A STUDY ON FACE, EYE DETECTION AND GAZE ESTIMATION

Zeynep Orman[1], Abdulkadir Battal[2] and Erdem Kemer[3]

Department of Computer Engineering, Istanbul University, Istanbul, Turkey
[1]ormanz@istanbul.edu.tr, {[2]erdemkemer,[3]akadirbattal}@gmail.com

ABSTRACT

Face and eye detection is one of the most challenging problems in computer vision area. The goal of this paper is to present a study over the existing literature on face and eye detection and gaze estimation. With the uptrend of systems based on face and eye detection in many different areas of life in recent years, this subject has gained much more attention by academic and industrial area. Many different studies have been performed about face and eye detection. Besides having many challenging problems like, having different lighting conditions, having glasses, facial hair or mustache on face, different orientation pose or occlusion of face, face and eye detection methods performed great progress. In this paper we first categorize face detection models and examine the basic algorithms for face detection. Then we present methods for eye detection and gaze estimation.

KEYWORDS

Image Processing, Face Detection, Eye Detection, Gaze Estimation

1. INTRODUCTION

Face detection is one of the most challenging problems in disciplines such as image processing, pattern recognition and computer vision. With the improvements in information technology, face detection / recognition has wide usage in applications such as personal identity, video surveillance, witness face reconstruction, computerized aging, control systems (like tracing fatigue of drivers to avoid traffic accidents), HCI (human computer interaction, which provides control of computer based system, with no need of interaction through usage of mouse, keyboard etc.), video game controllers. Therefore, implementations of face and gaze detection have received a great deal of attention in the recent literature [1-7].

For last 15 years, face detection has become a popular subject for researchers in psychophysics, neural sciences (especially in uniqueness of faces; whether face recognition is done holistically or by local feature analysis) and engineering, image processing, analysis and computer vision. A formal method of classifying faces was first proposed by Francis Galton in 1888. He proposed a method that collects facial profiles as curves and then finds their norms. This method was able to classify other profiles by using their deviations from the norms. The classification was a multi-modal resulting in a vector of independent measures that could be compared with other vectors in a database.

Face recognition is a very challenging task because of variability of features in the photo taken. Different variations of scaling the faces in image, location, orientation of images (like rotated or not), pose of faces (like frontal, by side or profile) make face recognition difficult to achieve to build a performance system with practical usage for needs mentioned above.

Mentioned in Yang's survey [8], the main challenges associated with face detection are;

DOI : 10.5121/ijcses.2011.2303

Pose: The image of face can vary due to relative camera-face pose, like; frontal, 45 degree, profile, and upside-down. Also some features of face (like eyes, mouth ...) may be partially or wholly occluded.

Presence or absence of structural components: Facial features such as beards, mustaches, and glasses may or may not be present and there is a great deal of variability among these components including shape, color, and size.

Facial expression: The appearances of faces are directly affected by a person's facial expression.

Occlusion: Faces may be partially occluded by other objects. In an image with a group of people, some faces may partially occlude other faces.

Image orientation: Face images directly vary for different rotations about the camera's optical axis.

Imaging conditions: When the image is formed, factors such as lighting (spectra, source distribution and intensity) and camera characteristics (sensor response, lenses) affect the appearance of a face.

2. FACE RECOGNITION

There are more than 100 different techniques for face detection in images. Most of these methods use the same techniques/analysis for feature extraction. To enable better understanding of these techniques, they can be categorized into four different categories regarding their analysis methods.

2.1. Knowledge-Based Methods

These rule-based methods encode human knowledge of what constitutes a typical face. Usually, the rules capture the relationships between the facial features. These methods are designed mainly for face localization.

Knowledge based methods can be studied by two different approaches; which are top-down methods and bottom-up methods.

2.1.1. Top-Down Methods

In this approach, methods use rules to describe features of a face derived from knowledge of human face. For example, a face image is consist of, two eyes that are symmetric to each other, a nose and a mouth. The relationships between features can be represented by their relative distances and positions.

Problem with this approach is; it is difficult to translate human knowledge into well-defined rules. If this method chooses to use strict rules, than it may fail while detecting faces because it may not pass all the rules defined. But on the other hand if the rules are too general than there may be many false detections.

2.1.1.1. Researches Using This Approach

One popular work about this approach was performed by Yang and Huang [9]. They used a hierarchical knowledge-based method to detect faces. Their system consists of three levels of rules. At the highest level, all possible face candidates are found by scanning a window over the input image and applying a set of rules at each location. The rules at higher level are general descriptions of what a face looks like while the rules at lower levels rely on details of facial features. A multi-resolution hierarchy of images is created by averaging and subsampling. (Figure 1) [8]

(a) (b) (c) (d)

Figure 1: Original and corresponding low resolution images [8]

In figure 2 [8], examples of the coded rules, which are based on the characteristics of human face, used to locate the face candidates in the lowest resolution include: "the center part of the face (Region-1) has a four cells with basically uniform intensity", "the upper round part of a face (Region-2) has a basically uniform intensity" and "the difference between the average gray values of the center part and the upper round part is significant". The lowest resolution (Level 1) image is searched for face candidates and these are further processed at finer resolutions. At Level 2, local histogram equalization is performed on the face candidates received from Level 1, followed by edge detection. Surviving candidate regions are then examined at Level 3 with another set of rules that respond to facial features such as the eyes and the mouth.

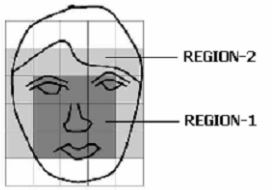

Figure 2: A typical face used in knowledge-based top-down methods [8]

Yang's method does not result in a high detection rate but later work [21] used the idea of multi-resolution images and rules based searching for face detection in frontal views.

2.1.2. Bottom-Up Methods

In this approach, it has been tried to find variant features of faces for detection. The underlying assumption is based on the observation that humans can effortlessly detect faces and objects in different poses and lighting conditions and, so, there must be existing properties or features which are invariant over this variability. Numerous methods have been proposed that first detect facial features and then infer the presence of a face. Facial features such as eyebrows, eyes, nose, mouth, and hair-line are commonly extracted using edge detectors. Based on the extracted features, a statistical model is built to describe their relationships and to verify the existence of a face. One problem with these feature-based algorithms is that the image features can be severely corrupted due to illumination, noise, and occlusion. Feature boundaries can be weakened for faces, while shadows can cause numerous strong edges which together render perceptual grouping algorithms useless.

General methods for face detection with bottom-up approach are; using Facial Features (using combination of features locations, existing on face), using texture (using distinct texture of face that can be used to separate face from other objects), using skin color (not using only chrominance of a color, also by using intensity of a color, even multi-racial faces can be detected) and using multiple features (which utilizes global features such as skin color, size and shape to find face candidates, and then verify these candidates using local, detailed features such as eye brows, nose and hair.)

2.1.2.1. Researches Using This Approach

Recent work using this approach is done by Amit Sharma. The method is developed for shape detection and is applied to detect frontal-view faces in still intensity images [10]. Detection follows two stages: focusing and intensive classification. Focusing is based on spatial arrangements of edge fragments extracted from a simple edge detector using intensity difference. A rich family of such spatial arrangements, invariant over a range of photometric and geometric transformations, is defined. From a set of 300 training face images, particular spatial arrangements of edges which are more common in faces than backgrounds are selected using an inductive method developed. Mean-while, the CART algorithm [11] is applied to grow a classification tree from the training images and a collection of false positives identified from generic background images. Given a test image, regions of interest are identified from the spatial arrangements of edge fragments. Each region of interest is then classified as face or background using the learned CART tree. Their experimental results on a set of 100 images report a false positive rate of 0.2 percent per 1,000 pixels and a false negative rate of 10 per cent.

2.2. Template Matching Methods

In template matching, face recognition is parameterized by using a function for a standard face pattern (like frontal taken face image). Several sub templates are used for a given input image and the correlation values with the standard patterns are computed for the face contour, eyes, nose and mouth independently. The existence of a face is determined based on these correlation values.

This approach has the advantage of being simple to implement. However, it has proven to be inadequate for face detection since it cannot effectively deal with variation in scale, pose and shape. Multi-resolution, multi-scale, sub templates and deformable templates have subsequently been proposed to achieve scale and shape invariance.

Template matching researches can be separated into two subcategories which are, researches using predefined templates and the other is researches using deformable templates.

2.2.1. Predefined Templates

General idea with using predefined templates consists of two phases. First face separates face using templates and reveals candidate locations for second phase. And in second phase, these areas are focused in detail to determine the existence of a face.

2.2.1.1. Researches Using This Approach

A hierarchical template matching method for face detection was proposed by Miao et al. [19]. At the first stage, an input image is rotated from -20^o to 20^o in steps of 5o, in order to handle rotated faces. A multi-resolution image hierarchy is formed and edges are extracted using the Laplacian operator. The face template consists of the edges produced by six facial components: two eyebrows, two eyes, one nose, and one mouth. Finally, heuristics are applied to determine the existence of a face. Their experimental results show better results in images containing a single face (frontal or rotated) than in images with multiple faces.

2.2.2. Deformable Templates

In this approach, facial features are described by parameterized templates. An energy function is defined to link edges, peaks, and valleys in the input image with their corresponding parameters in the template. The best fit of the elastic model is found by minimizing an energy function of the parameters. Although their experimental results demonstrate a good performance in tracking non-rigid features, one draw-back of this approach is that the deformable template must be initialized in the proximity of the object of interest.

2.2.2.1. Researches Using This Approach

Lanitis et al. described a face representation method with both shape and intensity information [12].They start with sets of training images in which sampled contours such as the eye boundary, nose, chin/cheek are manually labeled, and a vector of sample points is used to represent shape. They used a point distribution model (PDM) to characterize the shape vectors over an ensemble of individuals, and an approach to represent shape- normalized intensity appearance. A face-shape PDM can be used to locate faces in new images by using active shape model (ASM) search to estimate the face location and shape parameters. The face patch is then deformed to the average shape, and intensity parameters are extracted. The shape and intensity parameters can be used together for classification.

Cootes and Taylor applied a similar approach to localize a face in an image [12]. First, they define rectangular regions of the image containing instances of the feature of interest. Factor analysis is then applied to fit these training features and obtain a distribution function. Candidate features are determined if the probabilistic measures are above a threshold and are verified using the active shape model (ASM). After training phase of this method with 40 images, it is able to locate 35 faces in 40 test images. The ASM approach has also been extended with two Kalman filters to estimate the shape-free intensity parameters and to track faces in image sequences [13].

2.3. Appearance Based Methods

In this approach the templates are learned from examples in images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face and non-face images. The learned characteristics are in the form of distribution models or discriminant functions that are consequently used for face detection. Meanwhile, to overcome performance issues, dimensionality reduction is usually carried out.

There are many approaches using the appearance based methods. Mostly used and define approaches in Yang's survey [8] are, eigenfaces, distribution-based methods, neural networks, support vector machines, sparse network of winnows, Naïve Bayes classifiers, hidden Markov model, information theoretical approach and inductive learning.

Mostly used approaches are explained below.

2.3.1. Inductive Learning

Idea for face recognition with inductive learning is to build a decision tree from positive and negative examples of face patterns. Each training example is an 8 x 8 pixel window and is represented by a vector of 30 attributes which is composed of entropy, mean, and standard deviation of the pixel intensity values. From these examples, building a classifier as a decision tree whose leaves indicate class identity and whose nodes specify tests to perform on a single attribute. The learned decision tree is then used to decide whether a face exists in the input example. The experiments show a localization accuracy rate of 96 percent on a set of 2,340 frontal face images in the FERET data set.

2.3.2. Eigenfaces

Eigenfaces approach is a principal component analysis (PCA) based approach for detecting faces. In this approach, faces are represented as points in a space called eigenspace. Face detection is performed by computing the distance of image windows to the space of faces. Assuming raw images as points in a high dimensional space is impractical for two reasons; firstly, working in very high dimensional spaces is computationally too expensive, and secondly, raw images contain statistically irrelevant data which degrades the performance of the system. PCA reduces the dimensionality of a feature space by restricting the attention to those directions along which the scatter is greatest [37]. Thus, PCA is used for projecting the raw face

images onto the eigenspace of a representative set of normalized face images, for eliminating redundant and irrelevant information.

Many works on face detection, recognition, and feature extractions have adopted the idea of eigenvector decomposition and clustering.

2.3.2.1. Researches Using This Approach

Turk and Pentland applied PCA to face recognition and detection [14]. PCA on a training set of face images is performed to generate the eigenfaces which span a subspace of the image space. Images of faces are projected onto the subspace and clustered. Similarly, non-face training images are projected onto the same subspace and clustered. Since images of faces do not change radically when projected onto the face space, while the projection of non-face images appear quite different. To detect the presence of a face in a scene, the distance between an image region and the face space is computed for all locations in the image. The distance from face space is used as a measure of "faceness" and the result of calculating the distance from face space is a "face map." A face can then be detected from the local minima of the face map. Many works on face detection, recognition, and feature extractions have adopted the idea of eigenvector decomposition and clustering.

2.3.3. Distribution-Based Methods

The main idea of distribution-based methods is to prove that distributions of image patterns from one object class can be learned from positive and negative examples of that object.

It is easy to collect a representative sample face patterns, but much more difficult to get a representative sample of non-face patterns. This problem is alleviated by a bootstrap method that selectively adds images to the training set as training progress. Starting with a small set of non-face examples in the training set, the multilayer perceptron (MLP) classifier is trained with this database of examples. Then, they run the face detector on a sequence of random images and collect all the non-face patterns that the current system wrongly classifies as faces. These false positives are then added to the training database as new non-face examples. This bootstrap method avoids the problem of explicitly collecting a representative sample of non-face patterns and has been used in later works [15].

2.3.4. Support Vector Machines

In SVM approach, the idea is to train polynomial function, neural networks, or radial basis function classifiers. While most methods for training a classifier are based on of minimizing the training error, SVMs operates on another induction principle, called "structural risk minimization", which aims to minimize an upper bound on the expected generalization error. An SVM classifier is a linear classifier where the separating hyper-plane is chosen to minimize the expected classification error of the unseen test patterns. This optimal hyper-plane is defined by a weighted combination of a small subset of the training vectors, called support vectors. Estimating the optimal hyper-plane is equivalent to solving a linearly constrained quadratic programming problem. However, the computation is both time and memory intensive [15]. SVMs have also been used to detect faces and pedestrians in the wavelet domain.

2.3.4.1. Researches Using This Approach

Osuna et al. developed an efficient method to train an SVM for large scale problems, and applied it to face detection[15]. Based on two test sets of 10,000,000 test patterns of 19x19 pixels, their system has slightly lower error rates and runs with very high performance compared to similar methods.

2.3.5. Hidden Markov Model

The underlying assumption of the Hidden Markov Model (HMM) is that patterns can be characterized as a parametric random process and that the parameters of this process can be estimated in a precise, well-defined manner. In developing an HMM for a pattern recognition problem, a number of hidden states need to be decided first to form a model. Then, one can train HMM to learn the transitional probability between states from the examples where each example is represented as a sequence of observations. The goal of training an HMM is to maximize the probability of observing the training data by adjusting the parameters in an HMM model with the standard Viterbi segmentation method. After the HMM has been trained, the output probability of an observation determines the class to which it belongs.

Face pattern composed in an image can be divided into several regions such as the forehead, eyes, nose, mouth, and chin. A face pattern can then be recognized by a process in which these regions are observed in an appropriate order (from top to bottom and left to right). Instead of relying on accurate alignment as in template matching or appearance-based methods (where facial features such as eyes and noses need to be aligned well with respect to a reference point), this approach aims to associate facial regions with the states of a continuous density Hidden Markov Model. HMM-based methods usually treat a face pattern as a sequence of observation vectors where each vector is a strip of pixels. During training and testing, an image is scanned in some order (usually from top to bottom) and an observation is taken as a block of pixels. For face patterns, the boundaries between strips of pixels are represented by probabilistic transitions between states, and the image data within a region is modeled by a multivariate Gaussian distribution. An observation sequence consists of all intensity values from each block. The output states correspond to the classes to which the observations belong. After the HMM has been trained, the output probability of an observation determines the class to which it belongs. HMMs have been applied to both face recognition and localization. HMM is trained for a generic model of human faces from a large collection of face images. If the face likelihood obtained for each rectangular pattern in the image is above a threshold, a face is located.

2.3.5.1. Researches Using This Approach

Samaria and Young applied 1D and pseudo 2D HMMs to facial feature extraction and face recognition [16]. Their HMMs exploit the structure of a face to enforce constraints on the state transitions. Since significant facial regions such as hair, forehead, eyes, nose, and mouth occur in the natural order from top to bottom, each of these regions is assigned to a state in a one-dimensional continuous HMM. For training, each image is uniformly segmented, from top to bottom into five states (each image is divided into five non-overlapping regions of equal size). The uniform segmentation is then replaced by the Viterbi segmentation and the parameters in the HMM are re-estimated using the Baum-Welch algorithm. Each face image of width W and height H is divided into overlapping blocks of height L and width W. There are P rows of overlap between consecutive blocks in the vertical direction. These blocks form an observation sequence for the image, and the trained HMM is used to determine the output state.. Instead of using raw intensity values, the observation vectors consist of the (KLT) coefficients computed from the input vectors. On the MIT database, which contains 432 images each with a single face, this pseudo 2D HMM system has a success rate of 90 percent.

2.3.6. Neural Networks

Neural networks have been applied successfully in many pattern recognition problems, such as optical character recognition, object recognition, and autonomous robot driving. Since face detection can be treated as a two class pattern recognition problem, various neural network architectures have been proposed. The advantage of using neural networks for face detection is the feasibility of training a system to capture the complex class conditional density of face

patterns. However, one drawback is that the network architecture has to be extensively tuned (number of layers, number of nodes, learning rates, etc.) to get exceptional performance [8].

2.3.6.1. Researches Using This Approach

The most significant work done with neural network is arguably done by Rowley et al. [17]. A multilayer neural network is used to learn the face and non-face patterns from face/non-face images (i.e., the intensities and spatial relationships of pixels). There are two major components: multiple neural networks (to detect face patterns) and a decision-making module (to render the final decision from multiple detection results). The first component of this method is a neural network that receives a 20x20 pixel region of an image and outputs a score ranging from -1 to 1. Given a test pattern, the output of the trained neural network indicates the evidence for a non-face (close to -1) or face pattern (close to 1). To detect faces anywhere in an image, the neural network is applied at all image locations. To detect faces larger than 20x20 pixels, the input image is repeatedly subsampled, and the network is applied at each scale. Nearly 1,050 face samples of various sizes, orientations, positions, and intensities are used to train the network. In each training image, the eyes, tip of the nose, corners, and center of the mouth are labeled manually and used to normalize the face to the same scale, orientation, and position. The second component of this method is to merge overlapping detection and arbitrate between the outputs of multiple networks. Simple arbitration schemes such as logic operators (AND/OR) and voting are used to improve performance. Rowley et al. [17] reported several systems with different arbitration schemes that are less computationally expensive than Sung and Poggio's system and have higher detection rates based on a test set of 24 images containing 144 faces.

One limitation of the methods by Rowley [17] is that they can only detect upright, frontal faces. Recently, Rowley et al. [18] extended this method to detect rotated faces using a router network which processes each input window to determine the possible face orientation and then rotates the window to a canonical orientation; the rotated window is presented to the neural networks as described above. However, the new system has a lower detection rate on upright faces than the upright detector. Nevertheless, the system is able to detect 76.9% of faces over two large test sets with a small number of false positives.

3. EYE AND GAZE DETECTION

The eye-gaze tracking has become one of the most important human-computer interfaces and it has been shown to be useful in diverse applications. The eye-gaze tracking is the process of measuring either the point of gaze or the motion of an eye relative to the head. Eye tracking is for measuring eye positions and eye movement.

There are many areas that benefit from eye tracking systems. Specific applications include these systems in language reading, music reading, human activity recognition, the perception of advertising, playing of sport, HCI (especially for handicap people suffering from diseases), medical research and other areas.

One of the most promising applications of eye tracking research is in the field of automotive design. Research is currently underway to integrate eye tracking cameras into automobiles. The goal of this endeavor is to provide the vehicle with the capacity to assess in real-time the visual behavior of the driver. The National Highway Traffic Safety Administration (NHTSA) estimates that drowsiness is the primary causal factor in 100,000 police-reported accidents per year. Another NHTSA study suggests that 80% of collisions occur within three seconds of a distraction. By equipping automobiles with the ability to monitor drowsiness, inattention, and cognitive engagement driving safety could be dramatically enhanced. Lexus claims to have equipped its LS 460 with the first driver monitor system in 2006, providing a warning if the driver takes his or her eye off the road.

Eye trackers are used in research on the visual system, in psychology, in cognitive linguistics and in product design. There are a number of methods for measuring eye movement. The most popular variant uses video images from which the eye position is extracted.

Studies on eye can be classified as below:

Detection of eye: Given an arbitrary face image, the goal of eye detection is to determine the location of the eyes. Simply in eye detection, the areas where both eyes are located are found or two eyes individually localized. As a result of the process usually eye areas are indicated by a rectangle.

Detailed feature extraction: On the other hand the goal of this category is to give detailed information such as the contour of the visible eyeball region, circular area formed by iris and pupil, location of pupil in the visible eye area, state of the eye (blink/not blink). This type of work is more difficult in computer vision area as detection or real-time tracking of small details are highly effected from varying ambient conditions and result may easily fail.

A lot of work on eye detection area such as; eye pupil movement detection, eye feature extraction, eye state detection, eye gaze detection using different techniques both in still images and in video sequences for real-time applications.

There are many different approaches for eye/gaze detection, some of methods need extra hardware support to accomplish the task and for others, just a simple webcam is enough to do detection. Various methods are described below.

3.1. Electrooculography

Electrooculography (EOG) is a technique for measuring the resting potential of the retina. There is a permanent potential difference between the cornea and the fundus of approximately 1mV, small voltages can be recorded from the region around the eyes which vary as the eye position varies. Pairs of electrodes are placed either above and below the eye or to the left and right of the eye. By carefully placing electrodes it is possible to separately record horizontal and vertical movements. If the eye is moved from the center position towards one electrode, this electrode "sees" the positive side of the retina and the opposite electrode "sees" the negative side of the retina. Consequently, a potential difference occurs between the electrodes. Assuming that the resting potential is constant, the recorded potential is a measure for the eye position. However, the signal can change when there is no eye movement. It is dependent on the state of dark adaption (used clinically to calculate the Arden ratio as a measure of retinal health), and is affected by metabolic changes in the eye. It is prone to drift and giving spurious signals, the state of the contact between the electrodes and the skin produces and other source of variability. There have been reports that the velocity of the eye as it moves may itself contribute an extra component to the EOG. It is not a reliable method for quantitative measurement, particularly of medium and large saccades. However, it is a cheap, easy and non-invasive method of recording large eye movements, and is still frequently used by clinicians. The eye acts as a dipole in which the anterior pole is positive and the posterior pole is negative. 1. Left gaze; the cornea approaches the electrode near the outer canthus resulting in a positive-going change in the potential difference recorded from it. 2. Right gaze; the cornea approaches the electrode near the inner canthus resulting in a positive-going change in the potential difference recorded from it (A, an AC/DC amplifier).

3.2. Infra-Red Oculography

If a fixed light source is directed at the eye, the amount of light reflected back to a fixed detector will vary with the eye's position. This principle has been exploited in a number of commercially available eye trackers. Infra-red light is used as this is "invisible" to the eye, and doesn't serve

as a distraction to the subject. As infra-red detectors are not influenced to any great extent by other light sources, the ambient lighting level does not affect measurements. Spatial resolution (the size of the smallest movement that can reliably be detected) is good for this technique, it is of the order of 0.1°, and temporal resolutions of 1ms can be achieved. It is better for measuring horizontal than vertical eye movements. Blinks can be a problem, as not only do the lids cover the surface of the eye, but the eye retracts slightly, altering the amount of light reflected for a short time after the blink.

3.3. Scleral search coils

When a coil of wire moves in a magnetic field, the field induces a voltage in the coil. If the coil is attached to the eye, then a signal of eye position will be produced. In order to measure human eye movements, small coils of wire are embedded in a modified contact lens or annulus. This is inserted into the eye after local anesthetics has been introduced. A wire from the coil leaves the eye at the temporal canthus. The field is generated by two field coils placed either side of the head. This allows horizontal eye movement to be recorded. If it is necessary to also monitor vertical eye movements, then a second set of field coils, usually set orthogonally to the first set, it used. The two signals (one for horizontal, one for vertical eye movement) generated in the eye coil can then be disentangled using appropriate electronics. If the eye coil is of an appropriate design, then torsional movements can also be recorded. In experiments on eye movements in animals, the eye coils are frequently implanted surgically. The advantage of this method is that it has a very high temporal and spatial resolution allowing even the smaller types of eye movements (e.g. micrsaccades) to be studied. Its disadvantage is that it is an invasive method, requiring something to be placed into the eye. This method is rarely used clinically, but is an invaluable research tool.

3.4. Image based methods

With the development of video and image analysis technology, different methods that automatically extract the eye position from images of the eye have been developed. In some systems a bright light source is used to produce "Purkinje" images, these are reflection of the light source from various surfaces in the eye (the front and back surfaces of the cornea and lens). Tracking the relative movements of these images gives an eye position signal. More commonly a video image is combined with computer software to calculate the position of the pupil and its center. This allows vertical and horizontal eye movements to be measured. However, image based methods tend to have temporal resolutions lower that that achieved with IR techniques. Spatial resolution can also be limited. As technology improves, the resolutions these systems can deliver will also improve.

There are many proposed methods for eye detection, such as the ones using template matching, eigenspace, and integral projection. R. Brunelli [20] detected eyes using template matching, which is to search an image to find the highest similarity to the moving template image. But the template matching is very sensitive to face angle and facial expression. A. Pentland et al. [21] used the eigenspace method to detect eyes. The eigenspace method shows better performance than the template matching method, but its performance is largely dependent on the choice of training database set. Z. H. Zhou et al. [22] used the HPF (Hybrid Projection Function) to detect eyes, which achieves better performance than the existing PF (Projection Function). However, HPF is very sensitive to face angle and various lighting conditions. R. L. Hsu et al. [23] used the eye map based on color information to detect eyes. But this method often missed to detect closed eyes.

Below, mostly used image based eye/gaze detection methods are explained in detail.

3.4.1. Haar-like features

Haar-like features are digital image features used in object recognition. They were used in the first real-time face detector. Historically, working with only image intensities, made the task of feature calculation computationally expensive. It is discussed that working with an alternate feature set based on Haar wavelets instead of the usual image intensities. A Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in these regions and calculates the difference between them. This difference is then used to categorize subsections of an image. For example, let us say we have an image database with human faces. It is a common observation that among all faces the region of the eyes is darker than the region of the cheeks. Therefore a common haar feature for face detection is a set of two adjacent rectangles that lie above the eye and the cheek region. The position of these rectangles is defined relative to a detection window that acts like a bounding box to the target object (the face in this case).

The key advantage of a Haar-like feature over most other features is its calculation speed. Due to the use of integral images, a Haar-like feature of any size can be calculated in constant time (approximately 60 microprocessor instructions for a 2-rectangle feature).

3.4.1.1. Researches Using This Approach

Viola and Jones's [25] face detection algorithm, based on Haar-like features is used to detect a face. Haar-like features encode the existence of oriented contrast between regions in the image. A set of these features can be used to encode the contrast exhibited by a human face and their special relationships. Figure 1 shows four types of Haar-like features that are used to encode the horizontal, vertical and diagonal intensity information of face images at different positions and scales.

In Viola and Jones's method, a cascade of boosted classifiers (i.e. an ensemble of weak classifiers instead of one single strong classifier) working with Haar-like features is trained with a few hundred sample views of face and non-face examples, which are scaled to the same size, i.e.24x24. The motivation behind the cascade of classifier is that simple classifiers at early stage can filter out most negative examples efficiently, and stronger classifiers at later stage are only necessary to deal with instances that look like faces. After the classifier is trained, it can be applied to a region of interest in an input image. To search for the face, one can move the search window across the image and check every location using the classifier.

3.4.2. Neural Networks

As in face recognition, neural networks are also commonly used in eye/gaze detection. The artificial neural networks are capable of storing acquired knowledge to solve problems, therefore gaining new knowledge through experience [26], [27]. The systems simulate the structure of the brain, being able to develop what is called intelligence, using computers' ability to learn and through the errors, being able to recognize patterns.

With neural network, first a database is created, based on the most relevant information, especially found on the face and eyes. Using this database, it is possible to make a selection of attributes, which are important for the training and classification of the gaze direction. After being trained, the neural network is able to classify the gaze direction in real time.

3.4.2.1. Researches Using This Approach

In [28], a method for the eye detection, based on rectangle features and pixel-pattern based texture feature (PPBTF) is proposed. In [29], a method to locate eyes from face images is presented, based on multi-cue facial information. Using color characteristics is a useful way to detect the eyes according to [30].

The detection of the eyes used in [31] is an algorithm of low computational cost, which is divided into two parts.

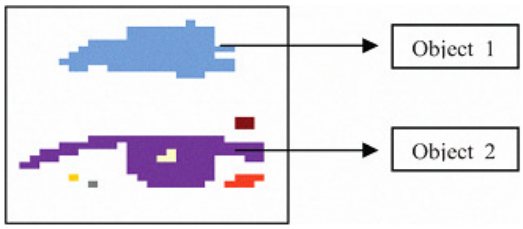

Figure 3: Identification of objects inside the eye area [31]

Based on parameters of facial geometry, an area is defined, which possibly represents the eye and eyebrow locations in addition to extra facial information. Following from that, an algorithm [32] is employed to detect the objects. This algorithm allows a separation between the eye and other objects which occasionally belong to that geometrically defined area.

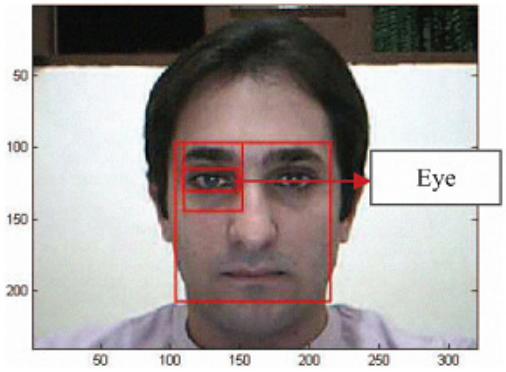

Figure 4: Detection of the eye's area [31]

Considering the two largest objects found in Figure 3[31], it is observed that object 1 is the eyebrow, whereas object 2 is the eye. Figure 4[31] shows a photograph that accurately represents the eye detection.

Finally, after finding the eye, it is found its center of mass, that will adjust the eye position in a box of previously defined size. The information contained in the box will be used later by the neural network, not only for the training database but also for the pattern classification.

3.4.3. Support Vector Machines

A support vector machine (SVM), which is a binary classification method, has been successfully applied to the detection and verification of human eyes [33]-[34]. The main idea behind SVM system is to find, the optimal linear decision surface based on the concept of structural risk minimization. The decision surface is a weighted combination of elements of the training set. These elements are called support vectors and characterize the boundary between the two classes.[35]

$$(\mathbf{x}_1, y_1), \cdots (\mathbf{x}_i, y_i), \cdots (\mathbf{x}_N, y_N) \quad \mathbf{x}_i \in \mathrm{R}^N, \; y_i \in \{-1, 1\}$$

In case of linear separable data, maximum margin classification aims to separate two classes with hyperplane that maximizes distance of supports vectors. This hyperplane is called OSH (Optimal Separating Hyperplane). OSH can be expressed as below.

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i (\mathbf{x}_i^T \mathbf{x}) + b$$

This solution is defined in terms of subset of training samples (supports vectors) whose αi is non- zero. [36]

SVM are commonly used in eye detection system. The SVM determines the presence/absence of eyes using an input vector consisting of gray values in a moving window. In the case of rotated face images, the method often fails to detect eyes because such images are inconsistent with the training image set. This does not present a problem for authentication applications with the constraint that a human face must be in an upright position. However, in order to be widely applicable to photo albums [13] and automatic video management systems [14], eye detection methods must be able to detect human eyes even in rotated faces.

3.4.3.1. Researches Using This Approach

In [36] data that is used for generating eye verification SVM consists of 400 images of each class (eye and non-eye). The image (20x10) data is preprocessed so that each pixel is normalized to a [1, +1] range before training, and Radial Basis Function (RBF) is used as kernel function.

First system detect candidates of eye pair and verify them using SVM to select one eye pair. Then extract and normalize face candidate region with center points of both eyes. Finally system detects face through verification of face candidate region using SVM. The proposed algorithm can detect face in real time because it is faster than the method which applies SVM at every location in the input image reducing size repeatedly to detect face. And in experimental results, it is demonstrated that face detection error rate significantly reduced by verifying candidates of eye pair and face candidate region using SVM.

In Figure 5, the flow diagram of the proposed face detection method using SVM is given[36].

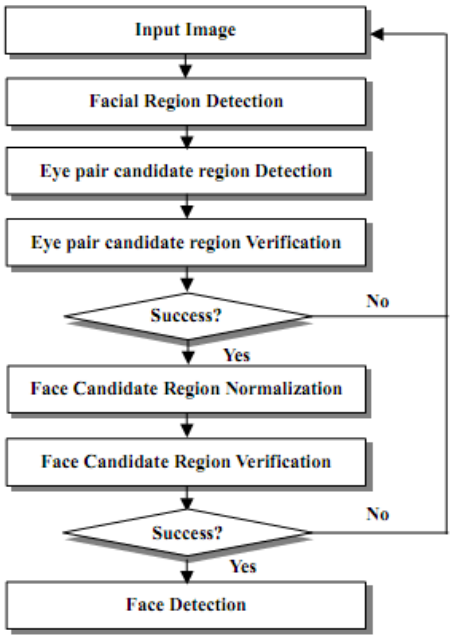

Figure 5: The proposed face detection method using SVM[36]

3.4.4. AdaBoost Algorithm

AdaBoost (Adaptive Boosting), is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost is sensitive to noisy data and outliers. However in some problems it can be less susceptible to the over-fitting problem than most learning algorithms.

AdaBoost is very popular for object detection since its first application in face detection [38]. Basically, AdaBoost selects the critical features and train weak classifiers as well as updates the sample weights [24]. As long as the weak classifiers are slightly better than random guessing, the final classifier will have much better accuracy after combining all the weak classifiers together. The main task in the AdaBoost is the selection of features to learn weak classifiers. We use more powerful discriminant features instead of rectangular Haar features to improve eye detection accuracy. Since the data weights in both discriminant analysis and AdaBoost represent the same distribution, they can be associated together.

3.4.4.1. Researches Using This Approach

In [39] adaboost algorithm is used to train a robust eye detector. Training data is collected from various sources (e.g. FERET images). More eye images are collected from the web in order to include more variance from the real world. The eyes are randomly rotated with small angles. In total, thousands of eyes have been collected for training. In application, only a left eye detector is trained due to the symmetry of eyes. In detection, the images are flipped to find the right eyes. The non-eye images were randomly collected from background images. More non-eyes were collected from the false detections. Those false detections were fed back for training. To improve the eye detection speed, a cascade structure is applied [40]. The first layer in the cascade only has two features yet it can remove 80% of the non-eye samples. The resulting eye detector classifier uses less than 100 features.

Also in [41] an improved version of adaboost algorithm, called AD AdaBoost is used for the detection of human eye, taking into account the clarity of human eye's own outline, sharp color contrast, using a large number of human eye samples for statistical learning. Releasing classified weights of negative samples in the classifier training process correctly, and through a weight normalization to solve the problem of expansion on weights of buffering difficult samples, slow down the degradation in classification process while improving the detection accuracy.

The basic idea of AD AdaBoost is to integrate multiple weak classifiers into a strong classifier. In the training process, all of the positive and negative samples are given equal initial weights. When one classifier training is completed, based on their classification result on the training set of samples, an adjustment is made on all of the sample weights, the correctly classified samples by previous class, its weight decreases when it enters the next iteration. The mistakenly classified samples by previous class, its weight increases when it enters the next iteration, making the next weak classifier training are more concerned about samples which have been misidentified [42]. The final judgment result of strong classifier weighted sum of all judgment results of all weak classifiers.

Compared to traditional AdaBoost, AD AdaBoost effectively reduces the negative sample error rate while positive sample error rate is relatively low, improving the identification ability of overall classifiers.

4. CONCLUSIONS

Many methods for eye detection use face location in images. So prior to study on eye detection, a review of face detection methods is needed. In this paper, various face detection methods are categorized in analysis aspects. They have advantages and disadvantages compared to each other, but the main problems stay still in all face recognition methods. These problems are: different lighting conditions; orientation, pose and partial occlusion of face; facial expressions – make-up; presence of glasses, facial hair or mustache. Many methods can overcome one or more of these difficulties, but there is no silver bullet to overcome all the problems and do fast detection without giving false positive results.

In our survey, we recognise that haar-like feature approach is the most used method for face recognition projects (especially for video detection of faces and eyes) in recent years, due to be able to work in real time systems with great performance.

Eye detection and gaze estimation is generally dependent on face recognition. With the help of algorithms finding location of faces in images, eye detection methods show better performance. There are new approaches that try to find eyes, without detection of faces in images which also shows great performance. Methods are generally similar for face and eye detection systems. The same algorithms can be applied to detect both faces and eyes (And even these can be applicable to other objects like cars etc.). With progress in recent studies, by using face recognition models in many areas, new intelligent systems, which will bring great comfort and ease to our life, benefit from results of these studies. In the future, we will work on developing an efficient gaze estimation technique that can be used for determining how often and which part of a billboard is being looked at in a public area for low resolution images.

REFERENCES

[1] S.C. Kuo, C.J. Lin, J.R. Liao, (2011). "3D reconstruction and face recognition using kernel-based ICA and neural networks", Expert Systems with Applications, Vol.38, pp. 5406-5415.

[2] J. Yang, X. Ling, Y. Zhu, Z. Zheng, (2008). "A face detection and recognition system in color image series". Mathematics and Computers in Simulation, Vol. 77, pp. 531–539.

[3] H. K. Ekenel , J. Stallkamp, R. Stiefelhagen, (2010). "A video-based door monitoring system using local appearance-based face models", Computer Vision and Image Understanding, Vol. 114, pp. 596–608.

[4] A. Mian, (2011). "Online learning from local features for video-based face recognition", Pattern Recognition, Vol. 44 pp. 1068–1075.

[5] C. Nitschke, A. Nakazawa, H. Takemura, (2011). "Display-camera calibration using eye reflections and geometry constraints", Computer Vision and Image Understanding, vol. 115, pp. 835–853.

[6] A. D. Santis, D. Iacoviello (2009). "Robust real time eye tracking for computer interface for disabled people", Computer Methods and Programs in Biomedicine, vol. 96.

[7] Dan Witzner Hansen, Qiang Ji ,"In the Eye of the Beholder: A Survey of Models for Eyes and Gaze" IEEE,2010

[8] M. H. Yang, N. Ahuja, (2002). "Detecting Faces in Images: A Survey". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No.1.

[9] G. Yang and T. S. Huang, (1994). "Human Face Detection in Complex Background". Pattern Recognition, vol. 27, no. 1, pp. 53-63.

[10] Y. Amit, D. Geman, and B. Jedynak, "Efficient Focusing and Face Detection," Face Recognition: From Theory to Applications, H. Wechsler, P.J. Phillips, V. Bruce, F. Fogelman-Soulie, and T.S. Huang, eds., vol. 163, pp. 124-156, 1998.

[11] L. Breiman, J. Friedman, R. Olshen, and C. Stone, Classification and Regression Trees. Wadsworth, 1984.

[12] A. Lanitis, C.J. Taylor, and T.F. Cootes, "An Automatic Face Identification System Using Flexible Appearance Models," Image and Vision Computing, vol. 13, no. 5, pp. 393-401, 1995.

[13] G.J. Edwards, C.J. Taylor, and T. Cootes, "Learning to Identify and Track Faces in Image Sequences." Proc. Sixth IEEE Int'l Conf. Computer Vision, pp. 317-322, 1998.

[14] M. Turk and A. Pentland, "Eigenfaces for Recognition," J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.

[15] E. Osuna, R. Freund, and F. Girosi, "Training Support Vector Machines: An Application to Face Detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 130-136, 1997.

[16] F. Samaria and S. Young, "HMM Based Architecture for Face Identification," Image and Vision Computing, vol. 12, pp. 537-583, 1994.

[17] H. Rowley, S. Baluja, and T. Kanade, "Neural Network-Based Face Detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 203-208, 1996.

[18] H. Rowley, S. Baluja, and T. Kanade, "Rotation Invariant Neural Network-Based Face Detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 38-44, 1998.

[19] R. E. Schapire, A brief introduction to boosting, Proc. of the Six- teenth International Joint Conference on Artificial Intelligence, 1999, pp. 246–252

[20] R. Brunelli, T. Poggio, "Face Recognition: Features Versus Templates", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, No. 10, pp. 1042-1052, Oct. 1993.

[21] A. Pentland, B. Moghaddam, and T. Starner, "View-based and Modular Eigenspaces for Face Recognition", In IEEE Conference on Computer Vision and Pattern Recognition, pp. 84-91, 1994.

[22] Z. H. Zhou and X. Geng, "Projection Functions for Eye Detection", Pattern Recognition, Vol. 37, No. 5, pp. 1049-1056, 2004.

[23] R. L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face Detection in Color Images", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, pp. 696-706, May 2002.

[24] J. Miao, B. Yin, K. Wang, L. Shen, and X. Chen, "A Hierarchical Multiscale and Multiangle System for Human Face Detection in a Complex Background Using Gravity-Center Template," Pattern Recognition, vol. 32, no. 7, pp. 1237-1248, 1999.,

[25] P. Viola and M. Jones, "Robust real time object detection," in Proc. of the 2nd International Workshop on Statistical and Computational Theories of Vision-Modeling, Learning, Computing and Sampling, Vancouver, Canada, July 2001.

[26] K. Grauman, M. Betke, J. Gips, G. R. Bradski, "Communication via Eye Blinks - Detection and duration analysis in real time". Proc. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Lihue, HI, Vol. 1 pp.:I-I0I0 - 1-1017 2001.

[27] S. Haykin, Neural Networks A Comprehensive Foundation. Second Edition, 2005.

[28] L. Huchuan ,Yo Z. Wei, Y. Del. "Eye detection based on rectangle features and pixel-pattern based Texture features". Department of Electronic Engineering, Dalian University of Technology, 2007.

[29] G. Yepeng. "Robust Eye Detection from Facial Image based on Multicue Facial Information". School of Communication and Information Engineering Shanghai University, 2007

[30] N. A. Jalal, K. Sara, P. R. Hamid. "Eye Detection Algorithm on Facial Color Images". Department of Computer Engineering, Ferdowsi University of Mashhad and Department of Computer Engineering, University, 2008

[31] Helton M. Peixoto, Ana M. G. Guerreiro, Adrian D. D. Neto , "Image Processing for Eye Detection and Classification of the Gaze Direction", Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June 14-19, 2009

[32] M. R. Haralick, S. G. Linda. "Computer and Robot Vision", Volume I, Addison-Wesley, pp. 28-48, 1992.

[33] E. Osuna, R. Freund, and F. Girosit, "Training Support Vector Machines: An Application to Face Detection," Proc. CVPR, 1997, pp. 130-136.

[34] G. Pan, W. Lin, Z. Wu, Y. Yang, "An Eye Detection System Based on SVM Filter," Proc. SPIE, vol. 4925, 2002, pp. 326-331.

[35] Hyoung-Joon Kim and Whoi-Yul Kim , "Eye Detection in Facial Images Using Zernike Moments with SVM", ETRI Journal, Volume 30, Number 2, April 2008

[36] Hyungkeun Jee 1, Kyunghee Lee 2, Sungbum Pan , "Eye and Face Detection using SVM", IEEE 2004

[37] Robust real-time object detection, Interntional workshop on statistical and computational theories of vision 57 (2004), no. 2, 137–154

[38] Shiguang Shan, Wen Gao, Yizheng Chang, Bo Cao, and Pang Yang, Review the strength of gabor features for face recognition from the angle of its robustness to mis-alignment, International Conference on Pattern Recognition, 2004, pp. 338–341.

[39] Peng Wang, Matthew B. Green, Qiang Ji , James Wayman , "Automatic Eye Detection and Its Validation" IEEE ,2005

[40] Paul Viola and Michael Jones, Robust real-time object detection, International Journal of Computer Vision 57 (2004), no. 2, 137–154.

[41] Benke Xiang , Xiaoping Cheng ,"Eye Detection Based on Improved AD AdaBoost Algorithm", 2010 2nd International Conference on Signal Processing Systems, 2010 IEEE

[42] LI Chuang, DING Xiao-qing and WU You-shou, "A Revised AdaBoost Algorithm-AD AdaBoost" CHINESE JOURNAL OF COMPUTERS, vol 30, Jan. 2007, pp. 103-109.

Zeynep Orman received the B.Sc., M.Sc. and Ph.D. degrees from Istanbul University, Istanbul, Turkey, in 2001, 2003 and 2007, respectively. She has studied as a postdoctoral research fellow in the Department of Information Systems and Computing, Brunel University, London, UK in 2009. She is currently working as an Assistant Professor in the Department of Computer Engineering, Istanbul University. Her research interests are artificial neural networks, nonlinear systems, image processing applications and intelligent systems.

Erdem KEMER is currently studying his Master of Engineering degree in Computer Science & Engineering at Istanbul University. His major research interests are Artificial Intelligence and Software Development.

Abdulkadir BATTAL is currently studying his Master of Engineering degree in Computer Science & Engineering at Istanbul University. His major research interests are Artificial Intelligence and Software Development.