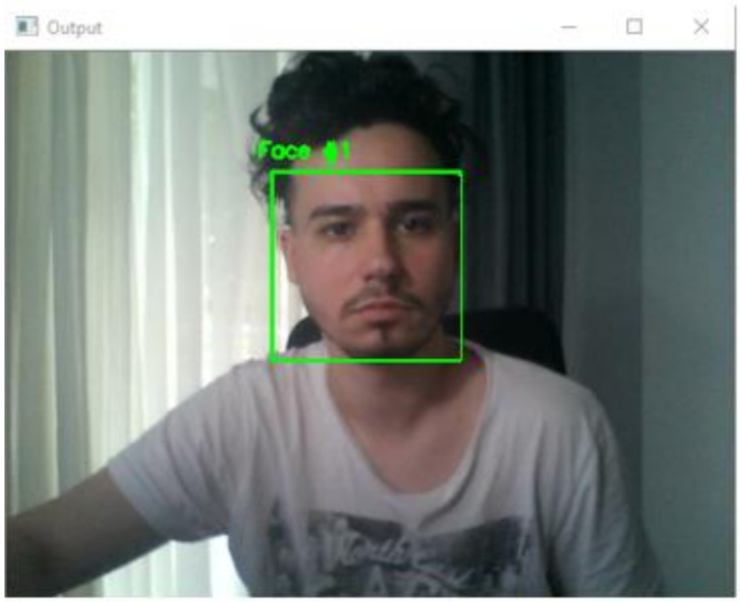


Fig 4.1.4: The Face Detection Block

To build our face recognition system, we'll first perform face detection.Face detection is one of the fundamental applications used in face recognition technology. Before the program is able to recognize the emotion of surprise, the software must be able to detect it firstly. The block detects the face and returns a vector (x,y,w,h) , containing the coordinates of the square that bounds the face.

Facial landmarks Block

Facial landmarks are used to detect and different regions of the face:

- Nose
- Mouth
- Jawline
- Eyes
- Eyebrows

Facial landmarks are used for blink detection ,face alignment, face swapping, head pose estimation but, In my license project, I am using them In order to be able to recognize the emotions on the face.

The goal of using facial landmarks is to detect important facial structures on the face. Detecting facial landmarks is therefore a two step process:

- Localize the face in the image.
- The detection of the key facial points on the face that are presented in the Fig 4.1.3.

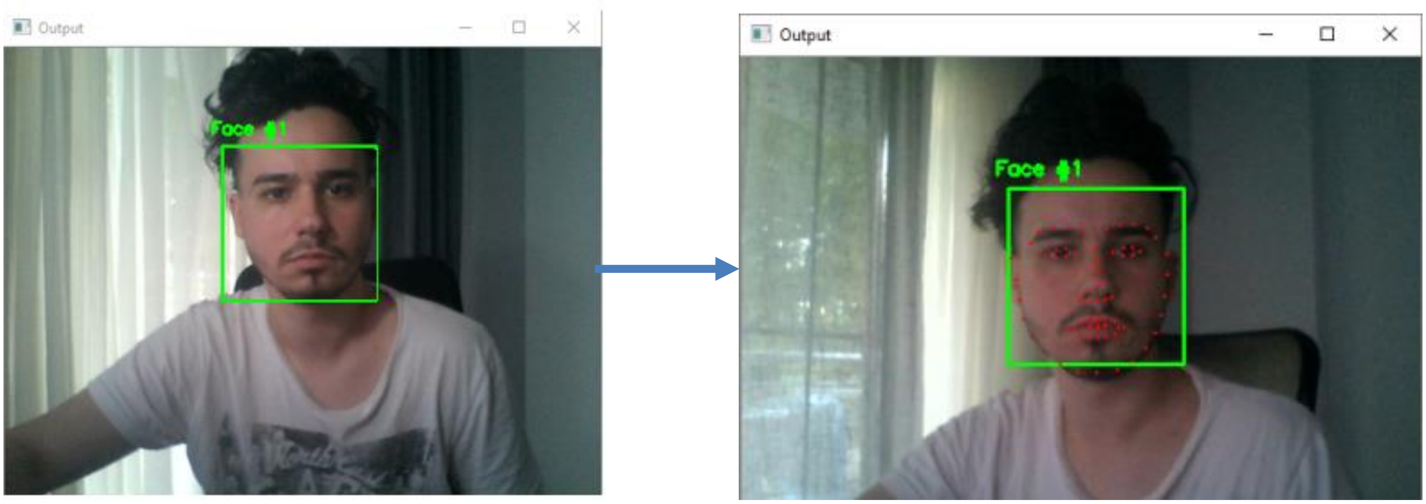


Fig 4.1.11: Facial landmarks Block

Face crop Block

Fig 4.1.5: The Face Crop Block

A face crop must have been done in order to apply the feature extractor algorithm (LBP) only on the interest zone , the face. My license project detects the emotion on the face so , I figured out that it would be useless and also, high resource consumption to compute the algorithm on the non-interest areas. To crop the face, I have used the vector returned from the face Face detection block (x,y,w,h) , to keep only the pixels that are in between these values of interest and delete the unnecessary part of the image.

Resize Image Block

In order to be able to slice the face image in a specific number of cells (49 in my case) , I have to have a fixed number of bits for every photo that I am going to process and further feed to my Random Forest Classifier. In order to find a proper Image size , I have followed the following steps :

- I have firstly analized the output image size of the face detection block for a several number of images .
- I have done the arithmetic mean between the sizes of the images and the number of the images.
- I have selected the closest number to the arithmetic mean , that divides perfectly by 49 (the number of cells that I have to divide the picture in).
- Finally, I have succeeded to find the proper Image Size, and that is 245x245.

Gray Image Block

Fig 4.1.6: Gray Image Block

Gray Image Block is necessary in order to be able to extract the features from the picture. This is possible because, Instead of having 3D model (RGB) , we have different levels of gray (1D) that we can further analyze.

Split Block

Fig 4.1.7: The Split Block

One of the key factors in achieving a good Win-Rate is the way that we choose process the Image. Thus, Instead of calculating LBP for the whole Face Image(256 values), I have choosen to split the Face Image in 49 equal cells and compute LBP for each of them, the Out Vector having 256x49 values. This method provides a higher level of data details extraction , being able to considerably improve the chances that the program to detect the emotion of surprise.Thus, Increasing the Win-Rate factor.

LBP block

Fig 4.1.8: LBP Block

LBP operator labels the pixels of an image with decimal numbers, which are called LBPs or LBP codes thatencode the local structure around each pixel. For each given pixel, a binary number isobtained by concatenating all these binary values in a clockwisedirection, which starts from the one of its top-left neighbor. Thecorresponding decimal value of the generated binary numberis then used for labeling the given pixel. The derived binarynumbers are referred to be the LBPs or LBP codes[Huang, di & Shan].

Histogram Block

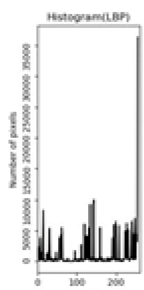


Fig 4.1.9: Histogram Block

The histogram of LBP labels calculated over a region can be ex-ploited as a texture descriptor.The LBP’s Histogram in computed with the values received from the LBP Block.The Histogram is computed for every picture in order to be later fed to the Classifier.

Training Block

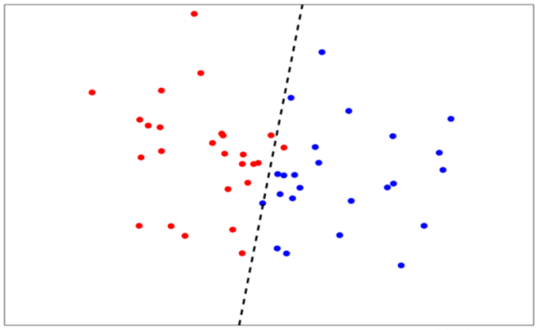


Fig 4.1.10: Training Block

The Training Block is responsible of the total understanding of the program.It has to make the program , understand what It should learn ,how It should learn and what to look at, in order to be able to further predict on an Image that It hasn't seen before.

4.2 Database

COHN-KANADE Database

The Cohn-Kanade database has been designed to help researchers To the development of detective algorithms and automatic recognition of facial emotions[T. Kanade, J. F. Cohn and Yingli Tian].The database contains 486 sequences of 78 subjects. Each sequence begins with a neutral expression and ends with an emotion at maximum intensity.

The emotion is encoded according to the FACS system and is labelled as such.

Fig 4.2.1: Front sequence; Exemplify the surprise AU 1 + 2 + 5 + 27, from the neutral state to maximum intensity

Fig 4.2.2: Secventa rotita la 30 grade; se exemplifica surpriza AU 1+2+5+27 de la starea neutral la intensitate maxima[Pollak, S. D.]

Database subjects are between 18 and 30 years of age, 65% women, 15% African-American and 3% Asian and Latinos. Each subject was filmed with two camera, one front located, and the second to 30 degrees to its right, as exemplified in the figures 4.1 and 4.2.

At this moment, only the frontal sequences are available, and the ones caught at 30 degrees will be published in the near future.Subjects were required to simulate 23 different facial expressions consisting of one or more AU. Each starts from a neutral expression and completes with maximum intensity of emotion.

In the figures 4.1 and 4.2 are displayed 3 stages of intensity of emotion, starting from the neutral state and ending with the maximum intensity. In each case of emotion, the subjects are taught, described and modeled the targeted expression.

To correctly accomplish the detection of the surprise emotion for the final project, I have used all 78 subjects. For each subject I selected 6 images of which: 3 Images with characters expressing the emotion of surprise, a subject that does not express any emotion (neutral) and two more pictures of randomly selected emotions, each of the two expressing one of the following emotions: Happiness, sadness, disgust, contempt, rage, fear.

4.3 Experimental Results

I have used Cohn-Kanade database that contains 486 photos of 78 subjects to test all the possible solutions on the training and testing set. To see what is going wrong and how my program got confused, I have decided to do the following procedure: For every subject from my database, to train the program to recognize the emotion of surprise for 77 of them and to test the algorithm on the one left that the program doesn't know anything about.

Results:

1) The program correctly detected all the emotions of surprise and the non-surprise emotions resulting in 100% Win-Rate for 38 Subjects.
2) The program correctly detected only 5 out of 6 photos for the emotions of surprise and the non-surprise emotions ,resulting in 83.33% Win-Rate for 23 Subjects.
3) The program correctly detected only 4 out of 6 photos for the emotions of surprise and the non-surprise emotions, resulting in 66.67% Win-Rate for 14 Subjects.
4) The program correctly detected only 3 out of 6 photos for the emotions of surprise and the non-surprise emotion, resulting in 50% Win-Rate for the last 3 Subjects.

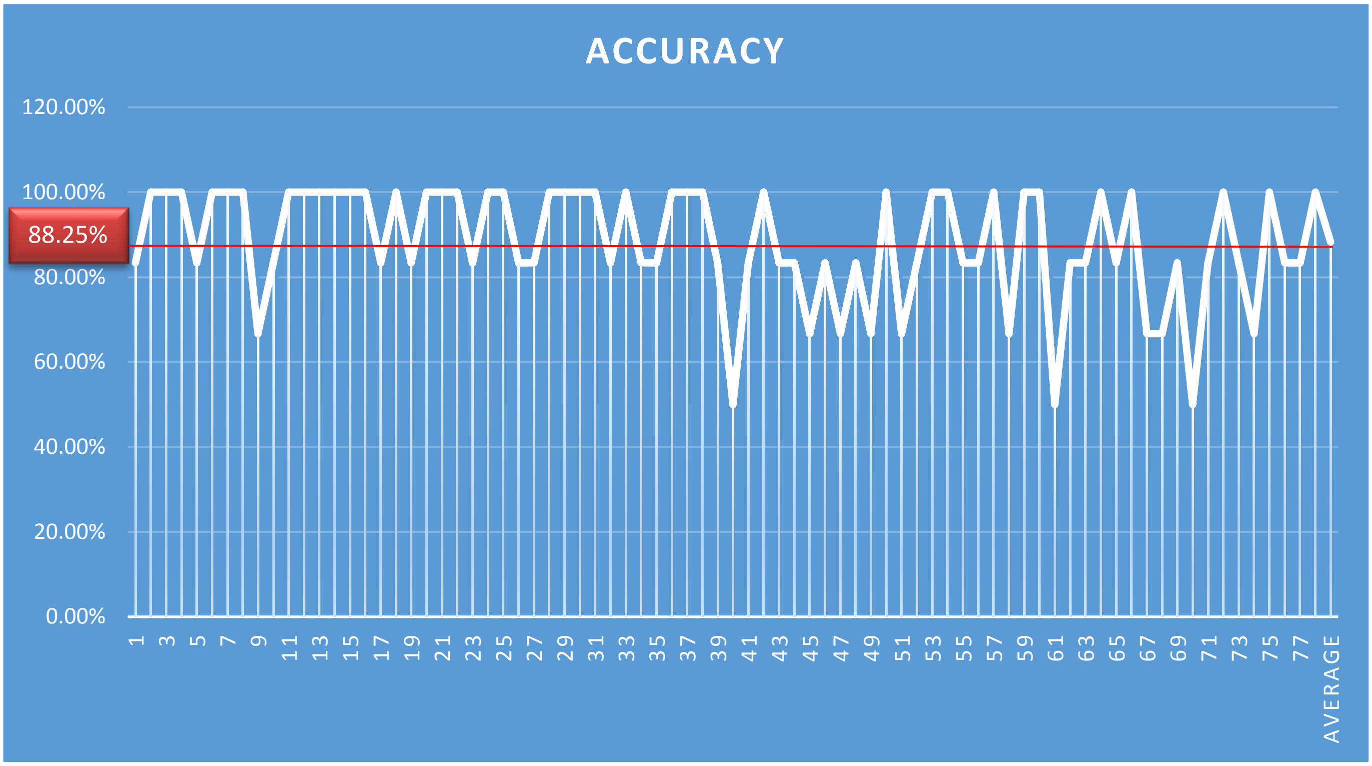
In The Table 1, are presented all the Results that I have obtained In Training and Test for each individual Subject.

SET	TOTAL TESTED PHOTOS	TOTAL CORRECT PHOTOS	ACCURACY
1	6	5	83.33%
2	6	6	100.00%
3	6	6	100.00%
4	6	6	100.00%
5	6	5	83.33%
6	6	6	100.00%
7	6	6	100.00%
8	6	6	100.00%
9	6	4	66.67%
10	6	5	83.33%
11	6	6	100.00%
12	6	6	100.00%
13	6	6	100.00%
14	6	6	100.00%
15	6	6	100.00%
16	6	6	100.00%
17	6	5	83.33%
18	6	6	100.00%
19	6	5	83.33%
20	6	6	100.00%
21	6	6	100.00%
22	6	6	100.00%
23	6	5	83.33%
24	6	6	100.00%
25	6	6	100.00%
26	6	5	83.33%
27	6	5	83.33%
28	6	6	100.00%
29	6	6	100.00%
30	6	6	100.00%
31	6	6	100.00%
32	6	5	83.33%
33	6	6	100.00%
34	6	5	83.33%
35	6	5	83.33%

36	6	6	100.00%
37	6	6	100.00%
38	6	6	100.00%
39	6	5	83.33%
40	6	3	50.00%
41	6	5	83.33%
42	6	6	100.00%
43	6	5	83.33%
44	6	5	83.33%
45	6	4	66.67%
46	6	5	83.33%
47	6	4	66.67%
48	6	5	83.33%
49	6	4	66.67%
50	6	6	100.00%
51	6	4	66.67%
52	6	5	83.33%
53	6	6	100.00%
54	6	6	100.00%
55	6	5	83.33%
56	6	5	83.33%
57	6	6	100.00%
58	6	4	66.67%
59	6	6	100.00%
60	6	6	100.00%
61	6	3	50.00%
62	6	5	83.33%
63	6	5	83.33%
64	6	6	100.00%
65	6	5	83.33%
66	6	6	100.00%
67	6	4	66.67%
68	6	4	66.67%
69	6	5	83.33%
70	6	3	50.00%
71	6	5	83.33%
72	6	6	100.00%
73	6	5	83.33%
74	6	4	66.67%
75	6	6	100.00%
76	6	5	83.33%
77	6	5	83.33%
78	6	6	100.00%
TOTAL	**468**	**413**	**88.25%**

Table1: The accuracy for each subject from the database

The accuracy for each subject from the Table 1 is also mapped on the **Graph 1** to allow a better visualization of the results.



Graph 1: The accuracy for each subject from the database

In The Table 2, are presented all the Results that I have obtained In Training and Test for each individual Emotion.

The surprise emotion scorred 90.04% accuracy while, the other non-surprise emotions scorred :85.37% accuracy for the neutral, 83.72% for happiness, 95,24% for fear, 82.61% for contempt, 89,66% for sadness, 88,98% for disgust and 86,67% for anger.

EMOTION	SURPRISE	NEUTRAL	HAPPINESS	FEAR	CONTEMPT	SADNESS	DISGUST	ANGER
PHOTOS	231	82	43	21	23	29	9	30
CORRECT	90.04%	85.37%	83.72%	95.24%	82.61%	89.66%	88.89%	86.67%
INCORRECT	9.96%	14.63%	16.28%	4.76%	17.39%	10.34%	11.11%	13.33%

Table 2: The accuracy for the surprise and for the non-surprise emotions

Taking into consideration the Results received from the program, I have decided to further investigate the problem, to see where and why my program does not correctly detect some emotions.

Comparing the photos from my database and the results received from my program, I have decided that , some photos from the database were not very explicit:

- For the 3 subjects , that the program detected correctly only a half of the photos, the photos were too dark:

Fig 4.3.1: Dark Photo Subjects

- For 7 subjects, that the program have not correctly detected 2 photos, I may say that the subjects from the database were wearing hats and were having bangs:

Fig 4.3.2: Subjects with Bangs

Fig 4.3.3: Subjects with Bangs and Hats

5. Conclusions and research directions

One of the simple forms of recognition involves withholding the primary information of an image in memory, in order to be compared later with another image.

The proper recognition of emotions in facial expressions would imply additional information about expressions and their connection to different stimuli environment with which that expression was directly or indirectly in contact.

Mathematical analyses have shown that the structure present in the images of facial expressions is sufficient, in principle, to generate part of the emotional categories perceived by people.

The evaluations of emotional expressions have been subjected to a multidimensional scaling algorithm. Currently the system detects with an accuracy of over 90%.

The system works with certain limitations depending on how the photos are accomplished.

In Conclusion, the license program works with an accuracy of 90.04% for the surprise emotion on the tested database but, like any other product it has some drawbacks. As I have tested , these drawbacks are regarded to the:

-Ambient light in which the picture was taken.

-The image contrast.

-The resolution of the image.

-Subjects should not wear hats, bangs or mustache that are covering important region on the face.

The research direction that I am going to follow would be to successfully implement the detection of the whole set of emotions, not only for the emotion of Surprise to accomplish a psychological analysis of a person.

My Expectations regarding the usefulness of the program :

In the future, if a more complex database with a lot more subject would be used, and the software would be improved to detect not only the surprise emotion, but all the 7 emotions. The program can be implemented in stores and offices to detect the people emotion when they receive a task or when they are looking for a product.

It certainly can be used in the automotive domain, to prevent car accidents: if the user is angry , some actions can be taken automatically by the car, to prevent a disaster.

Another way that the software could be used is to improve the face recognition, this study is certified and a project like this is already implemented and functional in China.

7. Bibliography

[1] Leonard Gavriliu Introductory Essay (1997)- J.P.Sartre - Psychology of emotion

[2] Charles Darwin (1872)-Expression of emotions in humans and animals

[3] Dolores Cañamero (1999)- What Emotions Are Necessary for HCI?

[4] Zajonc, R. B. (1980)- Feeling and Thinking: Preferences Need No Inferences. American Psychologist, 35, 151-175.

[5] Cole PM, Martin SE, Dennis TA (2004) Emotion regulation as a scientific construct: Methodologicalchallenges and directions for child development research. Child development 75(2):317–333

[6] Lazarus, R. S. (1982). Thoughts on the relations between emotion and cognition. *American Psychologist, 37*(9), 1019-1024.

[7] Goleman D (2006) Emotional intelligence. Bantam

[8] William James (1884)-What is a feeling

[9] Arnold, M. B. (1960). Emotion and personality. New York, NY, US: Columbia University Press.

[10] Cornelius RR (2000) Theoretical approaches to emotion. In: ISCA Tutorial and Research.Workshop (ITRW) on Speech and Emotion

[11] Ekman P. An argument for basic emotions. Cogn. Emot. 1992;6:169–200. doi: 10.1080/02699939208411068.

[12] Plutchik R. The Nature of Emotions. Am. Sci. 2001;89:344–350. doi: 10.1511/2001.4.344

[13] Lang P.J. The emotion probe: Studies of motivation and attention. Am. Psychol. 1995;50:372–385. doi: 10.1037/0003-066X.50.5.372.

[14] Mehrabian A. Comparison of the PAD and PANAS as models for describing emotions and for differentiating anxiety from depression. J. Psychopathol. Behav. Assess. 1997;19:331–357.

[15] Bai L., Ma H., Huang H., Luo Y. The compilation of Chinese emotional picture system—A trial of 46 Chinese college students. Chin. J. Ment. Health. 2005;19:719–722

[16] Lang P.J. International Affective Picture System (IAPS): Technical Manual and Affective Ratings. The Center for Research in Psychophysiology, University of Florida; Gainesville, FL, USA: 1995.

[17] Nasoz F., Alvarez K. Emotion Recognition from Physiological Signals for Presence Technologies. Int. J. Cogn. Technol. Work. 2003;6:4–14. doi: 10.1007/s10111-003-0143-x.

[18] Chung-Hsien Wu (2012) and Chao-Hong Liu, “Robust Speech Recognition for Adverse Environments.” Modern Speech Recognition - Approaches with Case Studies, Chapter 1.

[19] Ekman, P. and Friesen, W.V. (1969) The Repertoire or Nonverbal Behavior: Categories, Origins, Usage and Coding. Semiotica, 1, 49-98.

[20] P. Ekman and W. Friesen(1978)- Facial Action Coding System: A Technique for the Measurement of Facial Movement.

[21] Ekman P, Friesen WV (1977) Facial action coding system.

[22] Ekman, P. (1970)- Universal Facial Expressions of Emotions. California Mental Health Research Digest, 8(4), 151-158.

[23] Ekman, Friesen & Hager, 2002; Facial Action Coding System: Score Sheet, pg.513-514.

[24] Ekman, P., Friesen, W. V., & Hager, J. C. (2002). Facial Action Coding System.

[25] T. Ojala, M. Pietikainen and T. Maenpaa, Multiresolution Grayscale and Rotation Invariant Texture Classification with Local Binary Patterns(2002).

[26] Ojala, T., Pietikäinen, M., Harwood, D.: A Comparative Study of Texture Measures with Classification Based on Feature Distributions. Pattern Recognition 29 (1996) 51–59.

[27]Pollak, S. D., Messner, M., Kistler, D. J., & Cohn, J. F. (2009). Development of perceptual expertise in emotion recognition. Cognition, 110(2), 242-247

[28] Oravec, Milos & Pavlovicova, Jarmila & Mazanec, Jan & Omelina, Lubos & Feder, Matej & Ban, Jozef. (2011). Efficiency of Recognition Methods for Single Sample per Person Based Face Recognition. 10.5772/18432.

[29] Fu, Yijie. (2017). Combination of Random Forests and Neural Networks in Social Lending. Journal of Financial Risk Management. 06. 418-426. 10.4236/jfrm.2017.64030.

[30] Huber, Patrik. (2018). Real-time 3D morphable shape model fitting to monocular in-the-wild videos.

[31] Huang, di & Shan, Caifeng & Ardabilian, Mohsen & Wang, Yunhong & Chen, Liming. (2011). Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C. 41. 765-781. 10.1109/TSMCC.2011.2118750.

[32] Descartes' Error: Emotion, Reason, and the Human Brain, by Antonio Damasio,2006

[33] The verbal expressions of the emotions placed on the Pleasure–Arousal grid according to Russell Another version of emotional wheel is the Geneva Emotion Wheel (GEW) introduced as an experimental tool (Scherer, 2005; Bänziger et al., 2005)

[34] Advances in Intelligent Systems and Computing ,Chapter: Recognition of Emotion Intensity Basing on Neutral Speech Model,Publisher: Springer, Editors: A.Gruca, T.Czachórski, S.Kozielski,2004

[35] Yangzhou Du,Proceedings of the 6th ACM International Conference on Image and Video Retrieval, CIVR 2007, Amsterdam, The Netherlands, July 9-11, 2007

[36] C. Breazeal and B. Scassellati, “How to build robots that make friends and influence people,” in Proc. IROS99, Kyonjiu, Korea, 1999, pp. 858– 863

[37] Emotion Recognition from Physiological Signals for Presence Technologies, Kaye Alvarez,January 2003

[38] Illustration of facial muscular system. The approximate locations of facial muscles used in this study are marked. (Figure adapted and slightly modified from Wikimedia Commons, public domain. Author: Patrick B. Lynch, medical illustrator, January 2016

[39] C.-H. Wu, J.-C. Lin, and W.-L. Wei. Survey on audiovisual emotion recognition: databases, features, and data fusion strategies. APSIPA Transactions on Signal and Information Processing, 3, 2014.

[40] T. Kanade, J. F. Cohn and Yingli Tian, "Comprehensive database for facial expression analysis," Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580), Grenoble, France, 2000, pp. 46-53.

The Anex

	TESTED PHOTOS																								TOTAL PHOTOS TESTED	CORRECT TESTED PHOTOS																								TOTAL CORRECT DETECTED PHOTOS	ACCURACY
SET	SURPRISE			NEUTRAL			HAPPINESS			FEAR			CONTEMPT			SADNESS			DISGUST			ANGER				SURPRISE			NEUTRAL			HAPPINESS			FEAR			CONTEMPT			SADNESS			DISGUST			ANGER				
1	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	0	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
2	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	100.00%
3	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
4	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
5	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	100.00%
7	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
8	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
9	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	4	66.67%
10	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
11	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
12	0	0	0	1	0	0	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	6	100.00%
13	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
14	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	100.00%
15	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	100.00%
16	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6	100.00%
17	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5	83.33%
18	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
19	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	5	83.33%
20	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
21	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6	100.00%
22	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
23	1	1	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
24	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
25	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
26	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
27	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
28	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
29	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	100.00%
30	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
31	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
32	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	5	83.33%
33	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	100.00%
34	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	6	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	5	83.33%
35	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	5	83.33%
36	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
37	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	6	100.00%
38	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
39	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
40	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	50.00%
41	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	5	83.33%
42	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	100.00%
43	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	5	83.33%
44	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5	83.33%
45	1	1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	4	66.67%
46	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
47	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	4	66.67%
48	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5	83.33%
49	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	6	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	4	66.67%
50	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	100.00%
51	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	4	66.67%
52	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	5	83.33%
53	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	100.00%
54	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
55	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	5	83.33%
56	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	6	1	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	5	83.33%
57	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	100.00%
58	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	4	66.67%
59	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%
60	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	6	100.00%
61	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	50.00%
62	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	5	83.33%
63	1	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	5	83.33%
64	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	100.00%
65	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	5	83.33%
66	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	6	100.00%
67	1	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	66.67%
68	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	66.67%
69	1	1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	5	83.33%
70	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	50.00%
71	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	5	83.33%
72	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	100.00%

73	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	0	1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5	83.33%
74	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	6	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	66.67%
75	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	100.00%
76	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	6	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	5	83.33%
77	1	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	5	83.33%
78	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	6	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	6	100.00%
TOTAL	**77**	**77**	**77**	**78**	**4**	**0**	**42**	**1**	**0**	**19**	**1**	**1**	**21**	**1**	**1**	**28**	**1**	**0**	**8**	**1**	**0**	**30**	**0**	**0**	**468**	**66**	**75**	**67**	**67**	**3**	**0**	**35**	**1**	**0**	**18**	**1**	**1**	**#**	**1**	**1**	**25**	**1**	**0**	**8**	**0**	**0**	**26**	**0**	**0**	**413**	**88.25%**
TOTAL PHOTOS OF EMOTIONS			**231**			**82**			**43**			**21**			**23**			**29**			**9**			**30**				**208**			**70**			**36**			**20**			**19**			**26**			**8**			**26**		
CORRECT			**90.04%**			**85.37%**			**83.72%**			**95.24%**			**82.61%**			**89.66%**			**88.89%**			**86.67%**																											
INCORRECT			**9.96%**			**14.63%**			**16.28%**			**4.76%**			**17.39%**			**10.34%**			**11.11%**			**13.33%**																											

Table 3: Detailed analysis of study evolution

Training the algorithm

```
# import the necessary packages
from math import floor
import sys
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier, forest
import image_slicer
from imutils import face_utils
import numpy as np
import dlib
import cv2
from matplotlib import pyplot as plt
from PIL import Image
import glob
import os
import pickle

def set_rf_samples(n):
    forest._generate_sample_indices = (lambda rs, n_samples:
    forest.check_random_state(rs).randint(0, n_samples, n))

def get_pixel(img, center, x, y):
    new_value = 0
    try:
        if img[x][y] >= center:
            new_value = 1
    except:
        pass
    return new_value


def lbp_calculated_pixel(img, x, y):
    '''
     64 | 128 |   1
    ----------------
     32 |   0 |   2
    ----------------
     16 |   8 |   4
    '''
    center = img[x][y]
    val_ar = []
    val_ar.append(get_pixel(img, center, x - 1, y + 1))  # top_right
    val_ar.append(get_pixel(img, center, x, y + 1))  # right
    val_ar.append(get_pixel(img, center, x + 1, y + 1))  # bottom_right
```

```python
    val_ar.append(get_pixel(img, center, x + 1, y))   # bottom
    val_ar.append(get_pixel(img, center, x + 1, y - 1))  # bottom_left
    val_ar.append(get_pixel(img, center, x, y - 1))  # left
    val_ar.append(get_pixel(img, center, x - 1, y - 1))  # top_left
    val_ar.append(get_pixel(img, center, x - 1, y))  # top

    power_val = [1, 2, 4, 8, 16, 32, 64, 128]
    val = 0
    for i in range(len(val_ar)):
        val += val_ar[i] * power_val[i]
    return val


def show_output(output_list):
    output_list_len = len(output_list)
    figure = plt.figure()
    for i in range(output_list_len):
        current_dict = output_list[i]
        current_img = current_dict["img"]
        current_xlabel = current_dict["xlabel"]
        current_ylabel = current_dict["ylabel"]
        current_xtick = current_dict["xtick"]
        current_ytick = current_dict["ytick"]
        current_title = current_dict["title"]
        current_type = current_dict["type"]
        current_plot = figure.add_subplot(1, output_list_len, i + 1)
        if current_type == "gray":
            current_plot.imshow(current_img, cmap=plt.get_cmap('gray'))
            current_plot.set_title(current_title)
            current_plot.set_xticks(current_xtick)
            current_plot.set_yticks(current_ytick)
            current_plot.set_xlabel(current_xlabel)
            current_plot.set_ylabel(current_ylabel)
        elif current_type == "histogram":
            current_plot.plot(current_img, color="black")
            current_plot.set_xlim([0, 256])
            current_plot.set_title(current_title)
            current_plot.set_xlabel(current_xlabel)
            current_plot.set_ylabel(current_ylabel)
            ytick_list = [int(i) for i in current_plot.get_yticks()]
            current_plot.set_yticklabels(ytick_list, rotation=90)

    plt.show()



#to create the path
dir_name='C:\\Users\\Marius\\PycharmProjects\\licenta\\train_img'
format = 'png'
indice='\\'
label_vector=[]
hbins=256
slices=49
hist_vector = np.zeros([468, hbins*slices])
idx = 0
for n in range(1,79):
    print('SET=',n)
    idx2 = 0
    # indicele de poza
    base_filename=n
    base_filename=str(base_filename)
    base_filename=base_filename+indice
    txt_files = os.path.join(dir_name, base_filename)
    for filename in glob.glob(txt_files + '*.txt'):
```

```
        file = open(filename)
        train_labels_string = file.read()
        file.close()
        train_labels_list = train_labels_string.split(',')

    for filename in glob.glob(txt_files + '*.png'):
        detector = dlib.get_frontal_face_detector()
        predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
        image = cv2.imread(filename)
        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        rects = detector(gray, 1)
        # loop over the face detections
        for (i, rect) in enumerate(rects):
            # determine the facial landmarks for the face region, then
            # convert the facial landmark (x, y)-coordinates to a NumPy
            # array
            shape = predictor(gray, rect)
            shape = face_utils.shape_to_np(shape)

            # convert dlib's rectangle to a OpenCV-style bounding box
            (x, y, w, h) = face_utils.rect_to_bb(rect)
            cv2.rectangle(image, (x, y), (x + w, y + h), (0, 255, 0), 2)
            crop_img=image[y:y+h,x:x+w]
            cv2.imwrite('crop_img.jpg', crop_img)
                        cv2.putText(image, "Face #{}".format(i + 1), (x - 10, y - 10),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)
            for (x, y) in shape:
                cv2.circle(image, (x, y), 1, (0, 0, 255), -1)

        crop_image = Image.open('crop_img.jpg')
        # resize to 245x245
        basewidth = 245
        wpercent = (basewidth / float(crop_image.size[0]))
        hsize = int((float(crop_image.size[1]) * float(wpercent)))
        crop_image = crop_image.resize((basewidth, hsize), Image.ANTIALIAS)
        crop_image1=crop_image
        crop_image.save('resized.jpg')
        crop_image = 'resized.jpg'
        tiles = image_slicer.slice('resized.jpg', slices, save=False)
        image_slicer.save_tiles(tiles, '',
'C:\\Users\\Marius\\PycharmProjects\\licenta\\slice', 'png')
        hist_lbp_v = np.zeros([1, hbins*slices])
        idx1 = 0
        for filename in
glob.glob('C:\\Users\\Marius\\PycharmProjects\\licenta\\slice/*.png'):
            crop_read = cv2.imread(filename)
            height, width, channel = crop_read.shape
            img_gray = cv2.cvtColor(crop_read, cv2.COLOR_BGR2GRAY)
            img_lbp = np.zeros((height, width, 3), np.uint8)
            for i in range(0, height):
                for j in range(0, width):
                    img_lbp[i, j] = lbp_calculated_pixel(img_gray, i, j)
            cv2.imwrite('pic2.jpg', img_lbp)
            # show the output lbp_image with the face detections + facial landmarks
            hist_lbp = cv2.calcHist([img_lbp], [0], None, [hbins], [0, 256])
            hist_lbp_t=np.transpose(hist_lbp)
            hist_lbp_v[:, idx1 *hbins:(idx1 + 1) * hbins] = hist_lbp_t
            idx1 = idx1 + 1
            if(idx2<147):
                output_list = []
                output_list.append({
                    "img": crop_image1,
                    "xlabel": "",
                    "ylabel": "",
```

```
                    "xtick": [],
                    "ytick": [],
                    "title": "Gray Image",
                    "type": "gray"
                })
                output_list.append({
                        "img": img_gray,
                        "xlabel": "",
                        "ylabel": "",
                        "xtick": [],
                        "ytick": [],
                        "title": "Gray Image",
                        "type": "gray"
                    })
                output_list.append({
                        "img": img_lbp,
                        "xlabel": "",
                        "ylabel": "",
                        "xtick": [],
                        "ytick": [],
                        "title": "LBP Image",
                        "type": "gray"
                    })
                output_list.append({
                        "img": hist_lbp,
                        "xlabel": "Bins",
                        "ylabel": "Number of pixels",
                        "xtick": None,
                        "ytick": None,
                        "title": "Histogram(LBP)",
                        "type": "histogram"
                    })
            idx2=idx2+1
        hist_vector[idx, :] = hist_lbp_v
        idx = idx + 1
    label_vector=label_vector+train_labels_list
X = hist_vector
y = label_vector  # Labels
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3) # 70% training and
30% test
Train=468
model= RandomForestClassifier(n_estimators=1000)
set_rf_samples(floor(0.8*Train))

model.fit(X_train,y_train)
res1=model.score(X_test,y_test)
res2=model.score(X_train,y_train)
print('SCORE_TRAIN:',res2)
print('SCORE_TEST:',res1)

modell='finalized_model.sav'
pickle.dump(model,open(modell,'wb'))
loaded_model=pickle.load(open(modell,'rb'))
result=loaded_model.score(X,y)
sys.exit(0)
```

Testing the algorithm

```
# import the necessary packages
import sys
import image_slicer
from imutils import face_utils
import numpy as np
import dlib
import cv2
from matplotlib import pyplot as plt
from PIL import Image
import glob
import pickle

def get_pixel(img, center, x, y):
    new_value = 0
    try:
        if img[x][y] >= center:
            new_value = 1
    except:
        pass
    return new_value


def lbp_calculated_pixel(img, x, y):
    '''
     64 | 128 |   1
    ----------------
     32 |   0 |   2
    ----------------
     16 |   8 |   4
    '''
    center = img[x][y]
    val_ar = []
    val_ar.append(get_pixel(img, center, x - 1, y + 1))  # top_right
    val_ar.append(get_pixel(img, center, x, y + 1))  # right
    val_ar.append(get_pixel(img, center, x + 1, y + 1))  # bottom_right
    val_ar.append(get_pixel(img, center, x + 1, y))  # bottom
    val_ar.append(get_pixel(img, center, x + 1, y - 1))  # bottom_left
    val_ar.append(get_pixel(img, center, x, y - 1))  # left
    val_ar.append(get_pixel(img, center, x - 1, y - 1))  # top_left
    val_ar.append(get_pixel(img, center, x - 1, y))  # top

    power_val = [1, 2, 4, 8, 16, 32, 64, 128]
    val = 0
    for i in range(len(val_ar)):
        val += val_ar[i] * power_val[i]
    return val


def show_output(output_list):
    output_list_len = len(output_list)
    figure = plt.figure()
    for i in range(output_list_len):
        current_dict = output_list[i]
        current_img = current_dict["img"]
        current_xlabel = current_dict["xlabel"]
        current_ylabel = current_dict["ylabel"]
        current_xtick = current_dict["xtick"]
        current_ytick = current_dict["ytick"]
        current_title = current_dict["title"]
        current_type = current_dict["type"]
        current_plot = figure.add_subplot(1, output_list_len, i + 1)
```

```
        if current_type == "gray":
            current_plot.imshow(current_img, cmap=plt.get_cmap('gray'))
            current_plot.set_title(current_title)
            current_plot.set_xticks(current_xtick)
            current_plot.set_yticks(current_ytick)
            current_plot.set_xlabel(current_xlabel)
            current_plot.set_ylabel(current_ylabel)
        elif current_type == "histogram":
            current_plot.plot(current_img, color="black")
            current_plot.set_xlim([0, 256])
            current_plot.set_title(current_title)
            current_plot.set_xlabel(current_xlabel)
            current_plot.set_ylabel(current_ylabel)
            ytick_list = [int(i) for i in current_plot.get_yticks()]
            current_plot.set_yticklabels(ytick_list, rotation=90)

    plt.show()



#save image from laptop's camera
pict=cv2.VideoCapture(0)
while(True):
    ret, frame = pict.read()
    rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    cv2.imshow('Picframe', rgb)
    if cv2.waitKey(1):
        out = cv2.imwrite('pic.jpg', rgb)
        break


pict.release()
cv2.destroyAllWindows()
#end
hist_vector=np.zeros([1,12544])
idx=0
idx2=0
idx3=1


#make the saved image from the camera gray
image_file = 'pic.jpg'
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
image = cv2.imread(image_file)
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
rects = detector(gray, 1)
# loop over the face detections
for (i, rect) in enumerate(rects):
    # determine the facial landmarks for the face region, then
    # convert the facial landmark (x, y)-coordinates to a NumPy
    # array
    shape = predictor(gray, rect)
    shape = face_utils.shape_to_np(shape)

    # convert dlib's rectangle to a OpenCV-style bounding box
    (x, y, w, h) = face_utils.rect_to_bb(rect)
    cv2.rectangle(image, (x, y), (x + w, y + h), (0, 255, 0), 2)
    crop_img=image[y:y+h,x:x+w]
    #cv2.imshow('cropped',crop_img)
    cv2.imwrite('crop_img.jpg', crop_img)

    # show the face number
```

```
    cv2.putText(image, "Face #{}".format(i + 1), (x - 10, y - 10),
                cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)

    # loop over the (x, y)-coordinates for the facial landmarks
    # and draw them on the image
    for (x, y) in shape:
        cv2.circle(image, (x, y), 1, (0, 0, 255), -1)

crop_image = Image.open('crop_img.jpg')
# resize to 245x245
basewidth = 245
wpercent = (basewidth / float(crop_image.size[0]))
hsize = int((float(crop_image.size[1]) * float(wpercent)))
crop_image = crop_image.resize((basewidth, hsize), Image.ANTIALIAS)
crop_image1=crop_image
crop_image.save('resized.jpg')
crop_image = 'resized.jpg'
tiles = image_slicer.slice('resized.jpg', 49, save=False)
image_slicer.save_tiles(tiles, '', 'C:\\Users\\Marius\\PycharmProjects\\licenta\\slice',
'png')
hist_lbp_v = np.zeros([1, 12544])
idx1 = 0
print('poza:', idx3)
for filename in glob.glob('C:\\Users\\Marius\\PycharmProjects\\licenta\\slice/*.png'):
    crop_read = cv2.imread(filename)
    height, width, channel = crop_read.shape
    img_gray = cv2.cvtColor(crop_read, cv2.COLOR_BGR2GRAY)
    img_lbp = np.zeros((height, width, 3), np.uint8)
    for i in range(0, height):
        for j in range(0, width):
            img_lbp[i, j] = lbp_calculated_pixel(img_gray, i, j)
    cv2.imwrite('pic2.jpg', img_lbp)
    # show the output lbp_image with the face detections + facial landmarks
    hbins=256
    hist_lbp = cv2.calcHist([img_lbp], [0], None, [hbins], [0, 256])
    hist_lbp_t=np.transpose(hist_lbp)

    hist_lbp_v[:, idx1 * 256:(idx1 + 1) * 256] = hist_lbp_t
    idx1 = idx1 + 1

    if(idx2<147):
        output_list = []
        output_list.append({
            "img": crop_image1,
            "xlabel": "",
            "ylabel": "",
            "xtick": [],
            "ytick": [],
            "title": "Gray Image",
            "type": "gray"
        })
        output_list.append({
                "img": img_gray,
                "xlabel": "",
                "ylabel": "",
                "xtick": [],
                "ytick": [],
                "title": "Gray Image",
                "type": "gray"
            })
        output_list.append({
                "img": img_lbp,
                "xlabel": "",
                "ylabel": "",
```

```
                "xtick": [],
                "ytick": [],
               "title": "LBP Image",
                "type": "gray"
            })
        output_list.append({
                "img": hist_lbp,
                "xlabel": "Bins",
                "ylabel": "Number of pixels",
                "xtick": None,
                "ytick": None,
                "title": "Histogram(LBP)",
                "type": "histogram"
            })
    idx2=idx2+1
idx3=idx3+1
hist_vector[idx, :] = hist_lbp_v
idx = idx + 1
print(hist_vector)
modell='finalized_model.sav'
loaded_model=pickle.load(open(modell,'rb'))
result=loaded_model.predict(hist_vector)
print(result)
if (result=='1'):
    image = Image.open('100.png')
    image.show()


sys.exit(0)
```