NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Electrical and Electronic Engineering

Face Detection using Viola Jones Algorithm

Graduation Project

EE-400

Student: Rawan Mansour (20133337)

Supervisor: Assoc. Prof. Dr.: Özgür ÖZERDEM
Mohammed KMAIL

NICOSIA-2014

Acknowledgments

Words are often too less to reveal one's deep regards, I will take this opportunity to thanks everyone whose This work would not have been possible without their encouragement.

My special words of thanks and deepest appreciate should go to my supervisor Eng: Mohammed KMAIL, I will never find words to tell what I owe to him, and if I start doing it, I would not know where to stop , he has given his valuable time and advices. Thank you for encouragement and providing necessary information regarding the project & also for being patient, modesty, and friendly with us.

My deepest gratitude to all my family especially my parents who were a source of pride to me, thank you for your unflagging love and support in every field of my life, thank you for making my dream reality

I would like to express my special gratitude for all my friends and especially to my closest friend Jewel AL BAHRY, who has the purest heart, thank you for standing by my side, for listening, for encouragement and being true friend.

I would like to thank ....who I found my inspiration in his words and make the difference in my life to perfection personified.

Last gratitude and the most important is to my university and staff of Electric and Electronic Engineering Department for their help and support.

Abstract

In this article, we decipher the Viola-Jones algorithm, the first ever real-time face detection system. There are three ingredients working in concert to enable a fast and accurate detection:

The integral image for feature computation, Adaboost for feature selection and an intentional cascade for efficient computational resource allocation. Here we propose a complete algorithm is description, a learning code and a learned face detector that can be applied to any color image.

Since the Viola-Jones algorithm typically gives multiple detections, a post-processing step is also proposed to reduce detection redundancy using a robustness argument.

Contents

List of Figures

List of Tables

Chapter 1

The Face Recognition problem

1.1 Introduction

In recent years, face recognition has attracted much attention and its research has rapidly expanded by not only engineers but also neuroscientists, since it has many potential applications in computer vision communication and automatic access control system.

Especially, face detection is an important part of face recognition as the first step of automatic face recognition. However, face detection is not straightforward because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation, illuminating condition and facial expression. Many novel methods have been proposed to resolve each variation listed above. For example, the template-matching methods are used for face localization and detection by computing the correlation of an input image to a standard face pattern. The feature invariant approaches are used for feature detection of eyes, mouth, ears, nose, etc. The appearance-based methods are used for face detection with Eigen face neural network and information theoretical approach. Nevertheless, implementing the methods altogether is still a great challenge.

Since no objective distribution can describe the actual prior probability for a given image to have a face, the algorithm must minimize both the false negative and false positive rates in order to achieve an acceptable performance.

This task requires an accurate numerical description of what sets human faces apart from other objects. It turns out that these characteristics can be extracted with a remarkable committee learning algorithm called Adaboost, which relies on a committee of weak classifiers to form a strong a voting mechanism. A classifier is weak if, in general, it cannot meet a predefined classification target in error terms.

An operational algorithm must also work with a reasonable computational budget. Techniques such as integral image and intentional cascade make the Viola-Jones algorithm highly efficient.

1.2 Development through history:

Face recognition is one of the most relevant applications of image analysis. It's a true challenge to build an automated system which equals human ability to recognize faces. Although humans are quite good identifying known faces, we are not very skilled when we must deal with a large amount of unknown faces. The computers, with an almost limitless memory and computational speed, should overcome human's limitations.

Face recognition remains as an unsolved problem and a demanded technology - see table 1-1.A simple search with the phrase “face recognition” in the IEEE Digital Library throws 9422 results. 1332 articles in only one year 2009. There are many different industry areas interested in what it could offer. Some examples include video surveillance, human-machine interaction, photo cameras, virtual reality or law enforcement. This multidisciplinary interest pushesthe research and attracts interest from diverse disciplines. Therefore, it's not a problem restricted to computer vision research. Face recognition is a relevant subject in pattern recognition, neural networks, computer graphics, image processing and psychology. In fact, the earliest works on this subject were made in the 1950's in psychology. They came attached to other issues like face expression, interpretation of emotion or perception of gestures.

Engineering started to show interest in face recognition in the 1960's. One of the first researches on this subject was Woodrow W. Bledsoe. In 1960, Bledsoe, along other researches, started Panoramic Research, Inc., in Palo Alto, California. The majority of the work done by this company involved AI-related contracts from the U.S. Department of Defense and various intelligence agencies. During 1964 and 1965, Bledsoe, along with Helen Chan and Charles Bisson, worked on using computers to recognize human faces Because the funding of these researches was provided by an unnamed intelligence agency, little of the work was published. He continued later his researches at Stanford Research Institute. Bledsoe designed and implemented a semi-automatic system. Some face coordinates were selected by a human operator, and then computers used this information for recognition. He described most of the problems that even 50 years later Face Recognition still suffers - variations in illumination, head rotation, facial expression, and aging. Researches on this matter still continue, trying to measure subjective face features as ear size or between-eye distance. For instance, this approach was used in Bell Laboratories by A. Jay Goldstein, Leon D. Harmon and Ann B. Lesk. They described a vector, containing21 subjective features like ear protrusion, eyebrow weight or nose length, asthe basis to recognize faces using pattern classification techniques. In 1973, Fischlerand Elschanger tried to measure similar features automatically. Their algorithm used local template matching and a global measure of fit to find and measure facial features.

There were other approaches back on the 1970's. Some tried to define aface as a set of geometric parameters and then perform some pattern recognition based on those parameters. But the first one that developed a fully automated face recognition system was Kenade in 1973. He designed and implemented a face recognition program. It ran in a computer system designed for this purpose. The algorithm extracted sixteen facial parameters automatically. In he's work, Kenade compares this automated extraction to a human or manual extraction, showing only a small difference. He got a correct identification rate of 45-75%. He demonstrated that better results were obtained when irrelevant features were not used.

In the 1980's there were a diversity of approaches actively followed, most ofthem continuing with previous tendencies. Some works tried to improve the methods used measuring subjective features. For instance, Mark Nixon presented a geometric measurement for eye spacing. The template matching approach was improved with strategies such as "deformable templates" This decade also brought new approaches. Some researchers build face recognition algorithms using artificial neural networks.

The first mention to Eigen faces in image processing, a technique that would become the dominant approach in following years, was made by L. Sirovich and M. Kirby in 1986]. Their methods were based on the Principal Component Analysis. Their goal was to represent an image in a lower dimension without losing much information, and then reconstructing it. Their work would be later the foundation of the proposal of many new face recognition algorithms.

The 1990's saw the broad recognition of the mentioned Eigen face approaches the basis for the state of the art and the first industrial applications. In1992 Mathew Turk and Alex Pentland of the MIT presented a work which used Eigen faces for recognition. Their algorithm was able to locate, track and classify a subject's head. Since the 1990's, face recognition area has received a lot of attention, with a noticeable increase in the number of publications. Many approaches have been taken which has lead to different algorithms. Some of the most relevant are PCA, ICA, LDA and their derivatives. Different approaches and algorithms will be discussed later in this work.

The technologies using face recognition techniques have also evolved through the years. The first companies to invest in such researches were enforcement agencies; the Woodrow W. Bledsoe case. Nowadays diverse enterprises are using face recognition in their products. One good example could be entertainment business.

Products like Microsoft's Project Natal or Sony's PlayStation Eye will use face recognition. It will allow a new way to interact with the machine. The idea of detecting people and analyzing their gesture is also being used in automotive industry. Companies such as Toyota are developing sleep detectors to increase safety. These and other applications are raising the interest on face recognition. Its narrow initial application area is being widened.

Table 1.1: Applications of face recognition

Areas	Applications
Information Security	- Access security (OS, data bases) Data privacy (e.g. medical records) - User authentication (trading, on line banking)
Access management	- Secure access authentication (restricted facilities) - Permission based systems - Access log or audit trails
Biometrics	- Person identification (national IDs, Passports, voter registrations, driver licenses) - Automated identity verification (border controls)
Law Enforcement	- Video surveillance - Suspect identification Suspect tracking (investigation) Simulated aging - Forensic Reconstruction of faces from remains
Personal security	- Home video surveillance systems - Expression interpretation (driver monitoring system)
Entertainment - Leisure	- Home video game systems - Photo camera applications

1.3 Aims of the work

The aims of this work are concentrated on the following:

1- Study many algorithm for face detection.
2- Study and apply the Viola-Jones algorithm for face detection.
3- Study and apply the viola-Jones for tracking.
4- Apply the algorithm using the Matlabsoftware.

1.4 Scope of the work

This project is organized as follows:

Chapter one includes an introduction to the face detection work. A survey of some of the previous work in these fields is mentioned.

Chapter two will introduce the face detection problem and study its features.

Chapter three introduced the Viola Jones approach used in our work for face detection and tracking.

Chapter four contains the simulated program that applies the proposed method of Viola Jones including the applied code in Matlab.

Chapter five gives the final conclusions and suggestions for future works.

Chapter 2

Face detection features and analysis

2.1 Introduction:

This chapter will include the principles of face detection and the features it characterized by using theoretical approaches.

2.2 Psychological inspiration in automated face recognition:

Many researches tried to understand how humans recognize faces, most of them when the automatic face recognition problem arose, looking for design inspiration. It seems important to understand how we do this task, how we perceive humans. Then this knowledge could be applied in automatic face recognition systems.

However, many algorithms don't use this information, using just mathematical tools. Through these years some questions have emerged: Are features relevant to our eyes important for automatic face recognition? Can human vision system teach us useful thinks in this regard? Could psychological studies spotlight this problem in some way? In short, can the human face recognition ability help to develop a non-human face recognition system? This section will try to answer some relevant questions Is face recognition a dedicated process in the brain?

One early paper that answered this question was published by Diamond and Carey back in 1986. They presented four experiments. They tried to know if the difficulty of recognizing inverted faces was also common in other class of stimuli. At the same time, they tried to isolate the cause of this difficulty. They concluded that faces were no unique in the sense of being represented in memory in terms of special features. This may suggested that, consequently, face recognition has not a special spot in brain. This theory can be supported by the fact that patients with prosopag nosisa neurological condition in which it's very hard to recognize familiar faces- had also difficulties recognizing other familiar pictures.

More recent studies demonstrated that face recognition is a dedicated process in our brains. They demonstrated that recognizing human faces throw a negative ERP (event-related potential). They also found that it reflects the activity of cells turned to exclusively recognize human faces or face components. The same was true for inverted pictures. They suggested that there is a special process in our brains, and a special part of it, dedicated to recognize human faces.

This question remains unanswered and it is still a much debated issue. The dedication of the fusi-form face area (FFA) as a face processing module seems to be very strong. However, it may be responsible for performing subordinate or expert-level categorization of generic objects. We can conclude that there is a huge possibility that humans have a specialized face recognition mechanism .Are face and expression recognition separated systems?

It could be interesting to know if humans can extract facial expression independently from the identity of the subject and vice versa. Is facial expression an important constraint or condition in face recognition? Thus, can a bio- logical implementation of a computerized face recognition system identify faces in spite of facial expression? Many studies propose that identity and expression processes separate early in the facial perception procedure. Whether face recognition algorithm designers can find this information useful or not, that it's another matter .Is color an important factor in face recognition?

Many face recognition algorithms don't use color as a feature. However, it could be interesting to know if colors play a key role in human face recognition process. How objects are stored in the brain is a subject of much debate. Moreover, it isn't known if color cues play an important role in object recognition or not.

It is widely accepted that color cues do not provide diagnostic information for recognition, but they are not completely unrelated to face recognition systems. They could be nearly irrelevant when we try to recognize chromatically similar objects. On the other hand, it has been demonstrated that their contribution is essential under degraded conditions. So, color cues play an important role especially when shape cues are degraded. This feature could be extrapolated to face recognition system design .Does symmetry play an important role in face recognition?

From both neurological and computational point of view the answer is the same: yes. It has been demonstrated that an exceptional dimension reduction can be made by taking into account facial symmetry. The cited study also concludes that there are less than 70 dimensions for human recognition system. This result is smaller than the previously proposed $\simeq 100$ dimensions. The cause is the relevance of human face similarity.

2.3 Face recognition system structure:

Face Recognition is a term that includes several sub-problems. There are different classifications of these problems in the bibliography. Some of them will be explained on this section. Finally, a general or unified classification will be proposed.

2.3.1 A generic face recognition system:

The input of a face recognition system is always an image or video stream. The output is an identification or verification of the subject or subjects that appear in the image or video. Some approaches define a face recognition system as a three step process - see Figure 2.1. From this point of view, the Face Detection and Feature Extraction phases could run simultaneously.

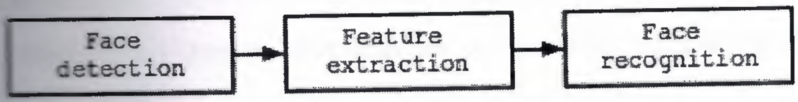


Figure (2.1): A generic face recognition system.

Face detection is defined as the process of extracting faces from scenes. So, the system positively identifies a certain image region as a face. This procedure has many applications like face tracking, pose estimation or compression. The next step - feature extraction- involves obtaining relevant facial features from the data. These features could be certain face regions, variations, angles or measures, which can be human relevant (e.g. eyes spacing) or not. This phase has other applications like facial feature tracking or emotion recognition. Finally, the system does recognize the face. In an identification task, the system would report an identity from a database. This phase involves a comparison method, a classification algorithm and an accuracy measure. This phase uses methods common to many other areas which also do some classification process -sound engineering, data mining.

These phases can be merged, or new ones could be added. Therefore, we could find many different engineering approaches to a face recognition problem. Face detection and recognition could be performed in tandem, or proceed to an expression analysis before normalizing the face.

2.4 Face detection:

In nowadays, some applications of Face Recognition don't require face detection. In some cases, face images stored in the data bases are already normalized. There is a standard image input format, so there is no need for a detection step. An example of this could be a criminal data base. There, the law enforcement agency stores faces of people with a criminal report. If there is new subject and the police has his or her passport photograph, face detection is not necessary. However, the conventional input images of computer vision systems are not that suitable. They can contain many items or faces. In these cases face detection is mandatory. It's also unavoidable if we want to develop an automated face tracking system. For example, video surveillance systems try to include face detection, tracking and recognizing. So, it's reasonable to assume face detection as part of the more ample face recognition problem.

Face detection must deal with several well known challenges. They are usually present in images captured in uncontrolled environments, such as surveillance video systems. These challenges can be attributed to some factors:

- ***Pose variation.*** The ideal scenario for face detection would be one in which only frontal images were involved. But, as stated, this is very unlikely in general uncontrolled conditions. Moreover, the performance of face detection algorithms drops severely when there are large pose variations. It's a major research issue. Pose variation can happen due to subject's movements or camera's angle.
- ***Feature occlusion*** .The presence of elements like beards, glasses or hats introduces high variability. Faces can also be partially covered by objects or other faces.
- ***Facial expression.*** Facial features also vary greatly because of different facial gestures.
- ***Imaging conditions.*** Different cameras and ambient conditions can affect the quality of an image, affecting the appearance of a face.

There are some problems closely related to face detection besides feature extraction and face classification. For instance, face location is a simplified approach of face detection. It's goal is to determine the location of a face in an image where there's only one face. We can differentiate between face detection and face location, since the latter is a simplified problem of the former. Methods like locating head boundaries were first used on this scenario and then exported to more complicated problems. Facial feature detection concerns detecting and locating some relevant features, such as nose, eye- brow, lips, ears, etc. Some feature extraction algorithms are based on facial feature detection. There is much literature on this topic, which is discussed later. Face tracking is other problem which sometimes is a consequence of face detection. Many systems' goal is not only to detect a face, but to be able to locate this face in real time. Once again, video surveillance system is a good example.

2.4.1 Face detection problem structure:

Face Detection is a concept that includes many sub-problems. Some systems detect and locate faces at the same time, others first perform a detection routine and then, if positive, they try to locate the face. Then, some tracking algorithms may be needed as defined in figure 2.2.

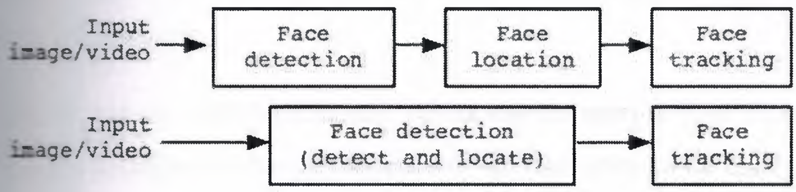


Figure (2.2), Face detection processes.

Face detection algorithms usually share common steps. Firstly, some data dimension reduction is done, in order to achieve a admissible response time. Some pre-processing could also be done to adapt the input image to the algorithm prerequisites. Then, some algorithms analyze the image as it is, and some others try to extract certain relevant facial regions. The next phase usually involves extracting facial features or measurements. These will then be weighted, evaluated or compared to decide if there is a face and where is it. Finally, some algorithms have a learning routine and they include new data to their models.

Face detection is, therefore, a two class problem where we have to decide if there is a face or not in a picture. This approach can be seen as a simplified face recognition problem. Face recognition has to classify a given face, and there are as many classes as candidates. Consequently, many face detection methods are very similar to face recognition algorithms. Or put another way, techniques used in face detection are often used in face recognition.

2.4.2 Approaches to face detection:

It's not easy to give a taxonomy of face detection methods. There isn't a globally accepted grouping criterion. They usually mix and overlap. In this section, two classification criteria will be presented. One of them differentiates between distinct scenarios. Depending on these scenarios different approaches may be needed. The other criterion divides the detection algorithms into four categories.

- **Detection depending on the scenario:**

Controlled environment: It's the most straightforward case. Photographs are taken under controlled light, background, etc. Simple edge detection techniques can be used to detect faces.

Color images. The typical skin colors can be used to find faces. They can be weak if light conditions change. Moreover, human skin color changes a lot, from nearly white to almost black. But, several studies show that the major difference lies between their intensity, so chrominance is a good feature. It's not easy to establish a solid human skin color representation. However, there are attempts to build robust face detection algorithms based on skin color.

Images in motion. Real time video gives the chance to use motion detection to localize faces. Nowadays, most commercial systems must locate faces in videos. There is a continuing challenge to achieve the best detecting results with the best possible performance. Another approach based on motion is eye blink detection, which has many uses aside from face detection.

- **Detection methods divided into categories:**

Methods are divided into four categories. These categories may over- lap, so an algorithm could belong to two or more categories. This classification can be made as follows:

- ***Knowledge-based methods.*** Ruled-based methods that encode our knowledge of human faces.
- ***Feature-invariant methods.*** Algorithms that try to find invariant features of a face despite it's angle or position.
- ***Template matching methods.*** These algorithms compare input images with stored patterns of faces or features.
- ***Appearance-based methods.*** A template matching method whose pattern database is learnt from a set of training images.

A- Knowledge-based methods:

These are rule-based methods. They try to capture our knowledge of faces, and translate them into a set of rules. It's easy to guess some simple rules. For example, a face usually has two symmetric eyes, and the eye area is darker than the cheeks. Facial features could be the distance between eyes or the color intensity difference between the eye area and the lower zone. The big problem with these methods is the difficulty in building an appropriate set of rules. There could be many false positives if the rules were too general. On the other hand, there could be many false negatives if the rules were too detailed. A solution is to build hierarchical knowledge-based methods to overcome these problems. However, this approach alone is very limited. It's unable to find many faces in a complex image.

Some researchers have tried to find some invariant features for face detection. The idea is to overcome the limits of our instinctive knowledge of faces. One early algorithm was developed by Han, Liao, Yu and Chen in 1997. The method is divided in several steps. Firstly, it tries to find eye-analogue pixels, so it removes unwanted pixels from the image. After performing these segmentation processes, they consider each eye-analogue segment as a candidate of one of the eyes. Then, a set of rule is executed to determinate the potential pair of eyes. Once the eyes are selected, the algorithms calculate the face area as a rectangle. The four vertexes of the face are determined by a set of functions. So, the potential faces are normalized to a fixed size and orientation. Then, the face regions are vivificated using a back propagation neural network. Finally, they apply a cost function to make the final selection. They report a success rate of 94%, even in photographs with many faces. These methods show themselves efficient with simple inputs.

There are other features that can deal with that problem. For example, there are algorithms that detect face-like textures or the color of human skin. It is very important to choose the best color model to detect faces.

However, the methods alone are usually not enough to build a good face detection algorithm. Skin color can vary significantly if light conditions change. Therefore, skin color detection is used in combination with other methods, like local symmetry or structure and geometry.

B- Template matching

Template matching methods try to define a face as a function. We try to find a standard template of all the faces. Different features can be defined independently. For example, a face can be divided into eyes, face contour, nose and mouth. Also a face model can be built by edges. But these methods are limited to faces that are frontal and unconcluded. A face can also be represented as a silhouette. Other templates use the relation between face regions in terms of brightness and darkness. These standard patterns are compared to the input images to detect faces. This approach is simple to implement, but it's inadequate for face detection. It cannot achieve good results with variations in pose, scale and shape. However, deformable templates have been proposed to deal with these problems

C- Appearance-based methods

The templates in appearance-based methods are learned from the examples in the images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. Some appearance-based methods work in a probabilistic net- work. An image or feature vector is a random variable with some probability of belonging to a face or not. Another approach is to define a discriminate function between face and non-face classes. These methods are also used in feature extraction for face recognition.

B- Appearance-based methods

The templates in appearance-based methods are learned from the examples in the images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. Some appearance-based methods work in a probabilistic net- work. An image or feature vector is a random variable with some probability of belonging to a face or not. Another approach is to define a discriminate function between face and non-face classes. These methods are also used in feature extraction for face recognition.

2.4.3 Face tracking:

Many face recognition systems have a video sequence as the input. Those systems may require being capable of not only detecting but tracking faces. Face tracking is essentially a motion estimation problem. Face tracking can be performed using many different methods, e.g., head tracking, feature tracking, image-based tracking, model-based tracking. These are different ways to classify these algorithms:

- Head tracking/Individual feature tracking. The head can be tracked as a whole entity, or certain features tracked individually.
- 2D/3D. Two dimensional systems track a face and output an image space where the face is located. Three dimensional systems, on the other hand, perform a 3D modeling of the face. This approach allows to estimate pose or orientation variations.

The basic face tracking process seeks to locate a given image in a picture. Then, it has to compute the differences between frames to update the location of the face. There are many issues that must be faced: Partial occlusions, illumination changes, computational speed and facial deformations.

2.5 Feature Extraction:

Humans can recognize faces since we are 5 year old. It seems to be an automated and dedicated process in our brains, though it's a much debated issue. What it's clear is that we can recognize people we know, even when they are wearing glasses or hats. We can also recognize men who have grown a beard. It's not very difficult for us to see our grandma's wedding photo and recognize her, although she was 23 years old. All these processes seem trivial, but they represent a challenge to the computers.

In fact, face recognition's core problem is to extract information from photographs. This feature extraction process can be defined as the procedure of extracting relevant information from a face image. This information must be valuable to the later step of identifying the subject with an acceptable error rate. The feature extraction process must be efficient in terms of computing time and memory usage. The output should also be optimized for the classification step.

Feature extraction involves several steps dimensionality reduction, feature extraction and feature selection. These steps may overlap, and dimensionality reduction could be seen as a consequence of the feature extraction and selection algorithms. Both algorithms could also be defined as cases of dimensionality reduction.

Ten times as many training samples per class as the number of features. This requirement should be satisfied when building a classifier. The more complex the classifier, the larger should be the mentioned ratio. This "curse" is one of the reasons why it's important to keep the number of features as small as possible. The other main reason is the speed. The classifier will be faster and will use less memory. Moreover, a large set of features can result in a false positive when these features are redundant. Ultimately, the number of features must be carefully chosen.

One can make a distinction between feature extraction and feature selection. Both terms are usually used interchangeably. Nevertheless, it is recommendable to make a distinction. A feature extraction algorithm extracts features from the data. It creates those new features based on transformations or combinations of the original data. In other words, it transforms or combines the data in order to select a proper subspace in the original feature space. On the other hand, a feature selection algorithm selects the best subset of the input feature set. It discards non-relevant features. Feature selection is often performed after feature extraction. So, features are extracted from the face images, then a optimum subset of these features is selected. The dimensionality reduction process can be embedded in some of these steps, or performed before them. This is arguably the most broadly accepted feature extraction process approach as shown in figure (2.3).

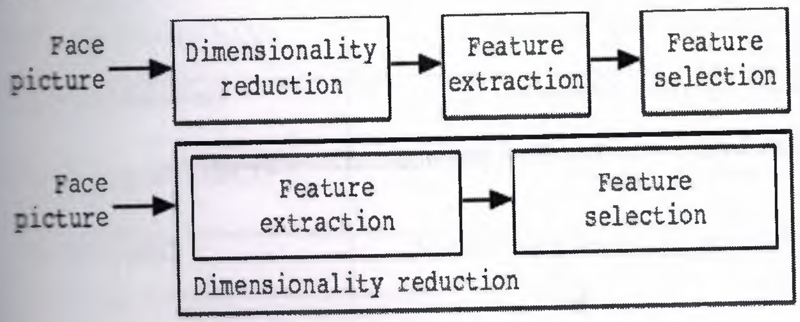


Figure (2.3): Feature extraction processes.

2.5.1Feature extraction methods:

There are many feature extraction algorithms. They will be discussed later on this paper. Most of them are used in other areas than face recognition. Researchers in face recognition have used many modified and adapted algorithms and methods for their purpose. For example, PCA was invented by Karl Pearson in 1901[88], but proposed for pattern recognition 64 years later. Finally, it was applied to face representation and recognition in the early 90's. See table 2.1 for a list of some feature extraction algorithms used in face recognition

Method	Notes
Principal Component Analysis (PCA)	Eigenvector-based, linear map
Kernel PCA	Eigenvector-based , non-linear map, uses kernel methods
Weighted PCA	PCA using weighted coefficients
Linear Discriminate Analysis (LDA)	Eigenvector-based, supervised linear map
Kernel LDA	LDA-based, uses kernel methods
Semi-supervised Discriminate Analysis (SDA)	Semi-supervised adaptation of LDA
Independent Component Analysis (ICA)	Linear map, separates non-Gaussian distributed features
Neural Network based methods	Diverse neural networks using PCA, etc.
Multidimensional Scaling (MDS)	Nonlinear map, sample size limited, noise sensitive.
Self-organizing map (SOM)	Nonlinear, based on a grid of neurons in the feature space
Active Shape Models (ASM)	Statistical method, searches boundaries
Active Appearance Models (AAM)	Evolution of ASM, uses shape and texture
Gavor wavelet transforms	Biologically motivated, linear filter
Discrete Cosine Transform (DCT)	Linear function, Fourier-related transform, usually used 2D-DCT
MMSD, SMSD	Methods using maximum scatter difference criterion.

Table 2.1: Feature extraction algorithms

2.5.2 Feature selection methods:

Feature selection algorithm's aim is to select a subset of the extracted features that cause the smallest classification error. The importance of this error is what makes feature selection dependent to the classification method used. The most straightforward approach to this problem would be to examine every possible subset and choose the one that fulfills the criterion function. However, this can become an unaffordable task in terms of computational time. Some effective approaches to this problem are based on algorithms like branch and bound algorithms. See table 2.2 for selection methods.

Method	Definition	Comments
haustive search	Evaluate all possible subsets of features.	Optimal, but too complex.
nch and bound	Use branch and bound algorithm.	Can be optimal. But also Complex
dividual features	Evaluate and select features individually.	Not very effective. Simple algorithm.
quential Forward Selection	Evaluate growing feature sets (starts with best feature).	Retained features can't be discarded. Faster than SBS.
quential Backward Selection(SBS)	Evaluate shrinking feature sets (starts with all the features).	Deleted features can't be re-evaluated.
l -take away r " selection	First do SFS then SBS.	Must choose l and r values
ial Forward Floating (SFFS) and Sequential ward Floating Search (SBFS)	Like "Plus l -take away r ", but l and r values automatic pick and dynamic update.	Close to optimal. Affordable computational cost.

Table 2.2: Feature selection methods

Recently more feature selection algorithms have been proposed. Some approaches have used resemblance coefficient or satisfactory rate as a criterion and quantum genetic algorithm (QGA).

2.6 Face classification:

Once the features are extracted and selected, the next steps to classify the image. Appearance-based face recognition algorithms use a wide variety of classification methods. Sometimes two or more classifiers are combined to achieve better results. On the other hand, most model-based algorithms match the samples with the model or template. Then, a learning method is can be used to improve the algorithm. One way or another, classifiers have a big impact in face recognition. Classification methods are used in many areas like data mining, finance, signal decoding, voice recognition, natural language processing or medicine. Therefore, there are many bibliographies regarding this subject. Here classifiers will be addressed from a general pattern recognition point of view.

Classification algorithms usually involve some learning - supervised, un- supervised or semi-supervised. Unsupervised learning is the most difficult approach, as there are no tagged examples. However, many face recognition applications include a tagged set of subjects. Consequently, most face recognition systems implement supervised learning methods. There are also cases where the labeled data set is small. Sometimes, the acquisition of new tagged samples can be infeasible. Therefore, semi-supervised learning is required.

2.6.1 Classifiers:

According to Jain, Duin and Mao, there are three concepts that are key in building a classifier - similarity, probability and decision boundaries. We will present the classifiers from that point of view.

- Similarity

This approach is intuitive and simple. Patterns that are similar should be- long to the same class. This approach has been used in the face recognition algorithms implemented later. The idea is to establish a metric that de- fines similarity and a representation of the same-class samples. For example, the metric can be the Euclidean distance. The representation of a class can be the mean vector of all the patterns belonging to this class. The 1-NN decision rule can be used with this parameters. It's classification performance is usually good. This approach is similar to a k-means clustering algorithm in unsupervised learning. There are other techniques that can be used. For example, Vector Quantization, Learning Vector Quantization or Self-Organizing Maps - see 1.4. Other example of this approach is template matching. Researches classify face recognition algorithm based on different criteria. Some publications defined Template Matching as a kind or category of face recognition algorithms. However, we can see template matching just as another classification method, where unlabeled samples are compared to stored patterns.

- Probability

Some classifiers are built based on a probabilistic approach. Bayes decision rule is often used. The rule can be modified to take into account different factors that could lead to miss-classification.

2.7 Face recognition:

Face recognition is an evolving area, changing and improving constantly. Many research areas affect face recognition - computer vision, optics, pattern recognition, neural networks, machine learning, psychology, etc. Previous sections explain the different steps of a face recognition process. However, these steps can overlap or change depending on the bibliography consulted. There is not a consensus on that regard. All these factors hinder the development of a unified face recognition algorithm classification scheme. This section explains the most cited criteria.

2.7.1 Geometric/Template Based approaches:

Face recognition algorithms can be classified as either geometry based or template based algorithms. The template based methods compare the input image with a set of templates. The set of templates can be constructed using statistical tools like Super Vector Machines (SVM), Principal Component Analysis (PCA), Linear

NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Electrical and Electronic Engineering

Face Detection using Viola Jones Algorithm

Graduation Project

EE-400

Student: Rawan Mansour (20133337)

Supervisor: Assoc. Prof. Dr.: Özgür ÖZERDEM

Mohammed KMAIL

NICOSIA-2014

Acknowledgments

Words are often too less to reveal one's deep regards, I will take this opportunity to thanks everyone whose This work would not have been possible without their encouragement.

My special words of thanks and deepest appreciate should go to my supervisor Eng: Mohammed KMAIL, I will never find words to tell what I owe to him, and if I start doing it, I would not know where to stop , he has given his valuable time and advices. Thank you for encouragement and providing necessary information regarding the project & also for being patient, modesty, and friendly with us.

My deepest gratitude to all my family especially my parents who were a source of pride to me, thank you for your unflagging love and support in every field of my life, thank you for making my dream reality

I would like to express my special gratitude for all my friends and especially to my closest friend Jewel AL BAHRY, who has the purest heart, thank you for standing by my side, for listening, for encouragement and being true friend.

I would like to thank ....who I found my inspiration in his words and make the difference in my life to perfection personified.

Last gratitude and the most important is to my university and staff of Electric and Electronic Engineering Department for their help and support.

Abstract

In this article, we decipher the Viola-Jones algorithm, the first ever real-time face detection system. There are three ingredients working in concert to enable a fast and accurate detection:

The integral image for feature computation, Adaboost for feature selection and an intentional cascade for efficient computational resource allocation. Here we propose a complete algorithm is description, a learning code and a learned face detector that can be applied to any color image.

Since the Viola-Jones algorithm typically gives multiple detections, a post-processing step is also proposed to reduce detection redundancy using a robustness argument.

Contents

List of Figures

List of Tables

Chapter 1

The Face Recognition problem

1.1 Introduction

In recent years, face recognition has attracted much attention and its research has rapidly expanded by not only engineers but also neuroscientists, since it has many potential applications in computer vision communication and automatic access control system.

Especially, face detection is an important part of face recognition as the first step of automatic face recognition. However, face detection is not straightforward because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation, illuminating condition and facial expression. Many novel methods have been proposed to resolve each variation listed above. For example, the template-matching methods are used for face localization and detection by computing the correlation of an input image to a standard face pattern. The feature invariant approaches are used for feature detection of eyes, mouth, ears, nose, etc. The appearance-based methods are used for face detection with Eigen face neural network and information theoretical approach. Nevertheless, implementing the methods altogether is still a great challenge.

Since no objective distribution can describe the actual prior probability for a given image to have a face, the algorithm must minimize both the false negative and false positive rates in order to achieve an acceptable performance.

This task requires an accurate numerical description of what sets human faces apart from other objects. It turns out that these characteristics can be extracted with a remarkable committee learning algorithm called Adaboost, which relies on a committee of weak classifiers to form a strong a voting mechanism. A classifier is weak if, in general, it cannot meet a predefined classification target in error terms.

An operational algorithm must also work with a reasonable computational budget. Techniques such as integral image and intentional cascade make the Viola-Jones algorithm highly efficient.

1.2 Development through history:

Face recognition is one of the most relevant applications of image analysis. It's a true challenge to build an automated system which equals human ability to recognize faces. Although humans are quite good identifying known faces, we are not very skilled when we must deal with a large amount of unknown faces. The computers, with an almost limitless memory and computational speed, should overcome human's limitations.

Face recognition remains as an unsolved problem and a demanded technology - see table 1-1.A simple search with the phrase "face recognition" in the IEEE Digital Library throws 9422 results. 1332 articles in only one year 2009. There are many different industry areas interested in what it could offer. Some examples include video surveillance, human-machine interaction, photo cameras, virtual reality or law enforcement. This multidisciplinary interest pushesthe research and attracts interest from diverse disciplines. Therefore, it's not a problem restricted to computer vision research. Face recognition is a relevant subject in pattern recognition, neural networks, computer graphics, image processing and psychology. In fact, the earliest works on this subject were made in the 1950's in psychology. They came attached to other issues like face expression, interpretation of emotion or perception of gestures.

Engineering started to show interest in face recognition in the 1960's. One of the first researches on this subject was Woodrow W. Bledsoe. In 1960, Bledsoe, along other researches, started Panoramic Research, Inc., in Palo Alto, California. The majority of the work done by this company involved AI-related contracts from the U.S. Department of Defense and various intelligence agencies. During 1964 and 1965, Bledsoe, along with Helen Chan and Charles Bisson, worked on using computers to recognize human faces Because the funding of these researches was provided by an unnamed intelligence agency, little of the work was published. He continued later his researches at Stanford Research Institute. Bledsoe designed and implemented a semi-automatic system. Some face coordinates were selected by a human operator, and then computers used this information for recognition. He described most of the problems that even 50 years later Face Recognition still suffers - variations in illumination, head rotation, facial expression, and aging. Researches on this matter still continue, trying to measure subjective face features as ear size or between-eye distance. For instance, this approach was used in Bell Laboratories by A. Jay Goldstein, Leon D. Harmon and Ann B. Lesk. They described a vector, containing21 subjective features like ear protrusion, eyebrow weight or nose length, asthe basis to recognize faces using pattern classification techniques. In 1973, Fischlerand Elschanger tried to measure similar features automatically. Their algorithm used local template matching and a global measure of fit to find and measure facial features.

There were other approaches back on the 1970's. Some tried to define aface as a set of geometric parameters and then perform some pattern recognition based on those parameters. But the first one that developed a fully automated face recognition system was Kenade in 1973. He designed and implemented a face recognition program. It ran in a computer system designed for this purpose. The algorithm extracted sixteen facial parameters automatically. In he's work, Kenade compares this automated extraction to a human or manual extraction, showing only a small difference. He got a correct identification rate of 45-75%. He demonstrated that better results were obtained when irrelevant features were not used.

I the 1980's there were a diversity of approaches actively followed, most ofthem continuing with previous tendencies. Some works tried to improve the methods used measuring subjective features. For instance, Mark Nixon presented a geometric measurement for eye spacing. The template matching approach was improved with strategies such as "deformable templates" This decade also brought new approaches. Some researchers build face recognition algorithms using artificial neural networks.

The first mention to Eigen faces in image processing, a technique that would become the dominant approach in following years, was made by L. Sirovich and M. Kirby in 1986]. Their methods were based on the Principal Component Analysis. Their goal was to represent an image in a lower dimension without losing much information, and then reconstructing it. Their work would be later the foundation of the proposal of many new face recognition algorithms.

The 1990's saw the broad recognition of the mentioned Eigen face approaches the basis for the state of the art and the first industrial applications. In1992 Mathew Turk and Alex Pentland of the MIT presented a work which used Eigen faces for recognition. Their algorithm was able to locate, track and classify a subject's head. Since the 1990's, face recognition area has received a lot of attention, with a noticeable increase in the number of publications. Many approaches have been taken which has lead to different algorithms. Some of the most relevant are PCA, ICA, LDA and their derivatives. Different approaches and algorithms will be discussed later in this work.

The technologies using face recognition techniques have also evolved through the years. The first companies to invest in such researches were enforcement agencies; the Woodrow W. Bledsoe case. Nowadays diverse enterprises are using face recognition in their products. One good example could be entertainment business.

Products like Microsoft's Project Natal or Sony's PlayStation Eye will use face recognition. It will allow a new way to interact with the machine. The idea of detecting people and analyzing their gesture is also being used in automotive industry. Companies such as Toyota are developing sleep detectors to increase safety. These and other applications are raising the interest on face recognition. Its narrow initial application area is being widened.

Table 1.1: Applications of face recognition

Areas	Applications
Information Security	- Access security (OS, data bases) Data privacy (e.g. medical records) - User authentication (trading, on line banking)
Access management	- Secure access authentication (restricted facilities) - Permission based systems - Access log or audit trails
Biometrics	- Person identification (national IDs, Passports, voter registrations, driver licenses) - Automated identity verification (border controls)
Law Enforcement	- Video surveillance - Suspect identification Suspect tracking (investigation) Simulated aging - Forensic Reconstruction of faces from remains
Personal security	- Home video surveillance systems - Expression interpretation (driver monitoring system)
Entertainment - Leisure	- Home video game systems - Photo camera applications

1.3 Aims of the work

The aims of this work are concentrated on the following:

1- Study many algorithm for face detection.
2- Study and apply the Viola-Jones algorithm for face detection.
3- Study and apply the viola-Jones for tracking.
4- Apply the algorithm using the Matlabsoftware.

1.4 Scope of the work

This project is organized as follows:

Chapter one includes an introduction to the face detection work. A survey of some of the previous work in these fields is mentioned.

Chapter two will introduce the face detection problem and study its features.

Chapter three introduced the Viola Jones approach used in our work for face detection and tracking.

Chapter four contains the simulated program that applies the proposed method of Viola Jones including the applied code in Matlab.

Chapter five gives the final conclusions and suggestions for future works.

Chapter 2

Face detection features and analysis

2.1 Introduction:

This chapter will include the principles of face detection and the features it characterized by using theoretical approaches.

2.2 Psychological inspiration in automated face recognition:

Many researches tried to understand how humans recognize faces, most of them when the automatic face recognition problem arose, looking for design inspiration. It seems important to understand how we do this task, how we perceive humans. Then this knowledge could be applied in automatic face recognition systems.

However, many algorithms don't use this information, using just mathematical tools. Through these years some questions have emerged: Are features relevant to our eyes important for automatic face recognition? Can human vision system teach us useful thinks in this regard? Could psychological studies spotlight this problem in some way? In short, can the human face recognition ability help to develop a non-human face recognition system? This section will try to answer some relevant questions Is face recognition a dedicated process in the brain?

One early paper that answered this question was published by Diamond and Carey back in 1986. They presented four experiments. They tried to know if the difficulty of recognizing inverted faces was also common in other class of stimuli. At the same time, they tried to isolate the cause of this difficulty. They concluded that faces were no unique in the sense of being represented in memory in terms of special features. This may suggested that, consequently, face recognition has not a special spot in brain. This theory can be supported by the fact that patients with prosopag nosisa neurological condition in which it's very hard to recognize familiar faces- had also difficulties recognizing other familiar pictures.

More recent studies demonstrated that face recognition is a dedicated process in our brains. They demonstrated that recognizing human faces throw a negative ERP (event-related potential). They also found that it reflects the activity of cells turned to exclusively recognize human faces or face components. The same was true for inverted pictures. They suggested that there is a special process in our brains, and a special part of it, dedicated to recognize human faces.

This question remains unanswered and it is still a much debated issue. The dedication of the fusi-form face area (FFA) as a face processing module seems to be very strong. However, it may be responsible for performing subordinate or expert-level categorization of generic objects. We can conclude that there is a huge possibility that humans have a specialized face recognition mechanism .Are face and expression recognition separated systems?

It could be interesting to know if humans can extract facial expression independently from the identity of the subject and vice versa. Is facial expression an important constraint or condition in face recognition? Thus, can a bio- logical implementation of a computerized face recognition system identify faces in spite of facial expression? Many studies propose that identity and expression processes separate early in the facial perception procedure. Whether face recognition algorithm designers can find this information useful or not, that it's another matter .Is color an important factor in face recognition?

Many face recognition algorithms don't use color as a feature. However, it could be interesting to know if colors play a key role in human face recognition process. How objects are stored in the brain is a subject of much debate. Moreover, it isn't known if color cues play an important role in object recognition or not.

It is widely accepted that color cues do not provide diagnostic information for recognition, but they are not completely unrelated to face recognition systems. They could be nearly irrelevant when we try to recognize chromatically similar objects. On the other hand, it has been demonstrated that their contribution is essential under degraded conditions. So, color cues play an important role especially when shape cues are degraded. This feature could be extrapolated to face recognition system design .Does symmetry play an important role in face recognition?

From both neurological and computational point of view the answer is the same: yes. It has been demonstrated that an exceptional dimension reduction can be made by taking into account facial symmetry. The cited study also concludes that there are less than 70 dimensions for human recognition system. This result is smaller than the previously proposed $\simeq 100$ dimensions. The cause is the relevance of human face similarity.

2.3 Face recognition system structure:

Face Recognition is a term that includes several sub-problems. There are different classifications of these problems in the bibliography. Some of them will be explained on this section. Finally, a general or unified classification will be proposed.

2.3.1 A generic face recognition system:

The input of a face recognition system is always an image or video stream. The output is an identification or verification of the subject or subjects that appear in the image or video. Some approaches define a face recognition system as a three step process - see Figure 2.1. From this point of view, the Face Detection and Feature Extraction phases could run simultaneously.

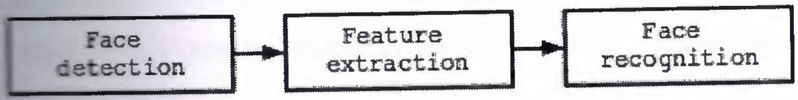


Figure (2.1): A generic face recognition system.

Face detection is defined as the process of extracting faces from scenes. So, the system positively identifies a certain image region as a face. This procedure has many applications like face tracking, pose estimation or compression. The next step -feature extraction- involves obtaining relevant facial features from the data. These features could be certain face regions, variations, angles or measures, which can be human relevant (e.g. eyes spacing) or not. This phase has other applications like facial feature tracking or emotion recognition. Finally, the system does recognize the face. In an identification task, the system would report an identity from a database. This phase involves a comparison method, a classification algorithm and an accuracy measure. This phase uses methods common to many other areas which also do some classification process -sound engineering, data mining.

These phases can be merged, or new ones could be added. Therefore, we could find many different engineering approaches to a face recognition problem. Face detection and recognition could be performed in tandem, or proceed to an expression analysis before normalizing the face.

2.4 Face detection:

In nowadays, some applications of Face Recognition don't require face detection. In some cases, face images stored in the data bases are already normalized. There is a standard image input format, so there is no need for a detection step. An example of this could be a criminal data base. There, the law enforcement agency stores faces of people with a criminal report. If there is new subject and the police has his or her passport photograph, face detection is not necessary. However, the conventional input images of computer vision systems are not that suitable. They can contain many items or faces. In these cases face detection is mandatory. It's also unavoidable if we want to develop an automated face tracking system. For example, video surveillance systems try to include face detection, tracking and recognizing. So, it's reasonable to assume face detection as part of the more ample face recognition problem.

Face detection must deal with several well known challenges. They are usually present in images captured in uncontrolled environments, such as surveillance video systems. These challenges can be attributed to some factors:

- ***Pose variation.*** The ideal scenario for face detection would be one in which only frontal images were involved. But, as stated, this is very unlikely in general uncontrolled conditions. Moreover, the performance of face detection algorithms drops severely when there are large pose variations. It's a major research issue. Pose variation can happen due to subject's movements or camera's angle.
- ***Feature occlusion*** .The presence of elements like beards, glasses or hats introduces high variability. Faces can also be partially covered by objects or other faces.
- ***Facial expression.*** Facial features also vary greatly because of different facial gestures.
- ***Imaging conditions.*** Different cameras and ambient conditions can affect the quality of an image, affecting the appearance of a face.

There are some problems closely related to face detection besides feature extraction and face classification. For instance, face location is a simplified approach of face detection. It's goal is to determine the location of a face in an image where there's only one face. We can differentiate between face detection and face location, since the latter is a simplified problem of the former. Methods like locating head boundaries were first used on this scenario and then exported to more complicated problems. Facial feature detection concerns detecting and locating some relevant features, such as nose, eye- brow, lips, ears, etc. Some feature extraction algorithms are based on facial feature detection. There is much literature on this topic, which is discussed later. Face tracking is other problem which sometimes is a consequence of face detection. Many systems' goal is not only to detect a face, but to be able to locate this face in real time. Once again, video surveillance system is a good example.

2.4.1 Face detection problem structure:

Face Detection is a concept that includes many sub-problems. Some systems detect and locate faces at the same time, others first perform a detection routine and then, if positive, they try to locate the face. Then, some tracking algorithms may be needed as defined in figure 2.2.

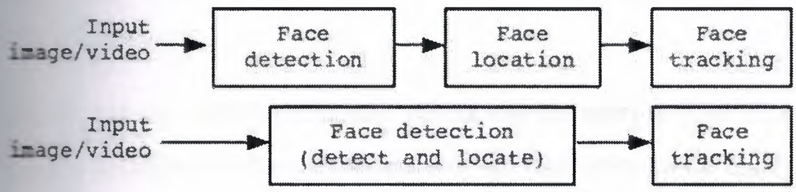


Figure (2.2), Face detection processes.

Face detection algorithms usually share common steps. Firstly, some data dimension reduction is done, in order to achieve a admissible response time. Some pre-processing could also be done to adapt the input image to the algorithm prerequisites. Then, some algorithms analyze the image as it is, and some others try to extract certain relevant facial regions. The next phase usually involves extracting facial features or measurements. These will then be weighted, evaluated or compared to decide if there is a face and where is it. Finally, some algorithms have a learning routine and they include new data to their models.

Face detection is, therefore, a two class problem where we have to decide if there is a face or not in a picture. This approach can be seen as a simplified face recognition problem. Face recognition has to classify a given face, and there are as many classes as candidates. Consequently, many face detection methods are very similar to face recognition algorithms. Or put another way, techniques used in face detection are often used in face recognition.

2.4.2 Approaches to face detection:

It's not easy to give a taxonomy of face detection methods. There isn't a globally accepted grouping criterion. They usually mix and overlap. In this section, two classification criteria will be presented. One of them differentiates between distinct scenarios. Depending on these scenarios different approaches may be needed. The other criterion divides the detection algorithms into four categories.

- **Detection depending on the scenario:**

Controlled environment: It's the most straightforward case. Photographs are taken under controlled light, background, etc. Simple edge detection techniques can be used to detect faces.

Color images. The typical skin colors can be used to find faces. They can be weak if light conditions change. Moreover, human skin color changes a lot, from nearly white to almost black. But, several studies show that the major difference lies between their intensity, so chrominance is a good feature. It's not easy to establish a solid human skin color representation. However, there are attempts to build robust face detection algorithms based on skin color.

Images in motion. Real time video gives the chance to use motion detection to localize faces. Nowadays, most commercial systems must locate faces in videos. There is a continuing challenge to achieve the best detecting results with the best possible performance. Another approach based on motion is eye blink detection, which has many uses aside from face detection.

- **Detection methods divided into categories:**

Methods are divided into four categories. These categories may over- lap, so an algorithm could belong to two or more categories. This classification can be made as follows:

- ***Knowledge-based methods.*** Ruled-based methods that encode our knowledge of human faces.
- ***Feature-invariant methods.*** Algorithms that try to find invariant features of a face despite it's angle or position.
- ***Template matching methods.*** These algorithms compare input images with stored patterns of faces or features.
- ***Appearance-based methods.*** A template matching method whose pattern database is learnt from a set of training images.

A- Knowledge-based methods:

These are rule-based methods. They try to capture our knowledge of faces, and translate them into a set of rules. It's easy to guess some simple rules. For example, a face usually has two symmetric eyes, and the eye area is darker than the cheeks. Facial features could be the distance between eyes or the color intensity difference between the eye area and the lower zone. The big problem with these methods is the difficulty in building an appropriate set of rules. There could be many false positives if the rules were too general. On the other hand, there could be many false negatives if the rules were too detailed. A solution is to build hierarchical knowledge-based methods to overcome these problems. However, this approach alone is very limited. It's unable to find many faces in a complex image.

Some researchers have tried to find some invariant features for face detection. The idea is to overcome the limits of our instinctive knowledge of faces. One early algorithm was developed by Han, Liao, Yu and Chen in 1997. The method is divided in several steps. Firstly, it tries to find eye-analogue pixels, so it removes unwanted pixels from the image. After performing these segmentation processes, they consider each eye-analogue segment as a candidate of one of the eyes. Then, a set of rule is executed to determinate the potential pair of eyes. Once the eyes are selected, the algorithms calculate the face area as a rectangle. The four vertexes of the face are determined by a set of functions. So, the potential faces are normalized to a fixed size and orientation. Then, the face regions are vivificated using a back propagation neural network. Finally, they apply a cost function to make the final selection. They report a success rate of 94%, even in photographs with many faces. These methods show themselves efficient with simple inputs.

There are other features that can deal with that problem. For example, there are algorithms that detect face-like textures or the color of human skin. It is very important to choose the best color model to detect faces.

However, the methods alone are usually not enough to build a good face detection algorithm. Skin color can vary significantly if light conditions change. Therefore, skin color detection is used in combination with other methods, like local symmetry or structure and geometry.

B- Template matching

Template matching methods try to define a face as a function. We try to find a standard template of all the faces. Different features can be defined independently. For example, a face can be divided into eyes, face contour, nose and mouth. Also a face model can be built by edges. But these methods are limited to faces that are frontal and unconcluded. A face can also be represented as a silhouette. Other templates use the relation between face regions in terms of brightness and darkness. These standard patterns are compared to the input images to detect faces. This approach is simple to implement, but it's inadequate for face detection. It cannot achieve good results with variations in pose, scale and shape. However, deformable templates have been proposed to deal with these problems

C- Appearance-based methods

The templates in appearance-based methods are learned from the examples in the images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. Some appearance-based methods work in a probabilistic net- work. An image or feature vector is a random variable with some probability of belonging to a face or not. Another approach is to define a discriminate function between face and non-face classes. These methods are also used in feature extraction for face recognition.

D- Appearance-based methods

The templates in appearance-based methods are learned from the examples in the images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. Some appearance-based methods work in a probabilistic net- work. An image or feature vector is a random variable with some probability of belonging to a face or not. Another approach is to define a discriminate function between face and non-face classes. These methods are also used in feature extraction for face recognition.

2.4.3 Face tracking:

Many face recognition systems have a video sequence as the input. Those systems may require being capable of not only detecting but tracking faces. Face tracking is essentially a motion estimation problem. Face tracking can be performed using many different methods, e.g., head tracking, feature tracking, image-based tracking, model-based tracking. These are different ways to classify these algorithms:

- Head tracking/Individual feature tracking. The head can be tracked as a whole entity, or certain features tracked individually.
- 2D/3D. Two dimensional systems track a face and output an image space where the face is located. Three dimensional systems, on the other hand, perform a 3D modeling of the face. This approach allows to estimate pose or orientation variations.

The basic face tracking process seeks to locate a given image in a picture. Then, it has to compute the differences between frames to update the location of the face. There are many issues that must be faced: Partial occlusions, illumination changes, computational speed and facial deformations.

2.5 Feature Extraction:

Humans can recognize faces since we are 5 year old. It seems to be an automated and dedicated process in our brains, though it's a much debated issue. What it's clear is that we can recognize people we know, even when they are wearing glasses or hats. We can also recognize men who have grown a beard. It's not very difficult for us to see our grandma's wedding photo and recognize her, although she was 23 years old. All these processes seem trivial, but they represent a challenge to the computers.

In fact, face recognition's core problem is to extract information from photographs. This feature extraction process can be defined as the procedure of extracting relevant information from a face image. This information must be valuable to the later step of identifying the subject with an acceptable error rate. The feature extraction process must be efficient in terms of computing time and memory usage. The output should also be optimized for the classification step.

Feature extraction involves several steps dimensionality reduction, feature extraction and feature selection. These steps may overlap, and dimensionality reduction could be seen as a consequence of the feature extraction and selection algorithms. Both algorithms could also be defined as cases of dimensionality reduction.

Ten times as many training samples per class as the number of features. This requirement should be satisfied when building a classifier. The more complex the classifier, the larger should be the mentioned ratio. This "curse" is one of the reasons why it's important to keep the number of features as small as possible. The other main reason is the speed. The classifier will be faster and will use less memory. Moreover, a large set of features can result in a false positive when these features are redundant. Ultimately, the number of features must be carefully chosen.

One can make a distinction between feature extraction and feature selection. Both terms are usually used interchangeably. Nevertheless, it is recommendable to make a distinction. A feature extraction algorithm extracts features from the data. It creates those new features based on transformations or combinations of the original data. In other words, it transforms or combines the data in order to select a proper subspace in the original feature space. On the other hand, a feature selection algorithm selects the best subset of the input feature set. It discards non-relevant features. Feature selection is often performed after feature extraction. So, features are extracted from the face images, then a optimum subset of these features is selected. The dimensionality reduction process can be embedded in some of these steps, or performed before them. This is arguably the most broadly accepted feature extraction process approach as shown in figure (2.3).

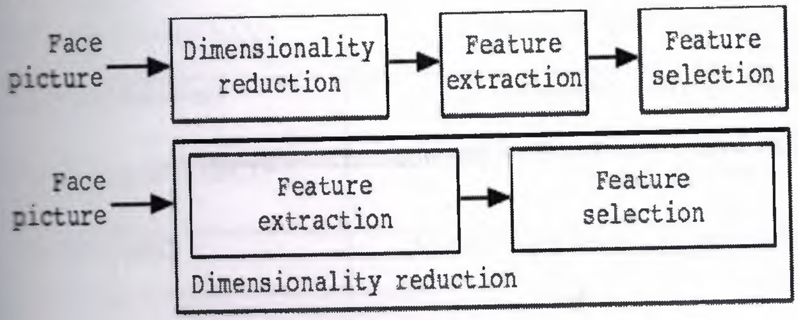


Figure (2.3): Feature extraction processes.

2.5.1Feature extraction methods:

There are many feature extraction algorithms. They will be discussed later on this paper. Most of them are used in other areas than face recognition. Researchers in face recognition have used many modified and adapted algorithms and methods for their purpose. For example, PCA was invented by Karl Pearson in 1901[88], but proposed for pattern recognition 64 years later. Finally, it was applied to face representation and recognition in the early 90's. See table 2.1 for a list of some feature extraction algorithms used in face recognition

Method	Notes
Principal Component Analysis (PCA)	Eigenvector-based, linear map
Kernel PCA	Eigenvector-based , non-linear map, uses kernel methods
Weighted PCA	PCA using weighted coefficients
Linear Discriminate Analysis (LDA)	Eigenvector-based, supervised linear map
Kernel LDA	LDA-based, uses kernel methods
Semi-supervised Discriminate Analysis (SDA)	Semi-supervised adaptation of LDA
Independent Component Analysis (ICA)	Linear map, separates non-Gaussian distributed features
Neural Network based methods	Diverse neural networks using PCA, etc.
Multidimensional Scaling (MDS)	Nonlinear map, sample size limited, noise sensitive.
Self-organizing map (SOM)	Nonlinear, based on a grid of neurons in the feature space
Active Shape Models (ASM)	Statistical method, searches boundaries
Active Appearance Models (AAM)	Evolution of ASM, uses shape and texture
Gavor wavelet transforms	Biologically motivated, linear filter
Discrete Cosine Transform (DCT)	Linear function, Fourier-related transform, usually used 2D-DCT
MMSD, SMSD	Methods using maximum scatter difference criterion.

Table 2.1: Feature extraction algorithms

2.5.2 Feature selection methods:

Feature selection algorithm's aim is to select a subset of the extracted features that cause the smallest classification error. The importance of this error is what makes feature selection dependent to the classification method used. The most straightforward approach to this problem would be to examine every possible subset and choose the one that fulfills the criterion function. However, this can become an unaffordable task in terms of computational time. Some effective approaches to this problem are based on algorithms like branch and bound algorithms. See table 2.2 for selection methods.

Method	Definition	Comments
haustive search	Evaluate all possible subsets of features.	Optimal, but too complex.
anch and bound	Use branch and bound algorithm.	Can be optimal. But also Complex
dividual features	Evaluate and select features individually.	Not very effective. Simple algorithm.
uential Forward Selection	Evaluate growing feature sets (starts with best feature).	Retained features can't be discarded. Faster than SBS.
uential Backward election(SBS)	Evaluate shrinking feature sets (starts with all the features).	Deleted features can't be re-evaluated.
l -take away r " selection	First do SFS then SBS.	Must choose l and r values
ial Forward Floating SFFS) and Sequential ward Floating Search (SBFS)	Like "Plus l -take away r ", but l and r values automatic pick and dynamic update.	Close to optimal. Affordable computational cost.

Table 2.2: Feature selection methods

Recently more feature selection algorithms have been proposed. Some approaches have used resemblance coefficient or satisfactory rate as a criterion and quantum genetic algorithm (QGA).

2.6 Face classification:

Once the features are extracted and selected, the next steps to classify the image. Appearance-based face recognition algorithms use a wide variety of classification methods. Sometimes two or more classifiers are combined to achieve better results. On the other hand, most model-based algorithms match the samples with the model or template. Then, a learning method is can be used to improve the algorithm. One way or another, classifiers have a big impact in face recognition. Classification methods are used in many areas like data mining, finance, signal decoding, voice recognition, natural language processing or medicine. Therefore, there are many bibliographies regarding this subject. Here classifiers will be addressed from a general pattern recognition point of view.

Classification algorithms usually involve some learning - supervised, un- supervised or semi-supervised. Unsupervised learning is the most difficult approach, as there are no tagged examples. However, many face recognition applications include a tagged set of subjects. Consequently, most face recognition systems implement supervised learning methods. There are also cases where the labeled data set is small. Sometimes, the acquisition of new tagged samples can be infeasible. Therefore, semi-supervised learning is required.

2.6.1 Classifiers:

According to Jain, Duin and Mao, there are three concepts that are key in building a classifier - similarity, probability and decision boundaries. We will present the classifiers from that point of view.

- Similarity

This approach is intuitive and simple. Patterns that are similar should be- long to the same class. This approach has been used in the face recognition algorithms implemented later. The idea is to establish a metric that de- fines similarity and a representation of the same-class samples. For example, the metric can be the Euclidean distance. The representation of a class can be the mean vector of all the patterns belonging to this class. The 1-NN decision rule can be used with this parameters. It's classification performance is usually good. This approach is similar to a k-means clustering algorithm in unsupervised learning. There are other techniques that can be used. For example, Vector Quantization, Learning Vector Quantization or Self-Organizing Maps - see 1.4. Other example of this approach is template matching. Researches classify face recognition algorithm based on different criteria. Some publications defined Template Matching as a kind or category of face recognition algorithms. However, we can see template matching just as another classification method, where unlabeled samples are compared to stored patterns.

- Probability

Some classifiers are built based on a probabilistic approach. Bayes decision rule is often used. The rule can be modified to take into account different factors that could lead to miss-classification.

2.7 Face recognition:

Face recognition is an evolving area, changing and improving constantly. Many research areas affect face recognition - computer vision, optics, pattern recognition, neural networks, machine learning, psychology, etc. Previous sections explain the different steps of a face recognition process. However, these steps can overlap or change depending on the bibliography consulted. There is not a consensus on that regard. All these factors hinder the development of a unified face recognition algorithm classification scheme. This section explains the most cited criteria.

2.7.1 Geometric/Template Based approaches:

Face recognition algorithms can be classified as either geometry based or template based algorithms. The template based methods compare the input image with a set of templates. The set of templates can be constructed using statistical tools like Super Vector Machines (SVM), Principal Component Analysis (PCA), Linear

Discriminate Analysis (LDA), Independent Component Analysis (ICA), Kernel Methods, or Trace Transforms.

The geometry feature-based methods analyze local facial features and their geometric relationships. This approach is sometimes called feature- based approach. Examples of this approach are some Elastic Bunch Graph Matching algorithms. This approach is less used nowadays. There are algorithms developed using both approaches. For instance, a 3D morph able model approach can use feature points or texture as well as PCA to build a recognition system.

2.7.2 Piecemeal/W holistic approaches:

Faces can often be identified from little information. Some algorithms follow this idea, processing facial features independently. In other words, the relation between the features or the relation of a feature with the whole face is not taken into account. Many early researchers followed this approach, trying to deduce the most relevant features. Some approaches tried to use the eyes, a combination of features, and so on. Some Hidden Markov Model (HMM) methods also fall in this category .Although feature processing is very important in face recognition, relation between features (configure processing) is also important. In fact, facial features are processed holistically. That's why nowadays most algorithms follow a holistic approach.

2.7.3Appeareance-based/Model-based approaches:

Facial recognition methods can be divided into appearance-based or model- based algorithms. The differential element of these methods is the representation of the face. Appearance-based methods represent a face in terms of several raw intensity images. An image is considered as a high-dimensional vector. Then statistical techniques are usually used to derive a feature space from the image distribution. The sample image is compared to the training set. On the other hand, the model-based approach tries to model a human face. The new sample is fitted to the model, and the parameters of the fatten model used to recognize the image. Appearance methods can be classified as linear or non-linear, while model-based methods can be 2D or 3D.

Linear appearance-based methods perform a linear dimension reduction. The face vectors are projected to the basis vectors, the projection coefficients are used as the feature representation of each face image. Examples of this approach are PCA, LDA or ICA. Non-linear appearance methods are more complicate. In fact, linear subspace analysis is an approximation of a non- linear manifold. Kernel PCA (KPCA) is a method widely used.

Model-based approaches can be 2-Dimensional or 3-Dimensional. These algorithms try to build a model of a human face. These models are often morph able. A morph able model allows classifying faces even when pose changes are present. 3D models are more complicate, as they try to capture the three dimensional nature of human faces.

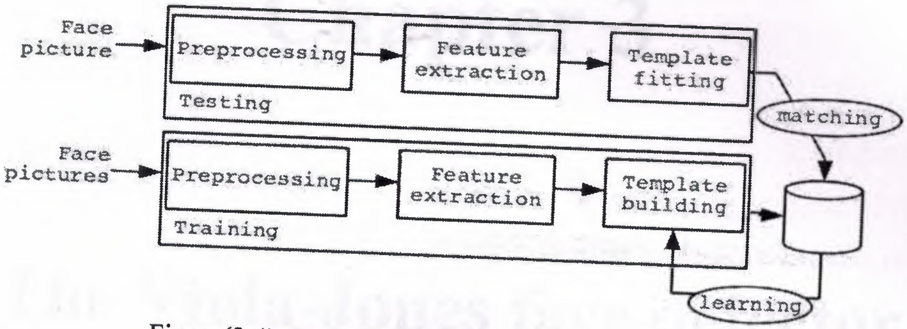
2.7.4 Template/statistical/neural network approaches:

A similar separation of pattern recognition algorithms into four groups is proposed by Jain and colleges. We can grope face recognition methods into three main groups. The following approaches are proposed:

Template matching. Patterns are represented by samples, models, pixels, curves, textures. The recognition function is usually a correlation or distance measure.

Statistical approach: Patterns are represented as features. The recognition function is a discriminate function.

Neural networks: The representation may vary. There is a network function in some point .Note that many algorithms, mostly current complex algorithms, may fall into more than one of these categories. The most relevant face recognition algorithms will be discussed later under this classification.



Figure(2.4), Template-matching algorithm diagram

Chapter 3

The Viola-Jones face detector

1 Introduction:

s chapter describes the work carried out concerning the Viola-Jones face detection orithm. The first part elaborates on the methods and theory behind three more ver full methods have been recently used for detection. Secondly proposed orithm which is relatively short, but still the most important points are explained interesting aspects of the actual implementation are emphasized and presented ether with results and comments on performance.

2 The Viola-Jones algorithm

s seems to be the first article where Viola-Jones present the coherent set of ideas constitute the fundamentals of their face detection algorithm. This algorithm only s frontal upright faces, but in 2003 presented in a variant that also detects profile rotated views in 2001, [2].

basic principle of the Viola-Jones algorithm is to scan a sub-window capable of cting faces across a given input image. The standard image processing approach ld be to rescale the input image to different sizes and then run the fixed size ctor through these images. This approach turns out to be rather time consuming to the calculation of the different size images.

ontrary to the standard approach Viola-Jones rescale the detector instead of the input nage and run the detector many times through the image – each time with a different ze. At first one might suspect both approaches to be equally time consuming, but iola-Jones have devised a scale invariant detector that requires the same number of lculations whatever the size. This detector is constructed using a so-called integral nage and some simple rectangular features. The next section elaborates on this tector. In steps, starting with the scale invariant detector as a first step of Viola nes algorithm

3.2.1 The scale invariant detector:

e first step of the Viola-Jones face detection algorithm is to turn the input image o an integral image. This is done by making each pixel equal to the entire sum of pixels above and to the left of the concerned pixel .This is demonstrated in Figure .1).

1	1	1
1	1	1
1	1	1

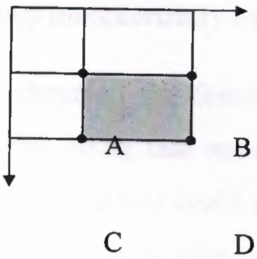
Input image

1	2	3
2	4	6
3	6	9

Integral image

Figure(3.1),The integral image.

s allows for the calculation of the sum of all pixels inside any given rectangle ng only four values. These values are the pixels in the integral image that ncide with the corners of the rectangle in the input image. This is demonstrated in ure(3.2).

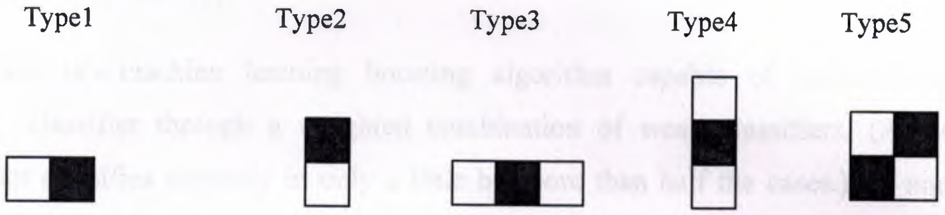


Sum of grey rectangle= D-(B+C)+A

Figure (3.2): Sum calculation.

Since both rectangle B and C include rectangle A the sum of A has to be added to the calculation.

It has now been demonstrated how the sum of pixels within rectangles of arbitrary size can be calculated in constant time. The Viola-Jones face detector analyzes a given sub-window using features consisting of two or more rectangles. The different types of features are shown in Figure (3.3).



Figure(3.3), The different types of features.

Each feature results in a single value which is calculated by subtracting the sum of the white rectangle(s) from the sum of the black rectangle(s).

Viola-Jones has empirically found that a detector with a base resolution of 24*24 pixels gives satisfactory results. When allowing for all possible sizes and positions of the features in Figure 4 a total of approximately 160.000 different features can then be constructed. Thus, the amount of possible features vastly outnumbers the 576 pixels contained in the detector at base resolution. These features may seem overly simple to perform such an advanced task as face detection, but what the features lack in complexity they most certainly have in computational efficiency.

One could understand the features as the computer's way of perceiving an input image. The hope being that some features will yield large values when on top of a face. Of course operations could also be carried out directly on the raw pixels, but the variation due to different pose and individual characteristics would be expected to mper this approach. The goal is now to smartly construct a mesh of features capable detecting faces and this is the topic of the next section.

3.2.2 The modified AdaBoost algorithm:

s stated above there can be calculated approximately 160.000 feature values within a tector at base resolution. Among all these features some few are expected to give most consistently high values when on top of a face. In order to find these features ola-Jones use a modified version of the AdaBoost algorithm developed by Freund d Schapiro in 1996 [5].

daBoost is a machine learning boosting algorithm capable of constructing a ong classifier through a weighted combination of weak classifiers. (A weak ssifier classifies correctly in only a little bit more than half the cases.) To match is terminology to the presented theory each feature is considered to be a potential eak classifier. A weak classifier is mathematically described as:

$$h(x,f,p,\theta) = \begin{cases} 1 \ if \ pf(x) > p\theta \\ 0 \quad Otherwise \end{cases}$$

here x is a 24*24 pixel sub-window, f is the applied feature, p the polarity and θ the eshold that decides whether x should be classified as a positive (a face) or a gative (a non-face).

ce only a small amount of the possible 160.000 feature values are expected to be tential weak classifiers the AdaBoost algorithm is modified to select only the best tures.

important part of the modified AdaBoost algorithm is the determination of the best ture, polarity and threshold. There seems to be no smart solution to this problem d Viola-Jones suggest a simple brute force method. This means that the termination of each new weak classifier involves evaluating each feature on all the ining examples in order to find the best performing feature. This is expected to be most time consuming part of the training procedure.

The best performing feature is chosen based on the weighted error it produces. is weighted error is a function of the weights belonging to the training examples.

.s seen in Figure 5 part 4) the weight of a correctly classified example is decreased nd the weight of a misclassified example is kept constant. As a result it is more xpensive' for the second feature (in the final classifier) to misclassify an example lso misclassified by the first feature, than an example classified correctly. An lternative interpretation is that the second feature is forced to focus harder on the xamples misclassified by the first. The point being that the weights are a vital part of ne mechanics of the AdaBoost algorithm.

Vith the integral image, the computationally efficient features and the modified daBoost algorithm in place it seems like the face detector is ready for nplementation, but Viola-Jones have one more ace up the sleeve.

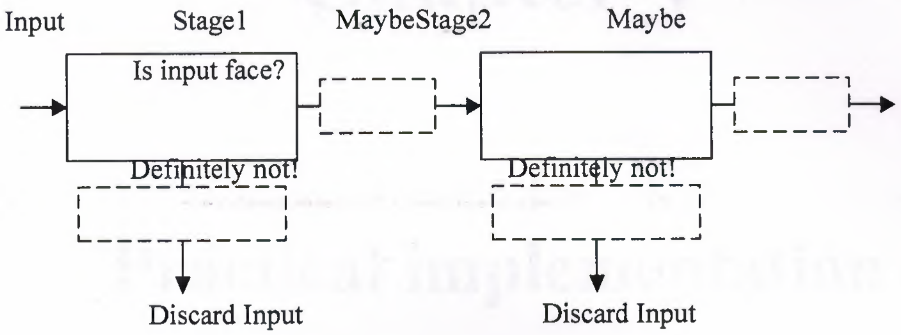
3.2.3 The cascaded classifier:

'he basic principle of the Viola-Jones face detection algorithm is to scan the detector nany times through the same image – each time with a new size. Even if an image hould contain one or more faces it is obvious that an excessive large amount of the valuated sub-windows would still be negatives (non-faces). This realization leads to different formulation of the problem:

Instead of finding faces, the algorithm should discard non-faces.

'he thought behind this statement is that it is faster to discard a non-face than to find face. With this in mind a detector consisting of only one (strong) classifier suddenly eems inefficient since the evaluation time is constant no matter the input. Hence the eed for a cascaded classifier arises.

'he cascaded classifier is composed of stages each containing a strong classifier. The ob of each stage is to determine whether a given sub-window is definitely not a face r maybe a face. When a sub-window is classified to be a non-face by a given stage it immediately discarded. Conversely a sub-window classified as a maybe-face is assed on to the next stage in the cascade. It follows that the more stages a given subvindow passes, the higher the chance the sub-window actually contains a face. The oncept is illustrated with two stages in Figure (3.4).



Figure(3.4), The cascaded classifier.

In a single stage classifier one would normally accept false negatives in order reduce the false positive rate. However, for the first stages in the staged classifier se positives are not considered to be a problem since the succeeding stages are pected to sort them out. Therefore Viola-Jones prescribe the acceptance of many se positives in the initial stages. Consequently the amount of false negatives in the al staged classifier is expected to be very small.

ola-Jones also refer to the cascaded classifier as an intentional cascade. This name plies that more attention (computing power) is directed towards the regions of the age suspected to contain faces.

follows that when training a given stage, say n, the negative examples should of urse be false negatives generated by stage n-1.

e majority of thoughts presented in the 'Methods' section are taken from the iginal Viola-Jones paper [1].

Chapter 4

Practical implementation

4.1 Introduction to chapter:

Object detection and tracking are important in many computer vision applications including activity recognition, automotive safety, and surveillance. This chapter includes the application of Viola –Jones algorithm for a simple face for detection and tracking system by dividing the tracking problem into three separate problems:

i. Detect a face to track.

ii. Identify facial features to track.

iii. Track the face.

4.2 Detect a Face To Track

Before we begin tracking a face, we need to first detect it. We use thevision.CascadeObjectDetectorto detect the location of a face in a video frame. The cascade object detector uses the Viola-Jones detection algorithm and a trained classification model for detection.

```
faceDetector = vision.CascadeObjectDetector();
```

It uses the Viola-Jones algorithm to detect people's faces, noses, eyes, mouth, or upper body. It creates a System object,faceDetector, that detects objects using

e Viola-Jones algorithm. TheClassificationModelproperty controls the type
 object to detect. By default, the detector is configured to detect faces.

```
ideoFileReader                                                  =
ision.VideoFileReader('visionface.avi');
```

reads video frames, images, and audio samples from a video file. The object can so readimages files. It returns a video file reader System ject,videoFileReader. The object can sequentially read video frames and/or dio samples from the input video file, visionface.avi.

```
ideoFrame        = step(videoFileReader);
```

outputs the next video frame. It uses the video file reader ject,videoFileReader, and returns the next video frame,videoFrame.

```
box              = step(faceDetector, videoFrame);
```

tect objects using the Viola-Jones algorithm. It returns bbox, an M-by-4 matrix fining Mbounding boxes containing the detected objects in the input age,videoFrame.

```
xInserter
vision.ShapeInserter('BorderColor','Custom',...
CustomBorderColor',[255 255 0]);
```

aw rectangles, lines, polygons, or circles on an image. It returns a System object, xInserter, that draws multiple rectangles, lines, polygons, or circles on images overwriting pixel values. Each specified property set to the specified value.

```
deoOut = step(boxInserter, videoFrame,bbox);
```

aw specified shape on image. It draws the shape specified by the, boxInserter, pertyon input image,videoFrame. The inputbboxspecify the coordinates for the ation of the shape. The shapes are embedded on the output imagevideoOut.

```
gure, imshow(videoOut), title('Detected face');
```

reate a figure with title`Detectedface`and display the image`videoOut`.This is own in Figure(4.1).

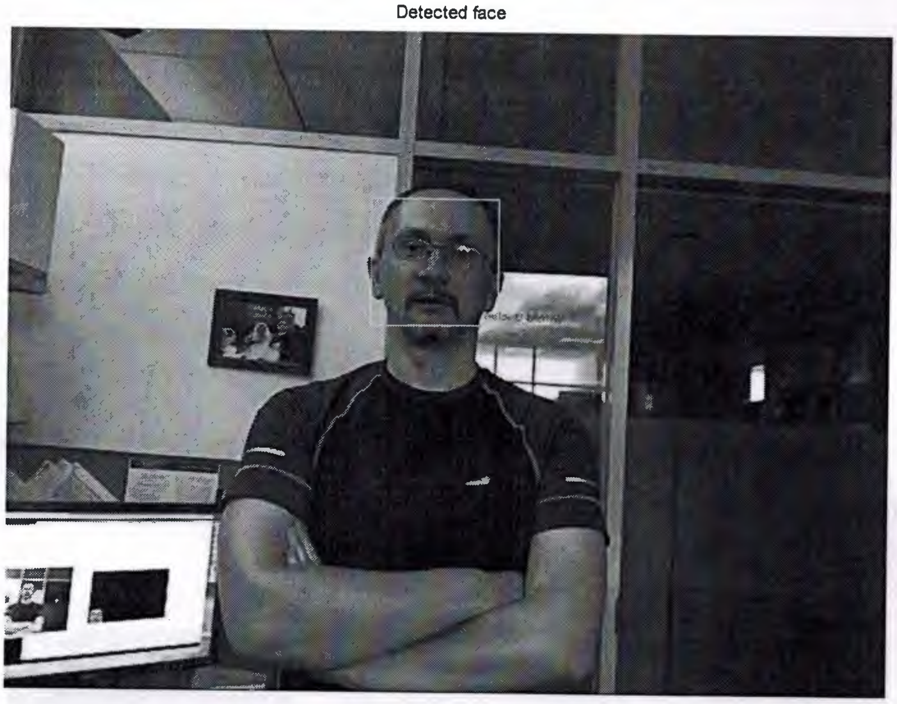


Figure (4.1): Detected face

We can use the cascade object detector to track a face across successive video mes. However, when the face tilts or the person turns theirs head, we may lose cking. This limitation is due to the type of trained classification model used for ection. To avoid this issue, and because performing face detection for every video me is computationally intensive, we use a simple facial feature for tracking.

3 Identify Facial Features To Track

ce the face is located in the video, the next step is to identify a feature that will p us track the face. For example, we can use the shape, texture, or colour. We ose a feature that is unique to the object and remains invariant even when the ect moves.

e use skin tone as the feature to track. The skin tone provides a good deal of ntrast between the face and the background and does not change as the face rotates moves.

```
ueChannel,~,~] = rgb2hsv(videoFrame);
```

onvert RGB colour map to HSV colour map. It converts the RGB image to the uivalent HSV image.`videoFrame`is an m-by-n-by-3 image array whose three anes contain the red, green, and blue components for the image. HSV is returned as m-by-n-by-3 image array whose three planes contain the hue, saturation, and value mponents for the imageas shown in figure (4.2).

```
igure, imshow(hueChannel), title('Hue channel data');
```

eate a figure with title`Hue channel data`and display the image`hueChannel`.

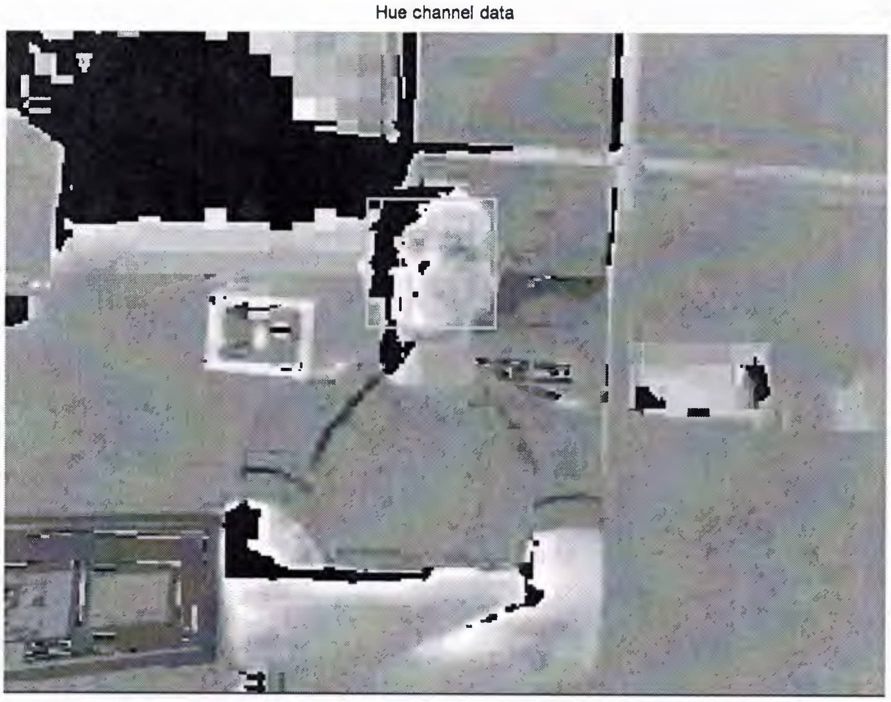


Figure (4.2): Hue channel data

```
ctangle('Position',bbox(1,:),'LineWidth',2,'EdgeColor',
 1 0])
```

is code draws the rectangle from the point specified in the `bbox` array.

.4 RGB to HSV & HSV to RGB

ne Hue/Saturation/Value model was created by A. R. Smith in 1978. It is based on ch intuitive color characteristics as tint, shade and tone (or family, purity and tensity). The coordinate system is cylindrical, and the colors' are defined inside a x cone. The hue value H runs from 0 to 360°. The saturation S is the degree of rength or purity and is from 0 to 1. Purity is how much white is added to the color, S=1 makes the purest color (no white). Brightness V also ranges from 0 to 1, where is the black.

4.4.1 RGB to HSV conversion formula:

ne R, G, B values are divided by 255 to change the range from 0 → 255 to 0 → 1:

$$= \mathrm{R/255}$$

$$\mathrm{G'} = \mathrm{G/255}$$

$$\mathrm{B'} = \mathrm{B/255}$$

$$\mathrm{Cmax} = \max(\mathrm{R'}, \mathrm{G'}, \mathrm{B'})$$

$$\mathrm{Cmin} = \min(\mathrm{R'}, \mathrm{G'}, \mathrm{B'})$$

$$\Delta = \mathrm{Cmax} - \mathrm{Cmin}$$

ue calculation: $$\mathrm{H} = \begin{cases} 60^{\circ} \times \Big(\Big(\frac{\acute{G}-\acute{B}}{\Delta} \mathrm{mod} 6\Big) & ,\mathrm{Cmax} = \acute{R} \\ 60^{\circ} \times \Big(\frac{\acute{B}-\acute{R}}{\Delta} + 2\Big) & ,\mathrm{Cmax} = \acute{G} \\ 60^{\circ} \times \Big(\frac{\acute{R}-\acute{G}}{\Delta} + 4\Big) & ,\mathrm{Cmax} = \acute{B} \end{cases}$$

turation calculation: $$\mathrm{S} = \begin{cases} 0, \Delta = 0 \\ \frac{\Delta}{\mathrm{Cmax}}, \Delta <> 0 \end{cases}$$

lue calculation: $\mathrm{V} = \mathrm{Cmax}$

- MATLAB Code:

```
RGB2HSV Convert red-green-blue colors to hue-saturation-
alue.
 H = RGB2HSV(M) converts an RGB color map to an HSV
olor map.
 Each map is a matrix with any number of rows, exactly
hree
olumns,
 and elements in the interval 0 to 1. The columns of the
nput
atrix,
 M, represent intensity of red, blue and green,
espectively. The
 columns of the resulting output matrix, H, represent
ue,
aturation
 and color value, respectively.

 HSV = RGB2HSV(RGB) converts the RGB image RGB (3-D
rray) to the
 equivalent HSV image HSV (3-D array).

 CLASS SUPPORT
 -------------
 If the input is an RGB image, it can be of class uint8,
int16,
r
 double; the output image is of class double. If the
nput is a
 colormap, the input and output colormaps are both of
lass double.

 See also HSV2RGB, COLORMAP, RGBPLOT.

 Undocumented syntaxes:
```

```
 [H,S,V] = RGB2HSV(R,G,B) converts the RGB image R,G,B
o the
 equivalent HSV image H,S,V.

 HSV = RGB2HSV(R,G,B) converts the RGB image R,G,B to
he
 equivalent HSV image stored in the 3-D array (HSV).

 [H,S,V] = RGB2HSV(RGB) converts the RGB image RGB (3-D
rray) to
 the equivalent HSV image H,S,V.

 See Alvy Ray Smith, Color Gamut Transform Pairs,
IGGRAPH '78.

 Copyright 1984-2006 The MathWorks, Inc.
 $Revision: 5.15.4.2 $ $Date: 2006/10/02 16:33:03 $
```

4.5 Tracking the Face

With the skin tone selected as the feature to track, we can now use the vision.HistogramBasedTracker for tracking. The histogram based tracker uses the CAMShift algorithm, which provides the capability to track an object using a histogram of pixel values. In our system, the Hue channel pixels are extracted from the nose region of the detected face. These pixels are used to initialize the histogram for the tracker. The system tracks the object over successive video frames using this histogram.

```
noseDetector = vision.CascadeObjectDetector('Nose');
faceImage     = imcrop(videoFrame,bbox);
It crops the image videoFrame. bbox is a four element
position vector that specifies the size and position of
the crop rectangle.
```

```
oseBBox       = step(noseDetector,faceImage);
```

amShift is a tracking algorithm, which is based on MeanShift algorithm, what amShift do is nothing but do mean Shift in every single frame of a video, and record e results we got by mean Shift.

amShift algorithm includes these three parts:

- 1. Back Projection
- 2. MeanShift
- 3. Track

nd I will simply explain each of these steps in this blog.

4.5.1Back Projection:

ack projection is a method which using the histogram of an image to show up the obabilities of colors may appear in each pixel. Let's see how to get the back ojection of an image.

```
cvtColor(image, hsv, CV_BGR2HSV);
int ch[]={0,0};
hue.create(hsv.size(), hsv.depth());
mixChannels(&hsv, 1, &hue, 1, ch, 1);
calcHist(&hue, 1, 0, Mat(), hist, 1, &hsize,
&phranges);
normalize(hist, hist, 0,255, CV_MINMAX);
calcBackProject( &hue, 1, 0, hist, backproj, &phranges,
1, true );
```

First we transform the picture space to HSV space (or any space which include H channel that represent the hue of each pixel, of course, value of hue is between 0 180, you can see more info inwiki.) Secondly, we split the H channel out, as a ngle grayscale image, and get its histogram, and normalize it. Thirdly, use calcBackProject()” function to calculate the back projection of the image.

et me use an example to explain how we get the back projection.

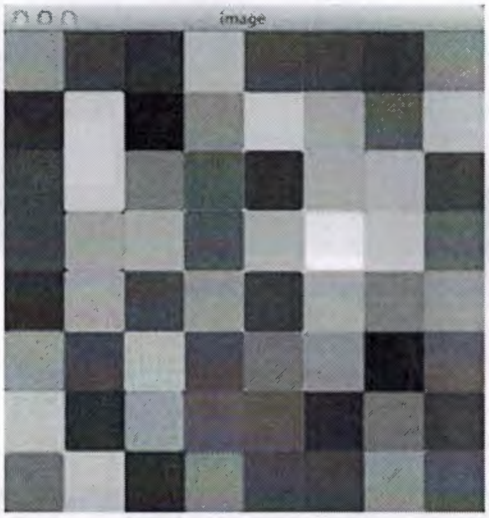
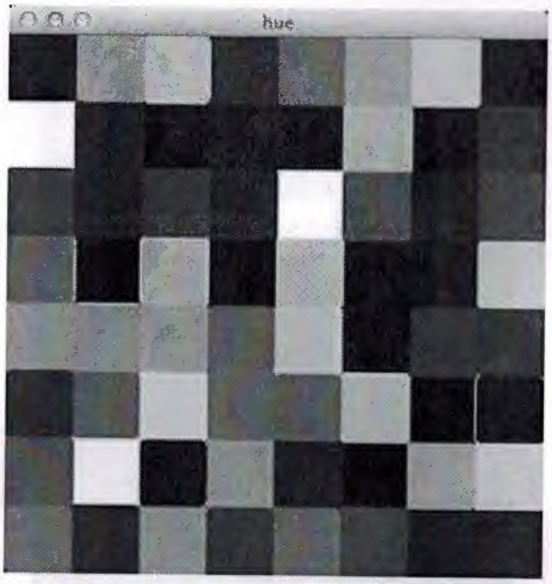


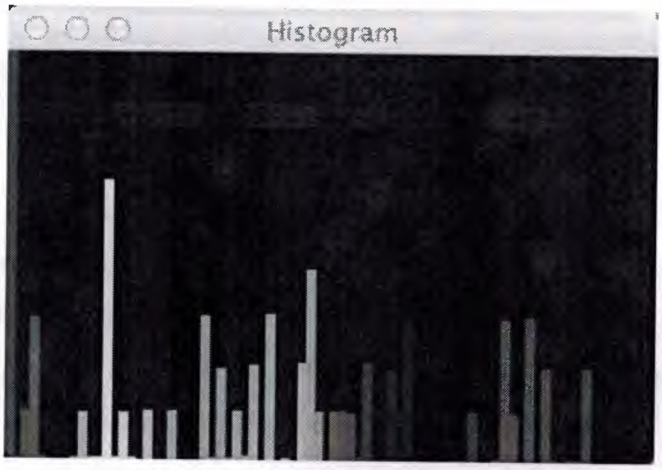
Figure (4.3): color image

the image shown in figure (4.3) is our input image, we can see it is a colorful osaic picture .As we talked above, transform the picture into HSV space and here is e hue channel as shown in figure (4.4).



Figure(4.4), Hue image

ne histogram is shown in figure(4.5).



Figure(4.5), Histogram image

e "calcBackProject()" function actually calculate the weight of each color in the ole picture using histogram, and change the value of each pixel to the weight of its or in whole picture. For instance, if one pixel's color is, say yellow, and the color low's weight in this picture is 20%, that is, there are 20% of pixels' color in the ole picture is this kind of yellow, we change this pixel's value from yellow to 0.2

(or 0.2*255 if using integer), by doing this method to all pixels, we get the back projection picture shown in figure(4.6).

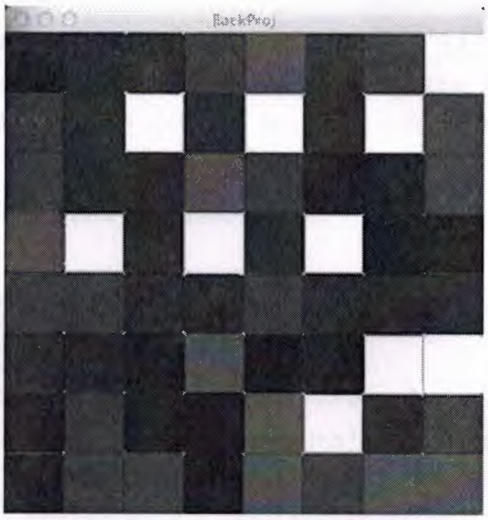


Figure (4.6),Back project image

4.5.2Tracking the Face

The last step is tracking, if we have a video, or frames captured by our web camera, what we need to do is just use our proposed algorithm every single frame, and the initial window of each frame is just the output window of the prior frame. Interesting, except center, size, we can also get an angle of the rectangle, which means we can track the orientation of our target, this is a very useful feature.

This this the last step for detecting and tracking a face in a video file simulated using the Matlab tool by the following code:

```
% Create a cascade detector object.
faceDetector = vision.CascadeObjectDetector();

% Read a video frame and run the detector.
videoFileReader = vision.VideoFileReader('visionface.avi');
videoFrame      = step(videoFileReader);
bbox            = step(faceDetector, videoFrame);
```

```
 Draw the returned bounding box around the detected face.
oxInserter  =vision.ShapeInserter('BorderColor','Custom',...
CustomBorderColor',[255 255 0])
ideoOut = step(boxInserter, videoFrame,bbox);
gure, imshow(videoOut), title('Detected face');
 Get the skin tone information by extracting the Hue from the video
ame
 converted to the HSV color space.
hueChannel,~,~] = rgb2hsv(videoFrame);

 Display the Hue Channel data and draw the bounding box around the
ce.
gure, imshow(hueChannel), title('Hue channel data');
ctangle('Position',bbox(1,:),'LineWidth',2,'EdgeColor',[1 1 0])

Detect the nose within the face region. The nose provides a more
curate
measure of the skin tone because it does not contain any background
pixels.
seDetector = vision.CascadeObjectDetector('Nose');
ceImage    = imcrop(videoFrame,bbox);
seBBox     = step(noseDetector,faceImage);

The nose bounding box is defined relative to the cropped face
age.
Adjust the nose bounding box so that it is relative to the original
deo
frame.
seBBox(1:2) = noseBBox(1:2) + bbox(1:2);

Create a tracker object.
cker = vision.HistogramBasedTracker;

nitialize the tracker histogram using the Hue channel pixels from

ose.
tializeObject(tracker, hueChannel, noseBBox);

reate a video player object for displaying video frames.
eoInfo    = info(videoFileReader);
```

```
ideoPlayer  =vision.VideoPlayer('Position',[300 300
ideoInfo.VideoSize+30]);

 Track the face over successive video frames until the video is
inished.
hile ~isDone(videoFileReader)

 Extract the next video frame
ideoFrame = step(videoFileReader);

 RGB -> HSV
     [hueChannel,~,~] = rgb2hsv(videoFrame);

 Track using the Hue channel data
box = step(tracker, hueChannel);

 Insert a bounding box around the object being tracked
ideoOut = step(boxInserter, videoFrame, bbox);

 Display the annotated video frame using the video player object
tep(videoPlayer, videoOut);

nd

 Release resources
elease(videoFileReader);
elease(videoPlayer);
```

Chapter 5

Conclusions and future work

.1 Introduction

This chapter includes conclusions and some of suggested future work.

.2 Conclusions

In our work, we created a simple face tracking system that automatically etects and tracks a single face. We can change the input video and see if we are able track a face. If we notice poor tracking results, we can check the Hue channel data see if there is enough contrast between the face region and the background.

.3 Future development

There are always a lot to do, one may leave or miss something to be done ter. The suggestions for future work can be summarized as follow:

1- Possible in the future development of the project to apply Viola-Jones face detection Algorithm in Real-Time systems using DSP (Digital Signal Processor) for DSP efficient in signal and image processing .And using this system in applications.

2- Also one may try other algorithms as Kalam filter in face detection and compare between the two approaches.

3- One of our suggested future work may be tracking multi faces in real time applications.

?eferences

1] Paul Viola, Michael J. Jones, Robust Real-Time Face Detection, International ournal of Computer Vision 57(2), 2004.

2] Paul Viola, Michael J. Jones, Fast Multi-view Face Detection, Mitsubishi lectric Research Laboratories, TR2003-096, August 2003.

3] Henry A. Rowley, ShumeetBaluja, Takeo Kanade, Neural Network-Based Face etection, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 0, Issue 1, Jan 1998. Digital Object Identifier 10.1109/34.655647.

4] Henry Schneiderman, Takeo Kanade, A Statistical Method for 3D Object etection Applied To Faces and Cars, IEEE Conference on Computer Vision and attern Recognition 2000 proceedings, Volume 1. Digital Object Identifier 0.1109/CVPR.2000.855895.

5] Christopher M. Bishop, Pattern Recognition and Machine Learning, first edition, pringer 2006.

6] P. J. Phillips, H. Moon, The Facial Recognition Technology (FERET) Database. tp://www.itl.nist.gov/iad/humanid/feret/feret_master.html

7] Gary B. Huang, Manu Ramesh, Tamara Berg, Erik Learned-Miller, Labeled Faces the Wild: A Database for Studying Face Recognition in Unconstrained nvironments, University of Massachusetts, Amherst, Technical Report 07-49, ctober 2007. http://vis-www.cs.umass.edu/lfw/index.html

3] Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language, econd edition, Prentice Hall Software Series, 1988.