Facial Expression Recognition using Local Binary Patterns

with classification based on Support Vector Machines

AALBORG UNIVERSITY
Department of Electronic Systems
Vision, Graphics and Interactive Systems
9^{th} Semester project

Autumn 2012

Maxime Coupez
Kim-Adeline Miguel
Julia Alexandra Vigo

AALBORG UNIVERSITY

Department of Electronic Systems
Fredrik Bajers Vej 7
DK-9220 Aalborg Ø
http://es.aau.dk

Title:
Project Title

Theme:
Interactive Systems

Project Period:
Fall Semester 2012

Project Group:
12gr942

Participant(s):
Maxime Coupez
Kim-Adeline Miguel
Julia Alexandra Vigo

Supervisor(s):
Zheng-Hua Tan

Copies: 1

Page Numbers: 53

Date of Completion:
November 26, 2012

Abstract:

Since the last decade, a lot of researches have been carried out about emotion recognition. The number of projects conducted in this field demonstrates the interest and the importance of systems which can recognize human mood.
In this project, an emotion recognition system is developed, using a Microsoft Kinect. This recognition is achieved in 3 steps: Face detection, extraction and classification of facial features, this structure being the usual modus operandi in emotion recognition research.
Face detection is performed using Viola-Jones' algorithm, then Local Binary Patterns (LBP) are used to extract facial features. Finally, Support Vector Machines (SVM) classify these features into six predefined emotions.
The system is implemented to run on a computer using a Kinect and works for one person in front of it. The classifier is trained with the Cohn-Kanade database, which includes enough different faces to obtain a satisfying result.

Preface

This report documents the semester project entitled *Facial expression recognition using Local Binary Patterns.* The project was carried out during the 9th semester of specialization *Vision, Graphics, and Interactive Systems* under the Department of Electronic Systems at Aalborg University in Autumn 2012.

The report is divided into four parts plus appendices: *Introduction*, *Feature Detection*, *Feature Classification*, *Implementation* and *Evaluation.* The first part review the general structure of a facial expression recognition system and its main issues, and concludes with a state of the art of existing systems. Analysis of possible solutions and design of our system are contained in the following two parts, and the fourth part describes our implementation. The last part evaluates the performance and accuracy of our system and concludes on the project as a whole.

References to secondary literature sources are made using the syntax [number]. The number refers to the alphabetically sorted bibliography found at the end of the report, just before the appendices.

We would like to thank our supervisor at Aalborg University Zheng-Hua Tan for supporting us in this challenging project.

A CD is attached to this report which includes:

- Source code of the developed program.
- PDF file of this report.

Aalborg University, November 26, 2012

Maxime Coupez
<mcoupe12@es.aau.dk>

Kim-Adeline Miguel
<kmigue12@es.aau.dk>

Julia Alexandra Vigo
<jvigo12@es.aau.dk>

Contents

Part I

Introduction

Contents

The main motive of this project is to understand real-time facial expression recognition systems and their applications. A review of the architecture of such systems will be done, along with a state of the art of already existing algorithms. After this study, issues coming along with this kind of recognition system will be studied. In the last part, the requirements of this project will be formulated.

Chapter 1

Motivations

A facial expression is a visible manifestation of the effective state, cognitive activity, intent, personality, and psychopathology of a person [7]; facial expressions play a significant role in human dialogue and interaction. Indeed, facial expressions carry more informations than mere speech, informations on which humans can relay for interaction. Facial expressions have a considerable effect on a listening interlocutor; a speaker facial expressions accounts for about 55 percent of the effect, 38 percent of the latter is conveyed by voice intonation and 7 percent by the spoken words [17].

Since Antiquity, researchers have been interested in emotion and more particularly in emotion recognition. But one of the important studies on facial expression analysis impacting on the modern day science of automatic facial expression recognition was the work carried out by Charles Darwin [3]. In 1872, Darwin wrote a treatise that established general expression principles and expression means for both humans and animals [5]. He also classified various kinds of expressions. This can be considered as the beginning of facial expression recognition.

Now, with the emergence of new technologies and computers, research is now focused on computer-based automatic facial expression recognition. Because facial expressions are major factors in human interaction, this research field will broaden the domain of Human-Machine Interaction. Indeed, emotion recognition will enable computers to be more responsive to users' emotions, and allow interactions to become more and more realistic.

Another domain where facial expression recognition is an important issue is robotics. With the advances made in robotics, robots nowadays tend to mimic human emotion and react as as human-like as possible, especially for humanoid robots. However, since robots are being more and more present in our daily lives, they need to understand and recognize human emotions.

A lot of real time applications in the robotics field have already been created. For example, Bartlett et al. have successfully used their face expression recognition system to develop an animated character that mirrors the expressions of the user (called CU Animate) [2]. They have also been successful in deploying the recognition system on Sony's Aibo Robot and ATR's RoboVie [2]. Another interesting application has been demonstrated by Anderson and McOwen, called "EmotiChat" [1]. It is a regular chatroom, except the fact that their facial expression recognition system is connected to the chat and convert the users' facial expressions into emoticons. Because facial expression recognition systems' robustness and reliability are constantly increasing, lots of innovative applications will appear.

There are also various other domains where emotion recognition can be used: Telecommunications, behavioural science, video games, animations, psychiatry, automobile safety, affect-sensitive music jukeboxes and televisions, educational software, etc [3].

Our project focuses on real-time facial expression recognition from a video stream. Indeed, facial expression recognition can be performed *statically* on input images, or *dynamically* on video sequences. Systems can also be *obtrusive*, or *non-obtrusive*, the former based on a device mounted on the user's head or body, therefore following each of his movements and perform facial expression recognition without much losses, while the latter can encounter difficulties if the user is not properly situated. However, non-obtrusive systems allow more natural user interactions. We chose our system to be non-obtrusive, and will detail further its setup in the next section.

1.1 Environment Setup

Our system will use the camera embedded into a Microsoft Kinect to record the user's video input, and we will consider a casual use of the camera, with the user sitting in front of the computer, the camera being next to it, as seen in **Insert picture of the setting & ref to figure**. This camera provides a 640×480 pixels frame resolution, while recording at 30 FPS.

For development and training purposes we will use some pre-existing emotion datasets, in order to validate the efficiency of the system before testing it in real conditions.

1.2 Emotion Datasets

Databases are very important for facial expression recognition system.

Using the same databases for studies that aims to improve existing systems is very useful. It allows to compare he results and to see if the new system is indeed better than the existing one. A lot of research studies work is based on the same databases than previous studies in order to compare the efficiency of their algorithm.

But databases are hard to construct. It has to fill all the requirements and that is why most of the work on facial expression recognition is based on existing databases. The hardest requirement to fill is to have a standardized database. Most of the actual databases use posed expressions and not spontaneous expression, and both are very different. New versions of the databases are coming out with spontaneous expressions in order to be more complete. Even with this transition from posed expressions to spontaneous expressions, there are other requirements that should be respected to have a database standardized for training and testing. It should contain images and video sequences and both should be of different resolutions. It should also contain people displaying expressions under different conditions: it could be change in the lighting, occlusions or rotations of the head [3].

Following are the databases that will be used to test this facial expression recognition system. These are part of the databases that are popular, freely available and mostly used in the past few years.

1.2.1 Japanese Female Facial Expression Database (JAFFE)

The database contains 213 images of 7 facial expressions (6 basic facial expressions: happy, angry, afraid, disgusted, sad, surprised + 1 neutral) posed by 10 Japanese female models. Each expression has been photographed three or four times. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. The database was planned and assembled by Miyuki Kamachi, Michael Lyons, and Jiro Gyoba [14].

This database contains only posed expressions. The photos have been taken under strict controlled conditions of similar lighting and with the hair tied away from the face [3].

An example of images contained in the database is given by the figure 1.1. Here the subject is a woman and she displays 7 different emotional expressions (neutral, happy, angry, afraid, disgusted, sad, surprised):

1.2.2 Karolinska Directed Emotional Faces Database (KDEF)

The Karolinska Directed Emotional Faces (KDEF) is a set of totally 4900 pictures of human facial expressions of emotion. The material was developed in 1998 by Daniel Lundqvist, Anders Flykt and Professor Arne Ohman at Karolinska Institutet, Department of Clinical Neuroscience, Section of Psychology, Stockholm, Sweden [12].

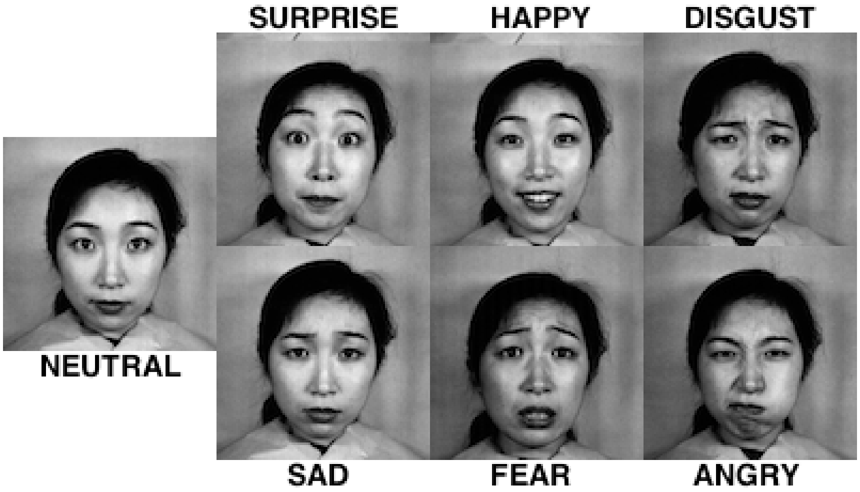

Figure 1.1: example of images from JAFFE database

The material was originally developed to be used for psychological and medical research purposes. More specifically material was made to be particularly suitable for perception, attention, emotion, memory and backward masking experiments. Hence, particular attention was for instance paid to create a soft, even light, shooting expressions in multiple angles, use of uniform T-shirt colors, and use of a grid to center participants face during shooting, and positioning of eyes and mouths in fixed image coordinates during scanning [12].

The set contains 70 individuals (35 males and 35 females), ranging from 20 to 30 years, each displaying 7 different emotional expressions (neutral, happy, angry, afraid, disgusted, sad, surprised), each expression being photographed (twice) from 5 different angles (-90, -45, 0, +45, +90 degrees: i.e. full left profile, half left profile, straight, half right profile, full right profile) [12].

An example of images contained in the database is given by the figure 1.2. Here the subject is a woman photographed from a straight angle and she displays 7 different emotional expressions (neutral, happy, angry, afraid, disgusted, sad, surprised):

1.2.3 Montreal Set of Facial Displays of Emotion Database (MSFDE)

The database consists of emotional facial expressions by men and women of European, Asian, and African descent. Each expression was created using a directed facial action task and all expressions were FCAS coded to assure identical expressions

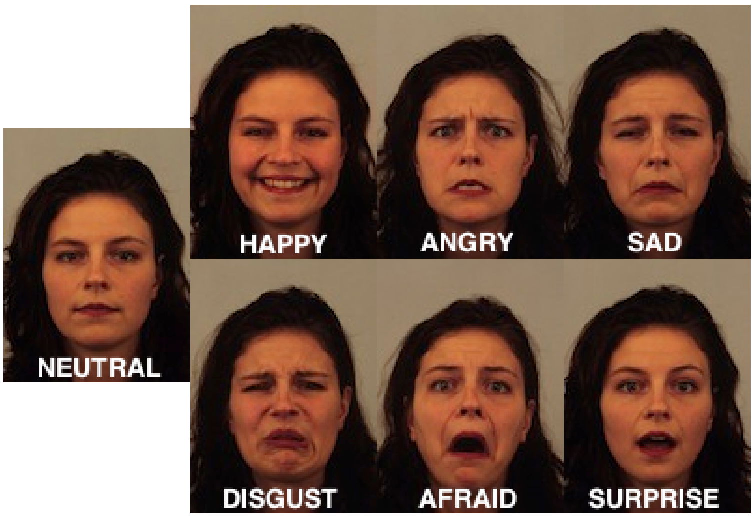

Figure 1.2: example of images from KDEF database

across actors [23].

The set contains expressions of happiness, sadness, anger, fear, disgust, and embarrassment as well as a neutral expression for each actor. All expressions have been morphed into 5 different levels of intensity [23].

An example of images contained in the database is given in the figure 1.3. Here the subject is an african woman and she displays 7 different emotional expressions (neutral, happy, angry, afraid, disgusted, sad, ashamed):

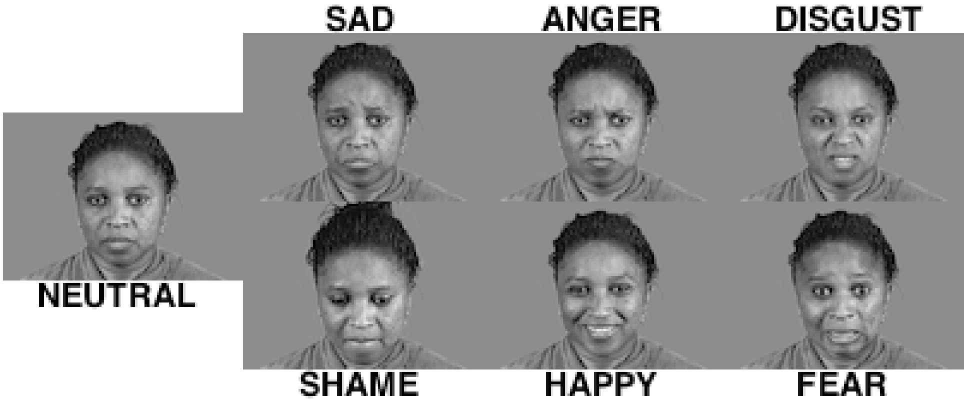

Figure 1.3: example of images from MSDFE database

Chapter 2

Facial expression recognition

After having stated the conditions and motivations of our project, we will now describe an overview of the system we will implement. A facial recognition system can roughly be summed up as classification applied to a pre-processed image. The image processing steps, especially the feature extraction part, along with commonly used classification algorithms, will be detailed further in this chapter. Next sections will be about issues raised facial expression recognition systems, and key requirements these systems have to meet in order to be considered acceptable.

2.1 General structure

Facial expression recognition is a system enabling an automatic recognition of emotions displayed by a human face. Facial expression recognition can be image or video-based; it can also be computed real-time. Most of the time, researchers try to recognize emotions out of images of human faces. This can also be achieved real-time on video streams : While the person displays his/her emotions, the facial expression recognition system analyses the video, and detect in real-time the displayed emotion.

In both cases, facial expression recognition process is structured as in the figure 2.1

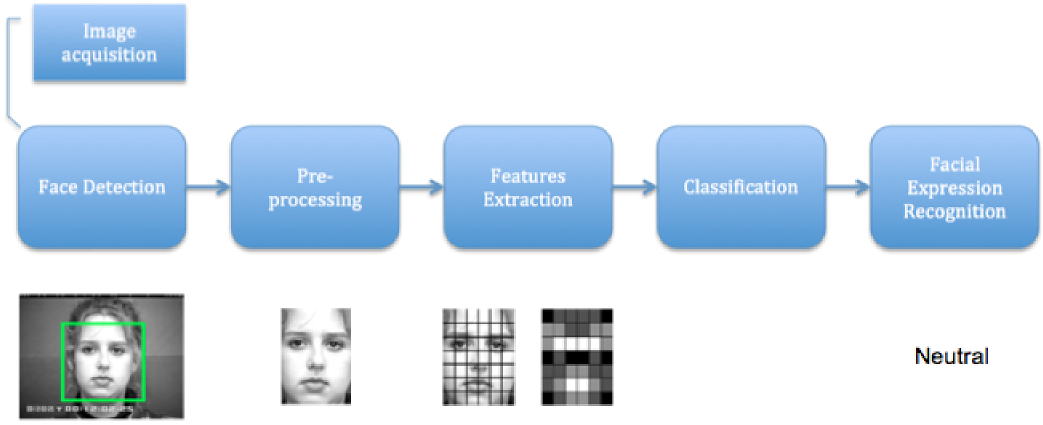

Figure 2.1: Facial Expression Recognition process

2.1.1 Image Acquisition

First step of the process is "Image Acquisition". Images used for facial expression recognition can be static images or image sequences. Image sequences give more informations about the facial expression, as the steps in muscles movement. About static images, facial expression recognition systems usually need 2D greyscale images as inputs. We can however expect future systems to use colour images; first because of the increasing affordability of technologies and devices capable of capturing images or image sequences; then because colours can give more information on emotions, i.e blushing [4].

2.1.2 Face Detection

Second step is "Face Detection". Indeed, in a static image and even more in an images sequence, this is an obvious need. Once the face has been detected, all other non-relevant information can be deleted, since only the face is needed. It could hence be included in the next step, which is "Pre-processing", but because of its importance it represents a step in itself. In a real-time facial expression recognition system working with image sequences, the face has to be detected, but also tracked. One of the most used and famous detection and tracking algorithm is the Viola-Jones

Algorithm, which we will explain in detail later in this report. This algorithm can be trained to detect all kind of objects, but is mostly used for face detection.

2.1.3 Pre-processing

Third step is "Pre-processing", which is about applying image processing algorithms to the image, in order to prepare it for the next step. Pre-processing is usually about noise removal, normalization against the variation of pixel position or brightness, segmentation, location, or tracking of parts of the face. Emotion recognition is also sensitive to transformation, scaling and rotation of the head in the image or image sequence. In order to solve this problem, the image can be geometrically standardized. References used for this standardization are usually the eyes [4].

2.1.4 Features Extraction

Once the image has gone through the "Pre-processing" step, the next one is "Features Extraction". In this step, data is converted into a higher representation of shape, motion, colour, texture, and spatial configuration of the face or its components. One of the main goals of this step is to reduce the dimensionality of the input data. The reduction procedure should retain essential information possessing high discrimination power and high stability [4]. There are a lot of features extraction methods. The most famous are : Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Problem Based Learning (PBL), Hidden Markov Models (HMM), Eigenfaces, Gabor Wavelets. The extracted data is then used in the "Classification" step.

2.1.5 Classification

The classification step marks the end of image processing steps (face detection, normalization, feature extraction). There are many kinds of classification algorithms, some of them can even be used in the feature extraction part, as stated in Section 2.2. This step takes into input a model previously trained with pre-processed data, and test data made of feature vectors extracted from the image we want to label. Feature vectors from pre-processed data and test data have to be obtained using the same feature extraction algorithm. The chosen classifier then outputs a value corresponding to the label of the class the picture belongs to.

2.2 Feature extraction algorithms

Before developing a facial expression recognition project, it is important to know what already exists; the state of the art of facial expression recognition system. In this chapter, an overview will be given of the existing systems before to decide on a system for the project.

2 main categories of feature extraction algorithms can be distinguished : *appearance-based* or *geometry-based*. The first ones are algorithms that try to find basic vectors characterising the whole picture, usually by a dimensionality reduction method. These algorithms lead to a simplification of the dataset, while retaining the main characteristics of the picture. However, these methods have to be carefully parametrized, so they do not encounter the "curse of dimensionality", which is about processing high-dimensional data.

Examples of appearance-based methods : Principal Component Analysis, Linear Discriminant Analysis, Hidden Markov Models, Eigenfaces

The second type of feature extraction algorithms is geometry-based algorithms. These methods tend to locate important features, and build the feature vectors depending on those regions of interest. The key point of these methods is that the face is not a global structure anymore. Indeed, it has been summarized in a set of features regions, which are themselves translated into feature vectors.

Examples of geometry-based methods : Gabor Wavelets, Local Binary Patterns

2.2.1 Principal Component Analysis (PCA)

This is a statistical method; one of the most used in linear algebra. PCA is mainly used to reduce high dimensionality of data and to obtain the most important information from this data. Because Facial Expression Recognition needs to reduce the dimensionality of data during features extraction, PCA is commonly used. It helps transforming high dimensionality of data to a new coordinate system of lower dimensions while still preserving the most important information. PCA computes a covariance matrix and a set of values called the eigenvalues and eigenvectors from the original data [8]. Since it is a statistical method, it can also be used in the classification step.

2.2.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis is also a statistical method, used to classify a set of objects into groups. It is done by observing a set of features that describe the objects. LDA as PCA are used to establish a linear relationship between the dimensions of the data. The main difference is that LDA uses the linear relationship to model the differences into classes of objects and PCA does not take any differences into account in the linear relationship. The idea is to perform a linear transformation on the data to obtain a lower dimensional set of features [8]. Like PCA, LDA is also a classification algorithm.

2.2.3 Local Binary Patterns (LBP)

This is an appearance-based method. It can be used to describe texture and shape. LBP extracts some informations from the neighbourhood of a central pixel. It compares the intensity values of the neighbourhood pixels with the intensity value of the central pixel [8]. This method is the one that will be used for this Facial Expression Recognition system.

2.2.4 Hidden Markov Models (HMM)

These models are a set of statistical models used to characterize the statistical properties of a signal [19]. It can be used as a classification algorithm, and canalso be developed to recognize expressions based on the maximum likelihood decision criterion [13].

2.2.5 Eigenfaces

Eigenfaces are a set of eigenvectors which are derived from the covariance matrix of a set of face images in a high-dimensional vector space. The eigenvectors are ordered and each one represents the different amount of the variation among the face images. It all together characterizes the variation between face images [22].

2.2.6 Gabor Filters

Gabor filters are applied in order to extract a set of Gabor wavelet coefficients. When convolving these Gabor filters with a simple face image, filter responses are obtained. These representations display desirable locality and orientation performance [11]. However, it has a limitation which is the processing time of Gabor feature extraction. It is very long and its dimension is prohibitively large [18].

2.3 Issues

2.3.1 Database

Databases can be a source of issues. As said previously, databases should fulfill a number of requirements in order to be the most efficient as possible.

If the Facial Expression Recognition system wants to be close to reality, it should be able to recognize spontaneous expressions rather than posed expressions. Indeed, spontaneous expressions are closer to reality than posed expressions. Posed expressions are exaggerated. While creating a database of spontaneous expressions, Sebe and colleagues [20] made some observations of the major problems they encountered [3]:

- Different subjects express the same emotions at different intensities
- If the subject becomes aware that he or she is being photographed, their expression loses its authenticity
- Even if the subject is not aware of the recording, the laboratory conditions may not encourage the subject to display spontaneous expressions.

In order to get round of these problems, they came up with a method. The method was to record facial expressions with a camera hidden in a video kiosk. The video kiosk was displaying emotion inducing videos. Once the recording was done, subjects were notified of the recording and were asked for their permission to use the captured images and videos for research studies. Then the subjects explained which emotions they felt and expressed and their replies were documented against the recordings of the facial expressions [20].

They found that a wide range of expressions are hard to induce and particularly fear and sadness. They also found that spontaneous expressions could be misleading: some subjects express one emotion while feeling another one (for example, one subject was showing sadness while being happy) [20].

At the end, databases bring some issues that can affect the authenticity of the recognition system. It depends of the type of the expressions : spontaneous or posed expressions. If the system aims to recognize facial expressions of people unaware of it, spontaneous expressions databases will be used but it leads to authenticity issues as seen previously. If the system aims to recognize facial expressions of people asked to express certain emotion, posed expressions databases will be used but the result will not be close to the reality.

2.3.2 Real-time

The goal of the Facial Expression Recognition system of this paper is to recognize facial expression in real time. For a real-time application, the processing should not be too heavy otherwise the time of processing will be too long and the application will not be in real time anymore.

This is one of the challenges of this kind of system. Because most of the time the processing is really heavy whatever the algorithm is and it is really difficult to make it work in real-time. Most of the applications in need of Facial Fxpression Recognition are in need of real-time recognition. For example in robotics, or in surveillance. The solution could be to find new algorithms for Facial Expression Recognition or to improve and lighten already existing algorithms.

2.3.3 Conditions

Another one of the challenges of this kind of system is to be independent to the conditions of the recording. It means that the recognition should not be disturbed by occlusions for example, or difference in the lighting, or even by the angle that the face makes with the camera lens. This examples cover almost all the conditions that can change during the recording and have an influence on the recognition system.

Occlusion

"Occlusion" represents all the elements that can cover the face or a part of it. For example, a beard, a scarf masking the bottom of the face, glasses or bangs. By hiding a part of the face, these occlusions can affect the recognition. Indeed, these Facial Expression Recognition systems are based on comparison of features and if all the features cannot be compared because something is covering a part of the face, the recognition is affected. In order to compensate for this problem, some databases includes data with already occlusions in it as beard, glasses or scarf for example. This is the case for the AR Face database and some examples of the images contained in this database are given by figure 2.2 and figure 2.3 [15]:

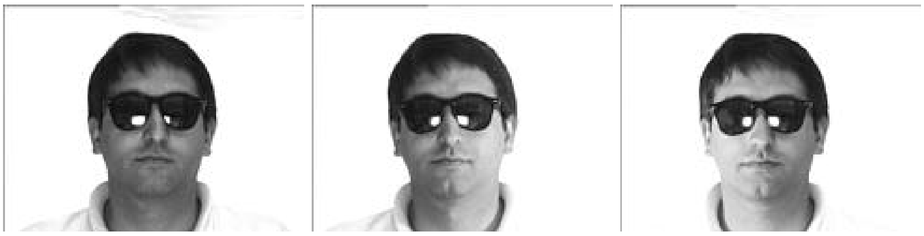

Figure 2.2: example of occlusion by sunglasses in the AR Face database

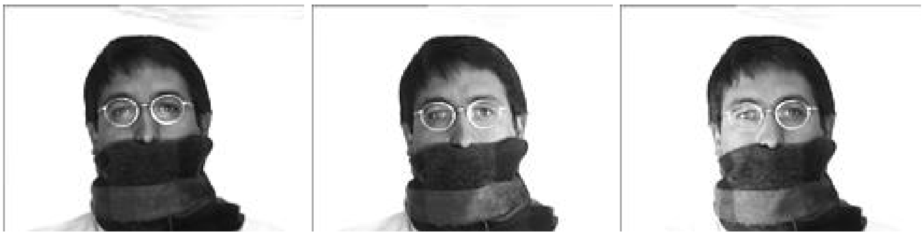

Figure 2.3: example of occlusion by scarf and glasses in the AR Face database

Lightning

As for occlusion, lightning is an element that can affect Facial Expression Recognition system. With a lightning different from the one used in the database, the recognition will be less efficient. All the conditions that change from the one of the database on which is based the recognition process will disturb the recognition in itself. If the images are brighter or darker than the one of the database, some details can disappear or some features cannot be recognized as well as if the conditions were the same. In order to compensate for this problem, some databases includes data with already changes in the lightning. This is the case for the AR Face database and an example of the images contained in this database is given by the figure 2.4 [15]:

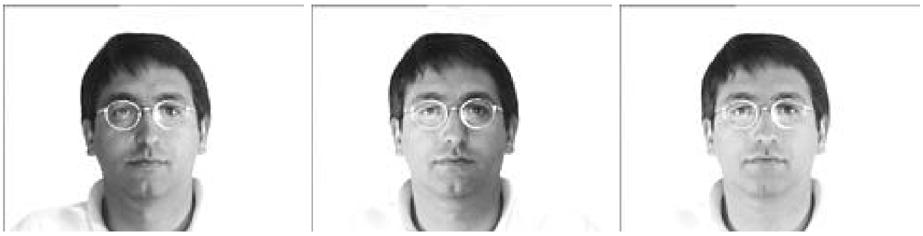

Figure 2.4: example of different lightnings (from left to right: dark to bright) in the AR Face database

Angle

The angle of the head from the camera lens is one of the conditions that can affect the most the recognition process. Indeed if the head is too much turned from straight angle, some features will disappear. For example, with a profile angle, one eyes and half of the nose and of the mouth disappear. And if the database does not contain sample with profile face, the recognition will not succeed. This constraint is based on the database; if the database contains images from straight angle as well as from profile angle and other, the recognition will be possible even if the head is not with a straight angle. But if the database contains images only from straight angle, the recognition will be almost impossible if the head is turned. An example of the different images taken from different angles for one emotion, "Fear", from the KDEF database is given in figure 2.5:

2.4 Requirements

Based on everything that was said previously, the Facial Exression Recognition system of this paper can be defined by some requirements. Additional requirements may be defined further in this paper. Here are the requirements already defined :

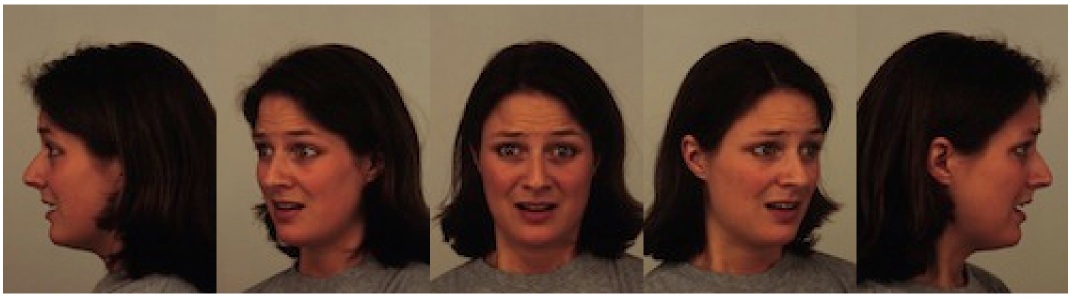

Figure 2.5: example of different angles of the "Fear" emotion (from left to right: full left profile, half left profile, straight, half right profile, full right profile) in the KDEF database

- Able to recognize basic emotions : As explained before, Facial Expression Recognition systems are able to recognize 6 basic emotions and the neutral state. This system would be able to do the same: recognize the 6 basic emotion that are "Happiness", "Fear", "Surprise", "Disgust", "Sadness", "Anger" and the neutral state.

- Able to work in real-time : This system would be able to recognize emotion in real-time. It means that it could recognize expressions based on a video sequences . It also means that the algorithm for the feature extraction could not have heavy processing or to use one that would be lightened.

- Recognition from straight angle of the face : This system would be able to recognize facial expression from a straight angle of the face. It means that the system would be able to detect faces that are in front of the camera lens and to recognize expressions in these faces. It would not be able to recognize emotions on a face that is from profile or half-profile.

- Recognition with no occlusion on the face : This system would be able to recognize emotions with no occlusion on the subject's face. It means that the face would not be cover in any way: no glasses, no beard or no scarf. The face would be complete and not masked.

- Recognition with no change in lightning : This system would be able to recognize emotions on faces under the same lighting conditions. It means that the light would not vary during the recognition part. The light would stay the same and the level of intensity of the light would be as close as possible that the one of the database. This way the lightning would not have any influence on the recognition.

Part II

Feature detection

Contents

Before getting to the main part of this project that is Feature extraction, there is a mandatory step that is Feature detection. In order to avoid the more computation and processing as possible, only the parts that contains the regions of interest for a Facial Expression Recognition system have to be processed. It means by consequence that there has to be beforehand the detection of the interesting features. This detection can be summarized into face detection. This part will explain how face detection works in general. Then the most famous and efficient algorithm for face detection will be introduced and studied. This algorithm is the Viola-Jones algorithm.

Chapter 3

Face detection

Face detection is the first step after image acquisition. It represents a requirement for a Facial Expression Recognition system. All the background is not taken into account. It allows to focus only on what is interesting in the input image: the face, which helps reducing the processing during the next step that is feature extraction.

3.1 Detection

Finding out whether or not the input image or video sequence represents or contains a particular object is what is called Detection. Usually after the detection step comes the recognition step. For this system, the recognition step consists in facial expression recognition. But depending on the recognition, there can be another step that is the tracking step. Tracking consists in following a moving target on the images of a video sequence [6].

At a high level, an object detector is a "black box" which gets an image as its input, and gives an annotated image as its output, saying where the object of interest appears in the input image (if it appears...) and the extension of each detected instance of the object [6]. For example, the output can look like in figure 3.1 [6].

But at a low level, the basic component of an object detector is just something required to say if a certain sub-region of the original image contains an instance of the object of interest or not. That is what a binary classifier does [6]. For example, what a binary classifier does can look like in figure 3.2 [6].

3.2 Classifiers

Classification aim to solve the problem of identifying in a set of categories or subpopulations to which a new observation belongs. It is based on a training set of data that contains instances whose category membership is known. An algorithm that implements classification is known as a classifier. Classifiers groups data into

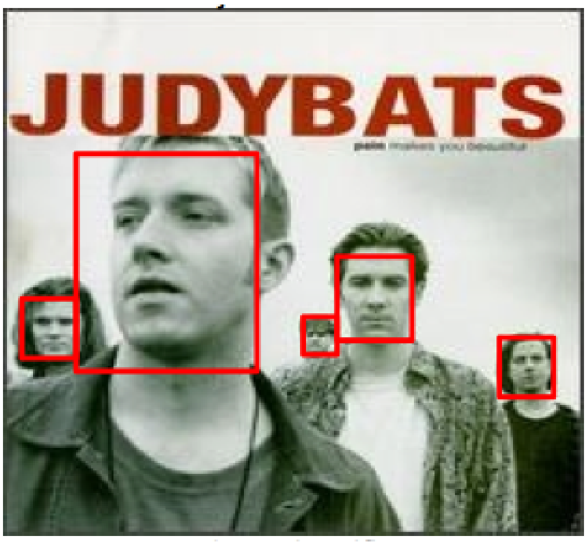

Figure 3.1: Example of an output of face detection

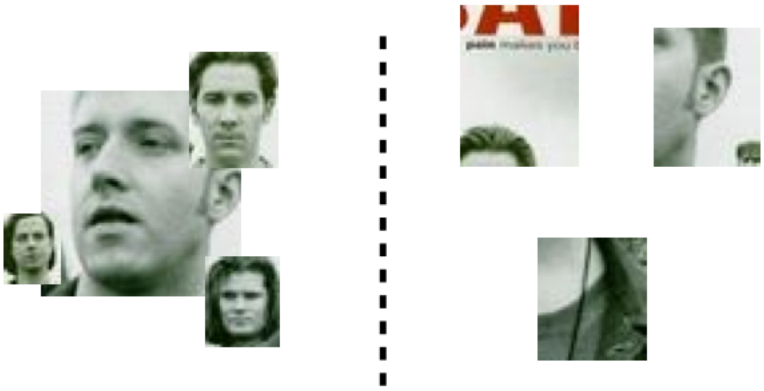

Figure 3.2: Example of what does a binary classifier for face detection

categories based on some measure of inherent similarity; for example, the distance between instances, considered as vectors in a multidimensional vector space [25].

Chapter 4

Viola-Jones

Viola-Jones algorithm is a visual object detection framework that is capable of processing images extremely rapidly while achieving high detection rates. It is based on a new image representation called "Integral Image" which allows the features used to be computed very quickly. It is also based on a learning algorithm, based on AdaBoost that gives extremely efficient classifiers. And it is also based on a method for combining classifiers in "cascade"; it allows to discard quickly the background of the image and to focus on the promising object-like regions [24].

4.1 Overview

The Viola-Jones algorithm works as following [6]:

- The Viola-Jones detector is a strong, binary classifier build of several weak detectors
- Each weak detector is an extremely simple binary classifier
- During the learning stage, a cascade of weak detectors is trained so as to gain the desired hit rate / miss rate (or precision / recall) using AdaBoost
- To detect objects, the original image is partitioned in several rectangular subregions, each of which is submitted to the cascade
- if a rectangular image sub-region passes through all of the cascade stages, then it is classified as "positive"
- The process is repeated at different scales

4.2 Haar features

The features used by Viola and Jones are called Haar features and are based on Haar wavelets. Haar wavelets are single wavelength square waves (one high interval and one low interval). In two dimensions, a square wave is a pair of adjacent rectangles:

one light and one dark. The actual rectangle combinations used for visual object detection are not true Haar wavelets. Instead, they contain rectangle combinations better suited to visual recognition tasks. That is because of this difference that these features are called Haar features, or Haarlike features, rather than Haar wavelets [10].

For example, the figure 4.1 shows the first two Haar features in the original Viola-Jones cascade [10]. These are all features from the original set of features. In figure 4.2, there is an example of features from the extended set of features [6]. In the figure 4.3 , it is an example of a early stage in the Haar cascade. Each black and white patch represents a feature that the algorithm hunts for in the image [9].

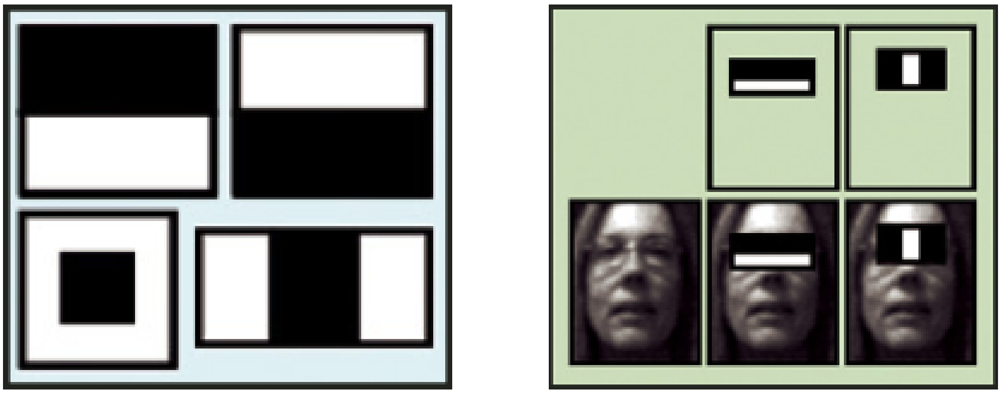

Figure 4.1: Example of the first two Haar features

The presence of a Haar feature is determined by subtracting the average dark-region pixel value from the average light-region pixel value. if the difference is above a threshold, that feature is said to be present and then it can go on to the next stage [10]. There is about 20-30 different stages. The first stage is a very coarse scan of the image. Stage 2 gets a little more detailed, stage 3 is a harder test to pass, stage 4 is even harder and it goes on and on. More it goes further into the cascade, more the features get increasingly complex and larger. It also takes more time to compute [9].

For example, the figure 4.4 shows the later stage in the Haar cascade where many more patterns of black and white rectangles need to match the candidate image [9].

Three kinds of feature are used by Viola and Jones. The value of a two-rectangle feature is the difference between the sum of the pixels within two rectangular regions. The regions have the same size and shape and are horizontally or vertically adjacent. A three-rectangle feature computes the sum within two outside rectangles subtracted

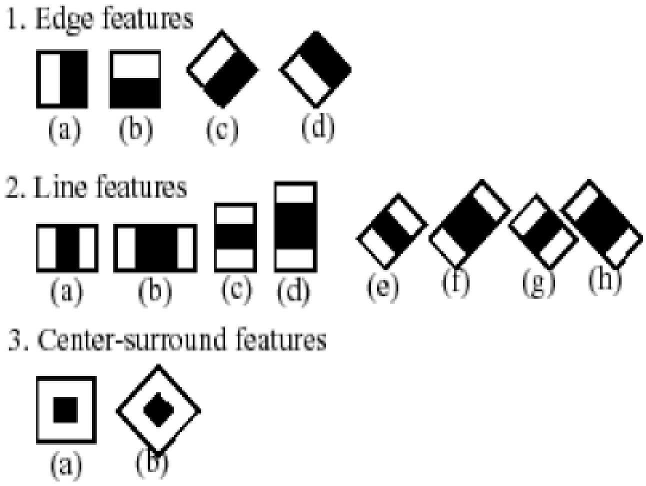

Figure 4.2: Extended set of features

from the sum in a center rectangle. Finally a four-rectangle feature computes the difference between diagonal pairs of rectangles [24].

For example, the figure 4.5 shows the different kinds of rectangle features used by the Viola-Jones algorithm. Two-rectangle features are shown in (A) and (B). Figure (C) shows a three-rectangle feature, and (D) a four-rectangle feature [24].

Rectangle features can be considered as somewhat primitive. In contrast with other features, rectangle features, while sensitive to the presence of edges, bars and other simple image structure, are quite coarse. It appears as though the set of rectangle features do however provide a rich image representation which supports effective learning. The extreme computational efficiency of rectangle features provides ample compensation for their limited flexibility [24].

4.3 Integral image

Rectangle features can be computed very rapidly using an intermediate representation for the image which Viola and Jones called the "integral image" [24]. This integral image technique allows to determine the presence or absence of hundreds of Haar features at every image location and at several scales efficiently. In general, "integrating" means adding small units together; here, the small units are pixel values. The integral value for each pixel is the sum of all the pixels above it and to its

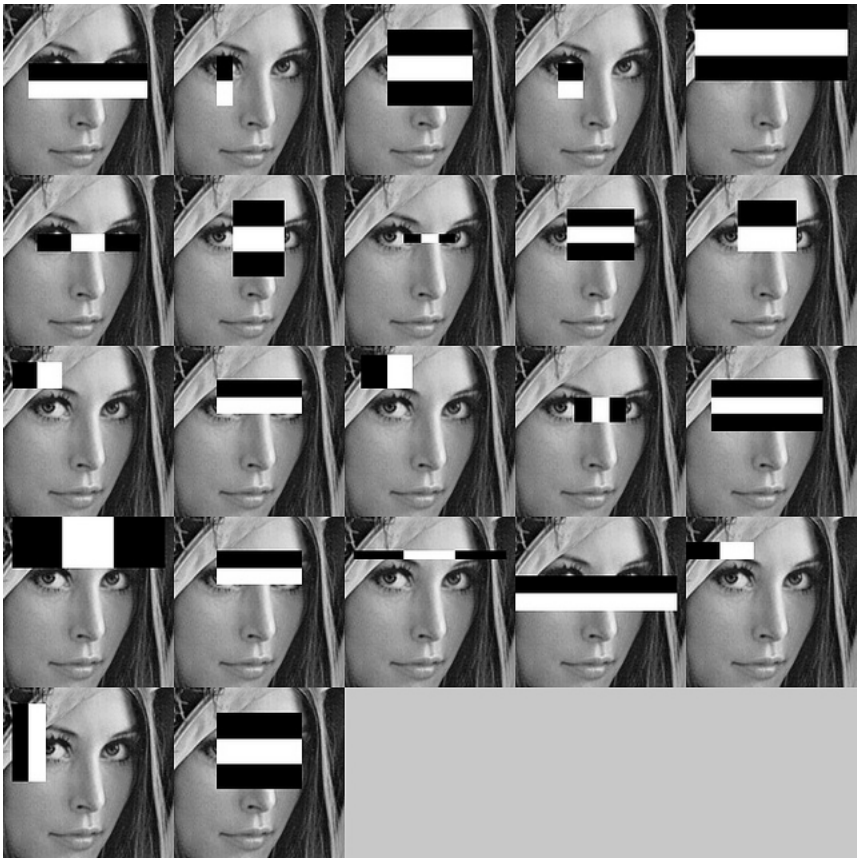

Figure 4.3: Example of an early stage in the Haar cascade

left. Starting at the left and traversing to the right and down, the entire image can be integrated with a few integer operations per pixel [10].

It means that the integral image, at location x, y contains the sum of the pixels above and to the left of x, y, inclusive:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

where $ii(x, y)$ is the integral image and $i(x, y)$ is the regional image. In the figure 4.6, the value of the integral image at point (x, y) is the sum of all the pixels above and to the left [24].

Using the following pair of recurrences:

$$s(x, y) = s(x, y - 1) + i(x, y) \tag{4.1}$$

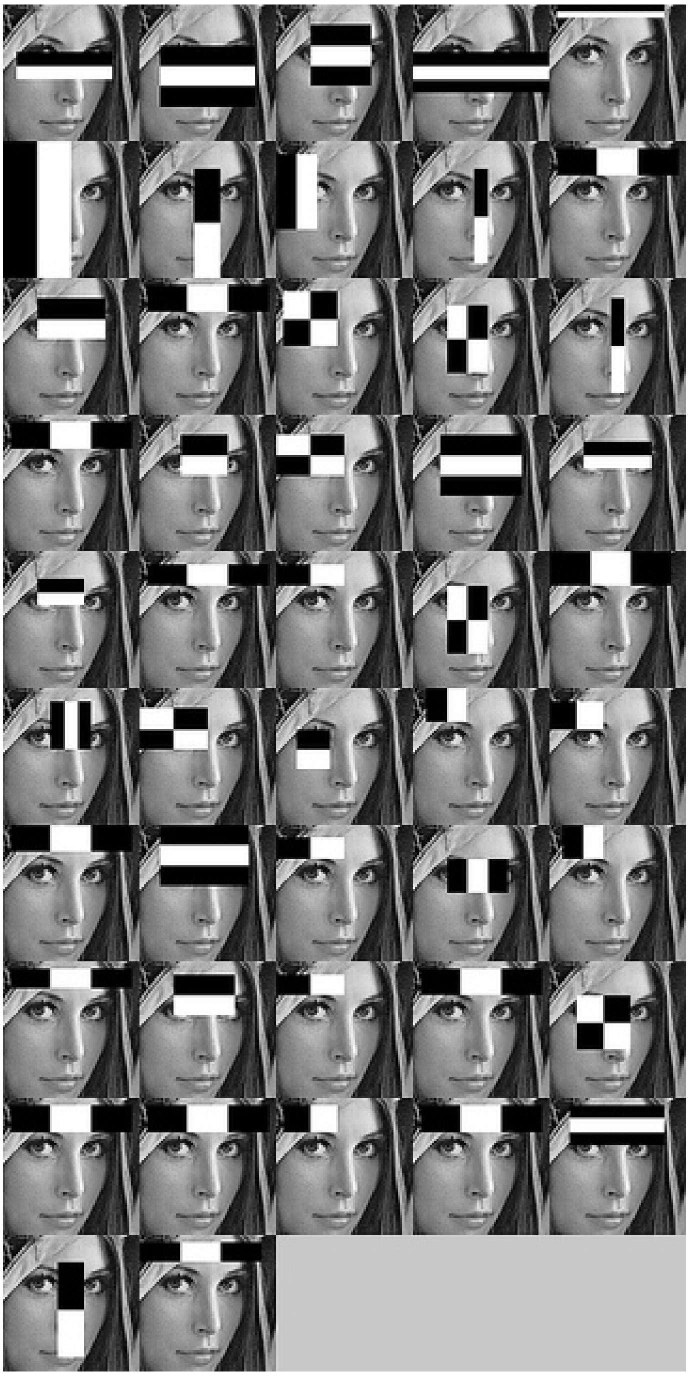

Figure 4.4: Example of the later stage in the Haar cascade

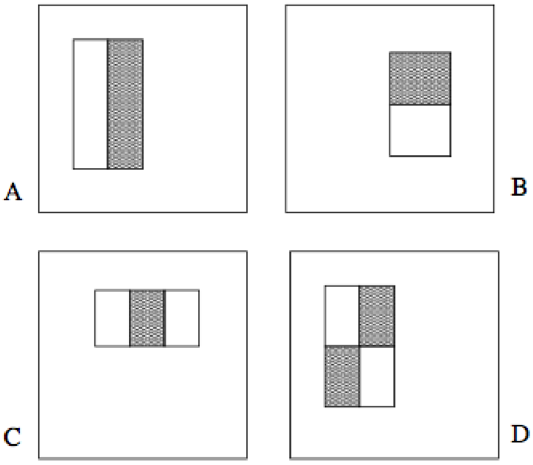

Figure 4.5: Example of the different kinds of rectangle features

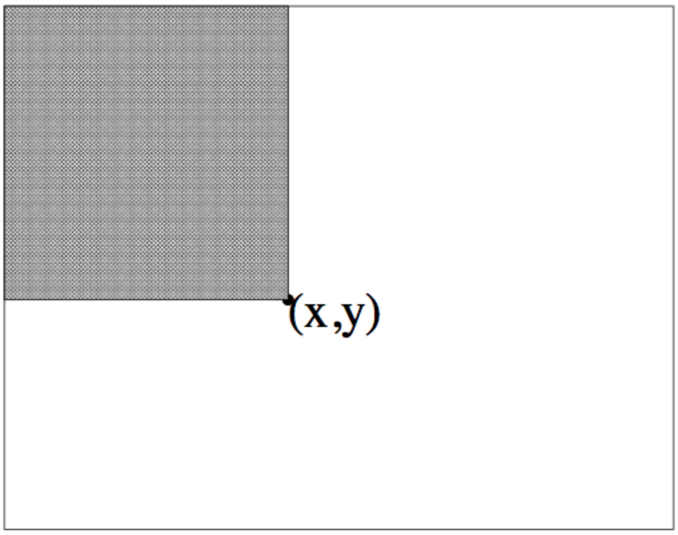

Figure 4.6: Integral image

$$ii(x,y) = ii(x-1,y) + s(x,y) \tag{4.2}$$

(where $s(x,y)$ is the cumulative row sum, $s(x,-1) = 0$, and $ii(-1,y) = 0$) the integral image can be computed in one pass over the original image. Using the integral image any rectangular sum can be computed in four array references. In the figure 4.7, the sum of the pixels within rectangle D can be computed with four array references. The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is $A + B$, a location 3 is $A + C$, and at location 4 is $A+B+C+D$. The sum within D can be computed as $4+1-(2+3)$ [24].

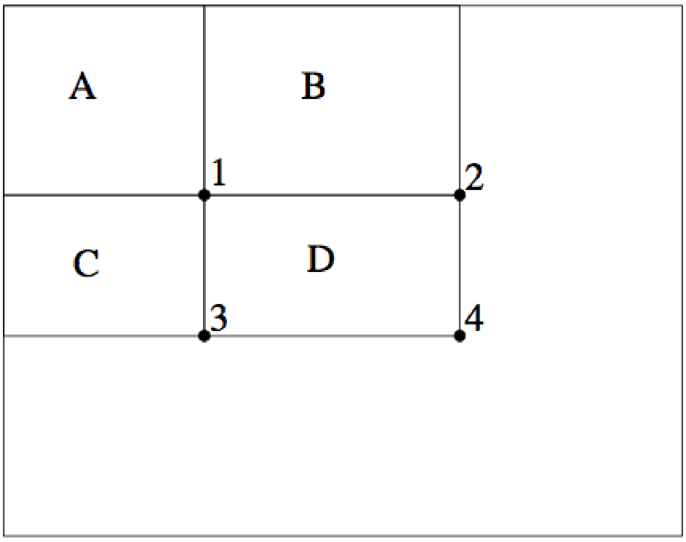

Figure 4.7: Integral image with four array references

The integral is in fact the double integral of the image (first along rows and then along columns). The second derivative of the rectangle (first in row then in column) yields four delta functions at the corners of the rectangle [24].

4.4 Weak classifiers and AdaBoost

Features are extracted from a sub windows of a sample image. The base size for this sub window is 24 by 24 pixels. Each of all the features types are scaled and shifted across all possible combinations (In a 24 pixel by 24 pixel sub window, there are about 160,000 possible features to be calculated) [21].

To select the specific Haar features to use, and to set threshold levels, Viola and Jones use a machine-learning method called AdaBoost. AdaBoost combines many "weak" classifiers to create one "strong" classifier. "Weak" here means that the classifier only gets the right answer a little more often than random guessing would. That is not very good. That is why a lot of weak classifiers are used. If a whole lot of these weak classifiers are used, and each one "pushed" the final answer a little bit in the right direction, this represents a strong, combined for to arrive at the correct solution.

AdaBoost selects a set of weak classifiers to combine and assigns a weight to each (see figure 4.8). This weighted combination is the strong classifier [10]. The challenge is to associate a large weight with each good classification function and a smaller weight with poor functions. AdaBoost is an aggressive mechanism for selecting a small set of good classification functions which nevertheless have signficant variety [24].

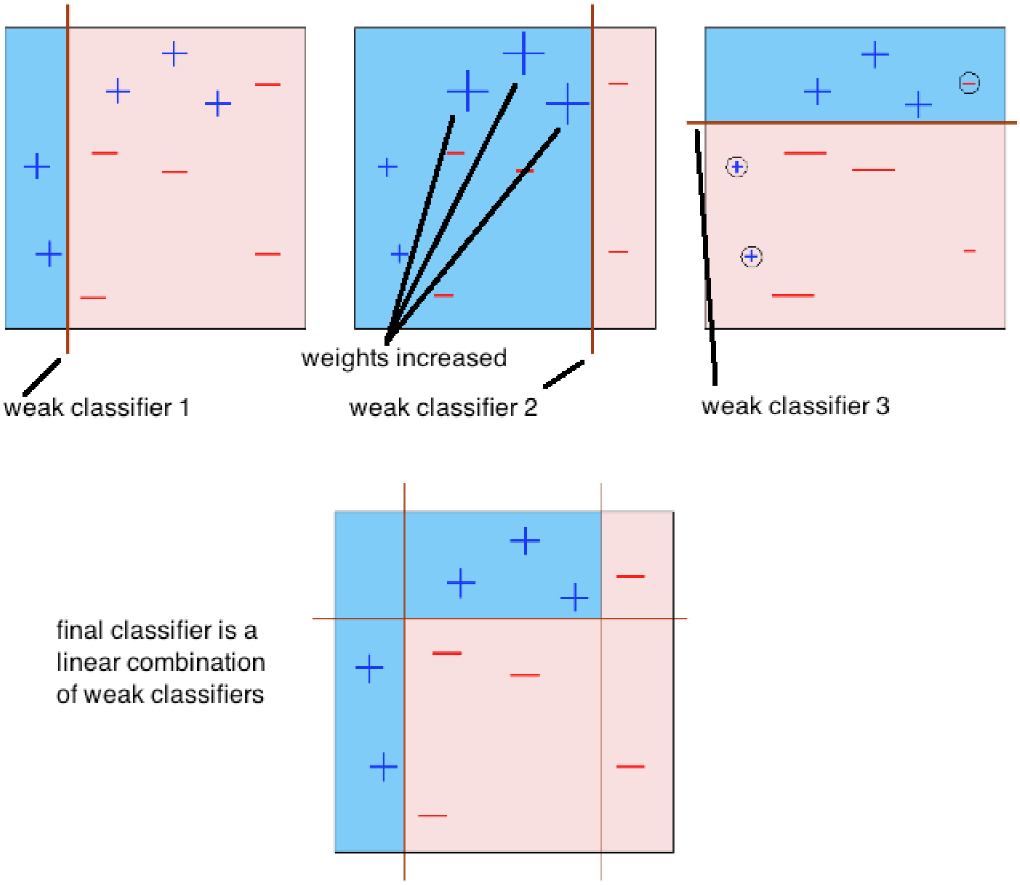

Figure 4.8: AdaBoost method

Initial experiments demonstrated that a classifier constructed from 200 features using AdaBoost would yields reasonable results. Given a detection rate of 95%, the classifier yielded a false positive rate of 1 in 14084 on a testing dataset (see figure 4.9)[24].

The 200-feature classifier provides initial evidence that a boosted classifier con-

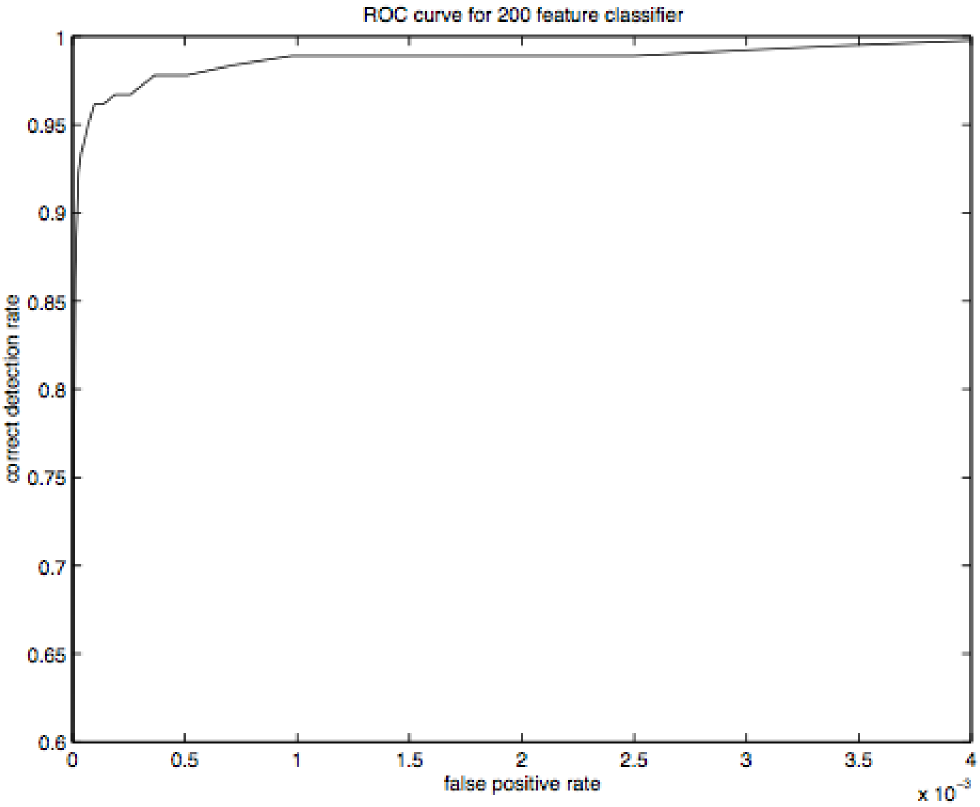

Figure 4.9: Receiver operating characteristic (ROC) curve for the 200 feature classifier

structed from rectangle features is an effective technique for object detection. In terms of detection, these results are compelling but not sufficient for many real-world tasks. In terms of computation, this classifier is probably faster than any other published system, requiring 0.7 seconds to scan an 384 by 288 pixel image. Unfortunately, the most straightforward technique for improving detection performance, adding features to the classifier, directly increases computation time [24].

4.5 Classifiers cascade

Viola and Jones combined a series of AdaBoost classifiers as a filter chain, that is especially efficient for classifying image regions. Each filter is a separate AdaBoost classifier with fairly small number of weak classifiers. As in figure 4.10, the classifier cascade is a chain of filters. Image subregions that make it through the entire cascade are classified as "Face". All others are classified as "Not Face" [10]. This algorithm for constructing a cascade of classifiers achieves increased detection performance while radically reducing computation time [24].

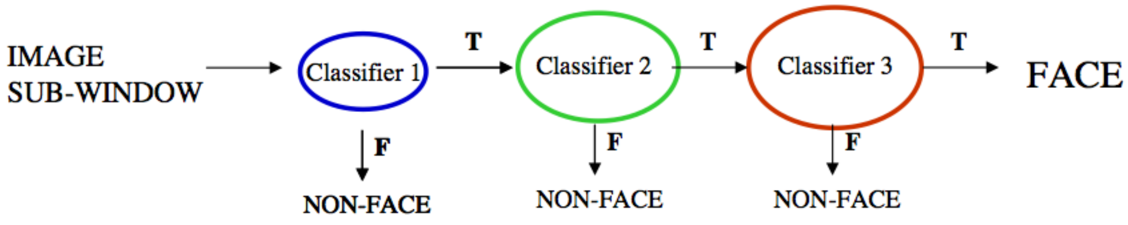

Figure 4.10: Cascade of boosted classifiers

In order to explore the feasibility of the cascade approach two simple detectors were trained: a monolithic 200-feature classifier and a cascade of ten 20-feature classifiers. Figure 4.11 gives the ROC curves comparing the performance of the two classifiers. It shows that there is little difference between the two in terms of accuracy. However, there is a big difference in terms of speed. The cascaded classifier is nearly 10 times faster since its first stage throws out most non-faces so that they are never evaluated by subsequent stage [24].

The acceptance threshold at each level is set low enough to pass all, or nearly all, face examples in the training set. The filters at each level are trained to classifiy training images that passed all previous stages. During use, if anyone of these filters fails to pass an image region, that region is immediately classified as "Not Face". When a filter passes an image region, it goes to the next filter in the chain. Image regions that pass through all filters in the chain are classified as "Face". Viola and Jones named this filtering chain a cascade [10].

The key insight is that smaller, and therefore more efficient, boosted classifiers can be constructed which reject many of the negative sub-windows while detecting almost all positive instances. Simpler classifiers are used to reject the majority of sub-windows before more complex classifiers are called upon to achieve low false positive rates [24].

The order of filters in the cascade is based on the importance weighting that AdaBoost assigns. The more heavily weighted filters come first, to eliminate non-face image regions as quickly as possible. In figure 4.12, it shows the first two features from the original Viola-Jones cascade superimposed on a face. The first one keys off the cheek area being lighter than the eye region. The second uses the fact that the bridge of the nose is lighter than the eyes [10].

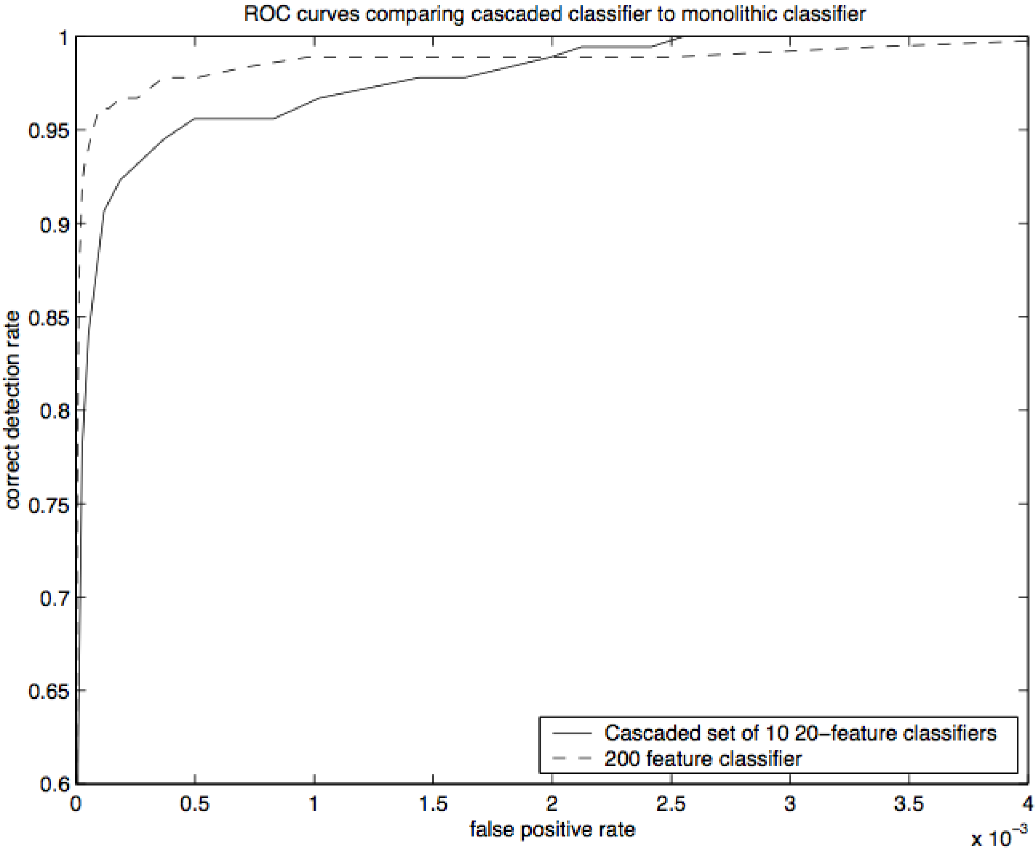

Figure 4.11: ROC curves comparing a 200-feature classifier with a cascaded classifier containing ten 20-feature classifiers

The structure of the cascade reflects the fact that within any single image an overwhelming majority of sub-windows are negative. As such, the cascade attempts to reject as many negatives as possible at the earliest stage possible. While a positive instance will trigger the evaluation of every classifier in the cascade, this is an exceedingly rare event [24].

Following are the different numbers about cascade classifiers (see figure 4.13) [16]:

- A 1 feature classifier achieves 100% detection rate and about 50% false positive rate
- A 5 feature classifier achieves 100% detection rate and about 40% false positive rate (20% cumulative)

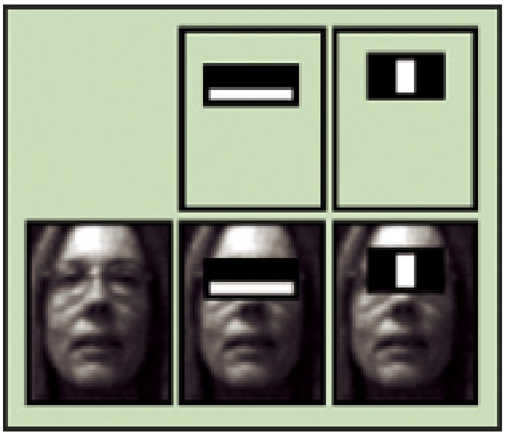

Figure 4.12: The first two Haar features in the original Viola-Jones cascade

- A 20 feature classifier achieves 100% detection rate and about 10% false positive rate (2% cumulative)

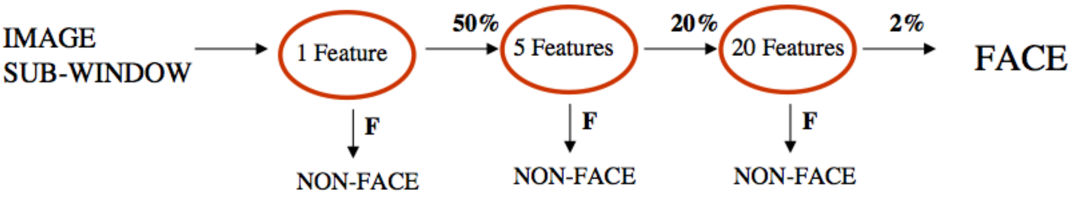

Figure 4.13: Cascade of boosted classifiers rate

4.6 Test set and training

The face training set consisted of 4916 hand labeled faces scaled and aligned to a base resolution of 24 by 24 pixels. The faces were extracted from images downloaded during a random crawl of the world wide web. Some typical face examples are shown in figure 4.14 [24].

The training set is composed of [16]:

- 5,000 faces

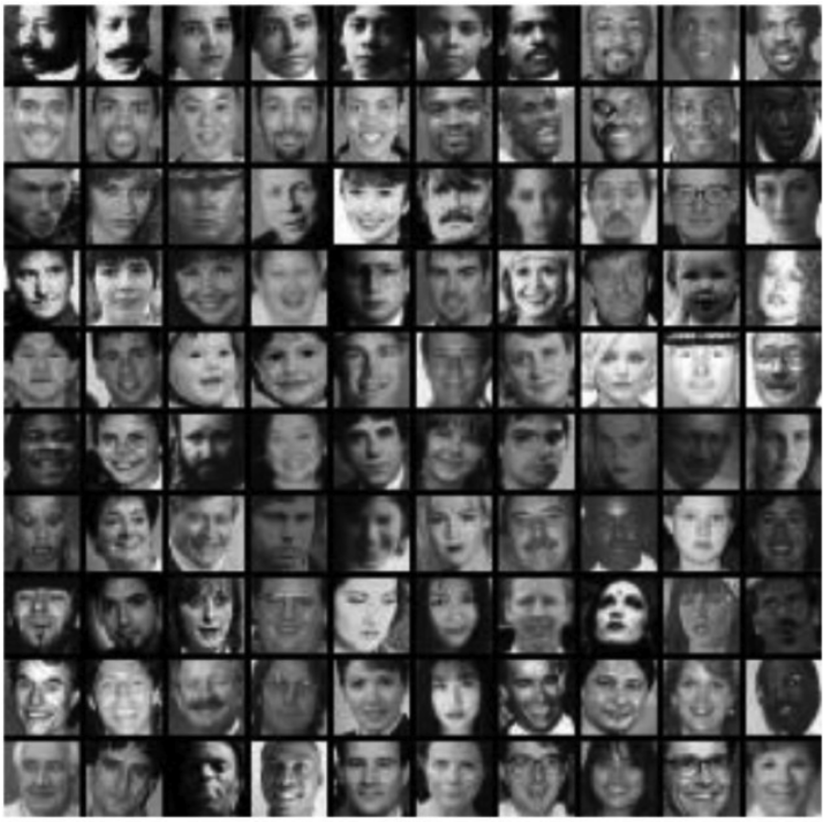

Figure 4.14: Example of frontal upright face images used for training

  - All frontal
- 300 million non faces
  - 9,400 non-face images
- Face are normalized
  - Scale, translation
- Many variations
  - Across individuals
  - Lightning
  - Pose (rotation both in plane and out)

The test set is usually divided into a training set and a validation set. A typical training set may contain about 5,000 positive samples (faces) and 10,000 negative samples (non-face sub-windows randomly chosen from non-face images) [6]. Training time for an entire 32 layer detector is on the order of weeks [24].

Viola-Jones training stage proceeds with the following step [6]:

- Given the number K of possible features (about 160,000 on a 24 × 24 gray-level image)
- Fix the number L of desired stages in the cascade
- Iterate until L weak classifiers have been selected:
  - Given reweighed data from the previous stage
  - Train all K weak classifiers (find the best threshold to classify the training set)
  - Select the best classifier at this stage
  - Reweight the data

The weak classifiers are associated to weights thet depends on their classification error. Those weights are used in a linear combination of the weak classifiers which represents a huge computational cost [6].

Part III

Feature extraction and classification

Contents

This part will focus on the steps following face detection, which are feature extraction and classification. Chapter 5 will present the main issues on feature extraction, and usual feature extraction methods, while the next chapter will focus on the Local Binary Patterns algorithm. Classification will then be introduced in Chapter 7, with a general overview of the classification problem, along with a presentation of some classification algorithms. The last chapter will describe Support Vector Machine classification.

Chapter 5

Feature extraction

bla

5.1 Overview

5.2 Appearance-based methods

bla

5.3 Geometry-based methods

bla

Chapter 6

Local Binary Patterns

bla

6.1 Overview

bla

6.2 Histogram computing

bla

6.3 Improvements

bla

6.3.1 Circular LBP

bla

6.3.2 Uniform LBP

bla

Chapter 7

Feature classification

bla

7.1 Supervised and unsupervised learning

bla

7.1.1 Supervised learning

bla

7.1.2 Unsupervised learning

bla

Chapter 8

Support Vector Machine

bla

8.1 Overview

bla

8.1.1 Margin maximization

bla

8.1.2 SVM kernels

bla

8.2 Combining LBP and SVM

bla

Part IV

Implementation

Contents

Bla bla bla

Part V

Evaluation

Contents

Bla bla bla

Conclusion

In case you have questions, comments, suggestions or have found a bug, please do not hesitate to contact me. You can find my contact details below.

Jesper Kjær Nielsen
jkn@es.aau.dk
http://kom.aau.dk/~jkn
Niels Jernes Vej 12, A6-302
9220 Aalborg Ø

Bibliography

[1] Keith Anderson and Peter W. McOwan. A real-time automated system for recognition of human facial expressions. *IEEE Trans. Systems, Man, and Cybernetics Part B*, 36(1):96–105, 2006.

[2] Marian Stewart Bartlett, Gwen Littlewort, Ian Fasel, and Javier R. Movellan. Real time face detection and facial expression recognition: Development and application to human computer interaction. *Proc. CVPR Workshop on Computer Vision and Pattern Recognition for Human-Computer Interaction*, 5:53, 2003.

[3] Vinay Bettadapura. Face expression recognition and analysis: The state of the art. *Tech Report*, 2012.

[4] Claude C. Chibelushi and Fabrice Bourel. Facial expression recognition: A brief tutorial overview, 2003.

[5] Charles Darwin. *The Expression of the Emotions in Man and Animals.* John Murray, 2. ed. edition, 1904.

[6] Fabrizio Dini. An application of viola-jones algorithm: face detection and tracking. `http://www.micc.unifi.it/dini/download/dbmm2008-Dini.pdf`, 2008.

[7] Gianluca Donato, Marian Stewart Bartlett, Joseh C. Hager, Paul Ekman, and Terrence J. Sejnowski. Classifying facial actions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(10):974–989, 1999.

[8] Abhiram Ganesh. Evaluation of appearance based methods for facial expression recognition, 2008.

[9] Adam Harvey. Adam harvey explains viola-jones face detection. `http://www.cognotics.com/opencv/servo_2007_series/part_2/sidebar.html`, 2012.

[10] Robin Hewitt. How face detection works. `http://www.cognotics.com/opencv/servo_2007_series/part_2/sidebar.html`, 2007.

[11] Yousra Ben Jemaa and Sana Khanfir. Automatic local gabor features extraction for face recognition. *International Journal of Computer Science and Information Security*, 3(1), 2009.

[12] Emotion Lab. Karolinska directed emotional faces (kdef). `http://www.emotionlab.se/resources/kdef`.

[13] Jenn-Jier James Lien. Automatic recognition of facial expressions using hidden markov models and estimation of expression intensity, 1998.

[14] Michael Lyons. The japanese female facial expression (jaffe) database. http://www.kasrl.org/jaffe.html.

[15] Aleix Martinez and Robert Benavente. The ar face database. http://www-sipl.technion.ac.il/new/DataBases/Aleix%20Face%20Database.htm.

[16] University of British Columbia. The viola/jones face detector, 2001.

[17] Maja Pantic and Leon J.M. Rothkrantzi. Automatic analysis of facial expressions: the state of the art. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, 2000.

[18] Lekshmi V Praseeda and M Sasikumar. Facial expression recognition from global and a combination of local features. *IETE Tech Rev*, 26(1):41–46, 2009.

[19] Lawrence R. Rabiner and Biing-Hwang Juang. Fundamentals of speech recognition, 1993.

[20] N Sebe, M S Lew, Y Sun, I Cohen, T Gevers, and T S Huang. Authentic facial expression analysis. *Image and Vision Computing*, 25:1856–1863, 2007.

[21] Padhraic Smyth. Face detection using the viola-jones method, 2007.

[22] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[23] UQUAM. The msfde. http://www.er.uqam.ca/nobel/r24700/Labo/Labo/MSEFE.html.

[24] Paul Viola and Michael Jones. Robust real-time object detection, 2001.

[25] Wikipedia. Statistical classification. http://en.wikipedia.org/wiki/Classifier_(mathematics).

Appendix A

Appendix A name

Here is the first appendix