The Instrumental System for Face Recognition Tasks

Rauf Rh. Sadykhov

Belarusian State University of Informatics and Radioelectronics
P. Browka 6, Minsk 220013 Republic of Belarus
rsadykhov@bsuir.by

Sergei Tulyakov

Belarusian State University of Informatics and Radioelectronics
P. Browka 6, Minsk 220013 Republic of Belarus
sergei.tulyakov@gmail.com

Abstract

In this paper we present our work on upright frontal face recognition system. The aim of our system is to recognize faces on machine readable travel documents (MRTD). Our goal is to create a system which will be able to handle large image databases. In order to increase detection speed and hit rate we introduce some heuristics. For recognition purposes eigenface approach is used. We introduce a measure to estimate how well eigenface basis is constructed.

1 Introduction

During two last decades the technologies of face detection and recognition have attracted considerable research interest. These technologies are used in biometric systems, border control systems, video surveillance, human-computer interface, access control systems, face expression recognition, content based image retrieval. One can even find websites that allow to determine one's similarity with celebrities.

The human ability to recognize faces is remarkable. We can easily recognize faces that have seen only several times. Such skills are quite robust to time, face expression, pose, and other distractions such as presence or absence of beard, mustache, glasses, traces of injuries and surgical operations.

However, the tasks of face detection and face recognition are serious problems for automation due to many factors such as orientation, imaging conditions, presence or absence of structural components such as beards, mustache, glasses (Yang 2004).

Methods to detect faces can be divided into four main groups: (i) knowledge-based methods, (ii) feature invariant approaches, (iii) template matching methods and (iv) appearance-based methods (Yang 2004). Knowledge-based methods use expert knowledge to encode typical face. Feature invariant approaches are aimed to find facial features even when imaging conditions (pose, viewpoint or illumination) vary. Template matching methods store several standard patterns to describe a whole face or facial features separately.

Nowadays the appearance-based methods gained most popularity due to their hit rate and speed (Yang 2004). These methods don't use a priori knowledge in the data that is present on the image. Instead, they try to extract different variation modes of the database and provide a set of subclasses which represent them best. Appearance-based methods are based on scanning the input image on different scales with fixed window size in order to find faces. Each window then is given as an input to some classifier, trained to separate two classes of objects (face/non-face).

There are many models that can be used as a classifier. To separate faces and non-faces Osuna et al. and later Romdhani et al. use approach based on support vector machines (Osuna et al. 1997, Romdhani et al. 2001). Schneiderman and Kanade proposed to apply a statistical methods to the problem of face detection (Schneiderman et al. 2000).

Rowley et al. present a face detection system based on artificial neural network (Rowley et al. 1998). The neural network contains three types of hidden units: one set of units for quadrants of the 20x20 image, one set for quadrants of the quadrants, and one set for looking at overlapping horizontal strips of the image. This approach has become an early standard in face detection.

Later Viola and Jones present a much faster detector than any of their contemporaries (Viola et al. 2004). They use rectangular features instead of using pixels directly. The reason for this is that feature-based systems operate much faster. In order to compute rectangular features very fast they introduced an intermediate representation for the image called integral image. This integral image is used as an input for the cascade of weak classifiers. As a learning algorithm they use AdaBoost learning (Freund et al. 1996).

Our approach is based on cascade of weak classifiers. In order to increase detection speed and hit rate we introduce some heuristics that will be discussed in the next sections. Recognition stage is done using eigenface approach (Turk et al. 1991).

In the next section our approach on face detection and recognition is discussed. The system flow chart is given. In section 2.1 the algorithm of feature detection is discussed. Section 2.2 shows image processing that must be done to move to the identification stage that is discussed in the section 2.3. In this section we introduce a way to compare different eigenface bases. Section 3 contains conclusions and priority research directions.

2 Face recognition on machine readable documents

In our work we apply already developed methods along with suggestions to improve speed, hit rate and false hit rejection on machine readable travel documents:

> *Machine readable travel document (MRTD): Official document, conforming with the specifications contained in Doc 9303, issued by a State or organization which is used by the holder for international travel (e.g. passport, visa, official document of identity) and which contains mandatory visual (eye readable) data and a separate mandatory data summary in a format which is capable of being read by machine. (Doc 9303, 2006)*

In Doc 9303 are given requirements to MRTD portraits. Figure 1 shows some examples with only the last photo in each row fulfilling Doc 9303 requirements.

Our aim is to design a system that will be applicable to large databases of images. It is the reason why we need to find faces and facial features (eyes, nose, mouth) the fastest way.

The proposed face recognition program consists of several stages (see Figure 2). After input image is loaded, system starts feature detection stage, which consists of face search and eye search. If this stage is successfully completed, system moves to the next stage, otherwise the image has poor quality or it does not conform to the ICAO requirements, and the system prompts the user to provide another image.

Figure 1: Some portrait requirements

Next, the image processing starts. This stage ends up with storing image prepared for identification in the database. Identification stage uses known images to determine the most similar image. All this stages are discussed in following sections.

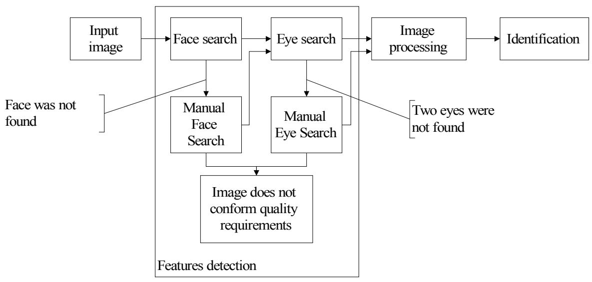


Figure 2: Structure of our computer face recognition system

2.1 Features detection

In our system we use Viola and Jones, cascade classifier due to its speed, hit rate and robustness to distractions. In their work they use a set of rectangular features (Viola et al. 2004). The sum of the pixels which lie within the white rectangles is subtracted from the sum of pixels in black rectangles (see figure 3).

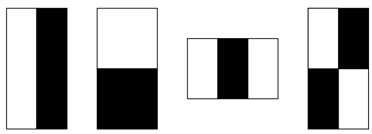


Figure 3: Examples of rectangular features, proposed by Viola and Jones.

They show that these rectangular features can be computed very rapidly using an intermediate representation of the image which they call integral image, and which at location *x, y* contains the sum of pixels above and to the left of *x, y* inclusive:

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

where *ii(x, y)* is the integral image and *i(x,y)* is the original image. They use the following pair of recurrencies to compute the integral image very rapidly:

$$\begin{aligned} s(x,y)&=s(x,y-1)+i(x,y) \\ ii(x,y)&=ii(x-1,y)+s(x,y) \end{aligned},$$

where *s(x,y)* is the cumulative row sum, *s(x,-1)=0*, and *ii(-1,y)=0.* Figure 4 shows two first features that where selected by AdaBoost. The first feature (Figure 4, B) measures intensity between the eyes and upper cheeks. The second feature (Figure 4, C) measures intensity between eyes regions and the bridge of the nose.

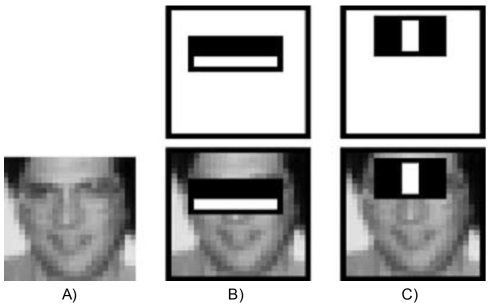


Figure 4: Rectangular features that were selected by AdaBoost.

Later R. Lienhart et al. propose an extended set of the Haar-like features (Lienhart et al. 2002). Figure 5 shows this extended feature set.

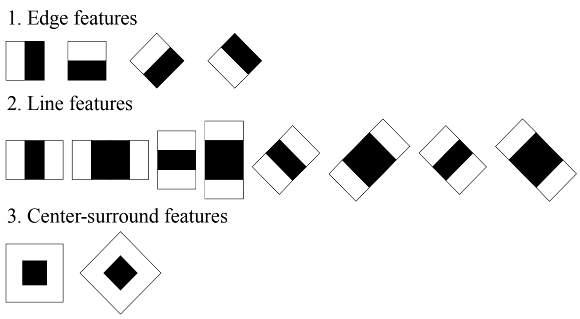


Figure 5: Extended set of Haar-like features

The aim of this stage is to find candidates to be faces and find eyes on them. Our goal is to find the best candidate for face on the whole image. As mentioned before appearance-based methods scan input image at some scale levels with fixed-size window. So if region represents a face it is found several times. If we know exactly that there is only one face in the image, we should choose region with maximum number of neighbors.

Source image A) Face-candidates found B)

Figure 6: False hit rejection

Image B in figure 6 contains two face-candidates. Left one was found 15 times, while right one was found 178 times (white label in the upper-left corner of the rectangle). Thus, as we know, that there is only one face on MRTD, we should consider right face-candidate to be a face. Alternatively, if we use “find the biggest face” strategy, we would also be able to reject false hit.

We have run experiments to find out is it better to search the biggest face or to find the candidate which has most neighbors in terms of speed. We used database that consists of 10 000 faces. Results of our experiments are shown in table 1.

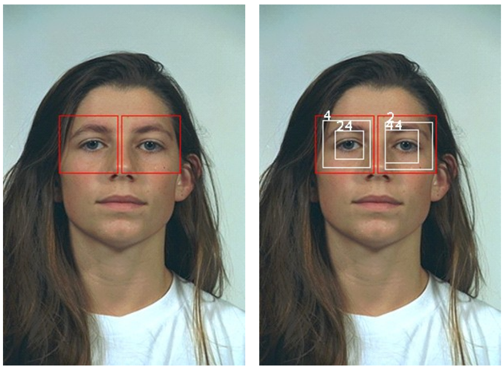
Table 1: Comparison table

Approach	Mean time, ms	Standard deviation
“The biggest face”	47.78	29.96
Most neighbors	339.38	95.9

As we can see from table 1 “the biggest face” approach works significantly faster rather than most neighbors approach other conditions being equal.

Next, we need to find eye points. In order to increase speed we narrow the region of interest. We have computed regions on image that most likely will contain eyes. These regions are determined using masks that are applied to face area. This idea helps to significantly increase detection speed. If eyes were found inside those regions system proceeds to next step. Figure 7 illustrates this process. Position and size of these regions depend on the parameters of the face region that was found earlier.

Here we can also apply “the biggest eye” approach. But our experiments showed that in terms of hit rate it is better to search for eye regions that have most neighbors.



Masked image
A)

Eye-candidates found
B)

Figure 7: Searching for eye-candidates

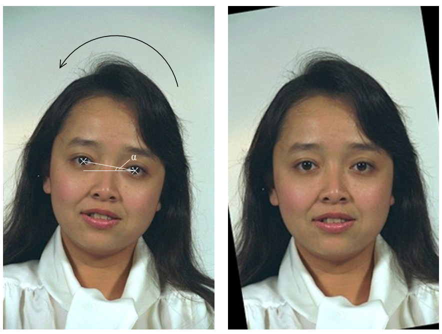
If there were no eyes found in predicted region this can mean following: (i) eyes are closed or incorrectly covered with glasses (Figure 1 A,B), (ii) image has low quality, (iii) image is rotated. The correction of image rotation will be discussed in the next section.

2.2 Image processing

The goal of this step is to adjust image so that it conforms these requirements:

> *To assist in the facial recognition process, the facial image shall be stored either as a full frontal image or as a token image in accordance with the specifications established in ISO/IEC 19794-5. A token image is a facial image in which the image is rotated if necessary to ensure that an imaginary horizontal line drawn between the centers of the eyes is parallel to the top edge of the picture and the size adjusted. ICAO recommends that the centers of the eyes be approximately 90 pixels (Doc 9303, 2006).*

Each image is rotated so that imaginary line between eyes and top edge of the image are parallel. Figure 8 illustrates this process.



Rotated image
A)

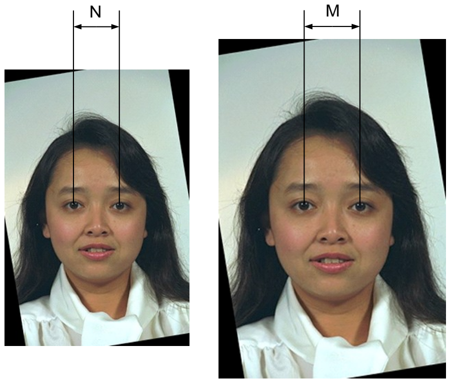
Corrected image
B)

Figure 8: Slope correction

After the slope correction image must be upright. New coordinates of the features that where found previously can be adjusted using following system of equations:

$$\begin{cases} x' = x_0 + (x - x_0) \cdot \cos(\phi) - (y - y_0) \cdot \sin(\phi) \\ y' = y_0 + (x - x_0) \cdot \sin(\phi) + (y - y_0) \cdot \cos(\phi) \end{cases},$$

where (x', y') are the new coordinates of the point, (x_0, y_0) are coordinates of the rotation center, ϕ is rotation angle.



Source image
A)

Resized image
B)

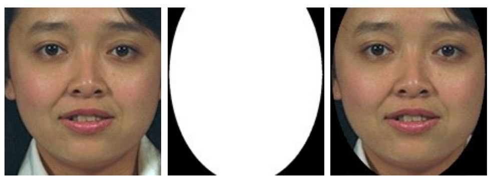
Figure 10: Size correction

Next, distance between eyes is measured, and every image is resized to set this parameter to a certain value. This is done to ensure that all images have the same size in pixels (Figure 10). Let us consider that all images in our database have M pixels between eyes. So, in order to be able to recognize new face we need to correct the size of the new image using the factor M / N, when new face has N pixels between eyes.

Next, each image is cropped to reduce image size. Doc 9303 says the following about cropping:

> *Cropping: Whilst images can be cropped to save storage and show just the eye/nose/mouth features, the ability for a human to easily verify that image as being of the same person who is in front of them, or appearing in the photograph in the data page of the passport, is diminished significantly (Doc 9303, 2006).*

Image A of the figure 11 will be stored in the database in order to help the user of the system identify person manually. Image C is used to identify person automatically.



Cropped image
A)

Mask
B)

Result
C)

Figure 11: Image cropping and masking

At this step we warp image only using eye regions. If we consider to resize image using both x and y axis we change the proportions of the face and that will probably decrease recognition abilities of our system. After this stage we have a set of extracted faces.

2.3 Identification

The goal of this step is to find individuals among known which are the most similar to the input individual. In our work we use eigenfaces approach that was proposed by Turk et al (Turk et al. 1991). The reason is that this approach is able to handle large amounts of data. Face image is considered to be a two-dimensional N by M array. An image can also be treated as a vector of dimension $N \cdot M \times 1$. So each image is considered to be a point in the $N \cdot M$-dimensional subspace. Face images, being similar in overall configuration, will not be randomly distributed in this huge image space. In order to decrease dimensionality principal component analysis (PCA) can be used. This approach consists of next steps: (i) compute the average face vector Ψ; (ii) obtain a set of face vectors Φ centered at expectation Ψ; (iii) compute the covariance matrix and find its K eigenvectors with the biggest eigenvalues; (iv) project each image onto new subspace to obtain coordinates of each image in this subspace.

Average face can be computed as follows:

$$\Psi = \frac{1}{n} \sum_{i=1}^{n} F_i \ ,$$

where F is training set of face vectors, n – number of faces in training set. Then, centered at expectation set of face vector is $\Phi_i = F_i - \Psi$.

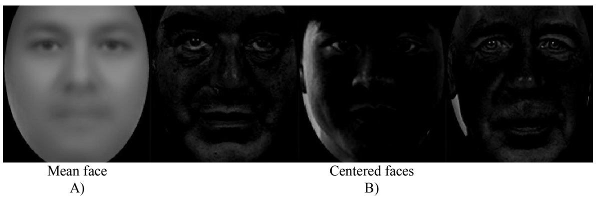


Figure 12: Mean face and examples of centered faces

Average face Ψ (see figure 12, A) is computed using faces of people of different race, gender, age, color, with and without glasses, beards, mustache. Our experiments showed that if we increase n average face remains the almost same. The Euclidean distance between mean faces based on 100 and 200 images is 4.01; 200 and 300 images – 3.3; 300 and 400 images – 0.66; 400 and 500 images – 0.58.

Covariance matrix is determined as follows:

$$C = AA^T \ ,$$

where $A = [\Phi_1 \Phi_2 ... \Phi_n]$ is $N \cdot M \times n$ matrix, and C is $(N \cdot M)^2$ matrix. Matrix C is too large to compute its eigenvectors ν_i. Turk et al. propose to compute ν_i as follows:

$$\nu_i = Au_i \ ,$$

where u_i are eigenvectors of $n \times n$ matrix $A^T A$ (Turk et al. 1991).

Now each face can be represented as a linear combination of first K eigenvectors as follows:

$$\Phi_i = \sum_{j=1}^{K} w_j \nu_j \ ,$$

where $w_j = \nu_j^T \Phi_i$. Figure 13 shows image decomposition to mean face and eigenfaces with biggest weights.

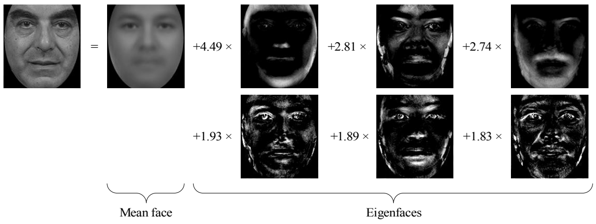


Figure 13: Example of linear combination

In our work we use Euclidean distance to classify an image:

$$\epsilon_k^2 = \|\Omega - \Omega_k\|^2 \ ,$$

where Ω_k is a vector describing kth face class which minimizes ϵ_k^2. A face is classified as belonging to class k when the minimum ϵ_k^2 is below some threshold θ. Otherwise, image is unknown to the system and can be added to the database.

Recognition quality of our system depends on how well eigenface basis is constructed. Let us introduce an idea to estimate how well basis is constructed. Let A be a set of faces which are used to construct basis. Let B be a set of faces which are used to test basis. $|A|, |B|$ are the numbers of images in defined sets respectively. Let $L(\nu)_i$ be a loss vector:

$$L(\nu)_i = (B_i - B'(\nu)_i)^2 \ ,$$

where $B'(\nu)_i$ - face images restored from basis, ν – basis:

$$B'(\nu)_i = (\nu^T W_i) + \Psi \ ,$$

where W_i – representation of ith image in the basis, Ψ – mean face. The target function can be written down as follows:

$$e(\nu)=\sum_{k=1}^{|B|} L(\nu)_k .$$

The task to minimize $e(\nu)$ is practically impossible. However, this target function can be used to compare existing eigenface bases. To illustrate this idea let us build two different bases: ν_{A1} consists of people of all races, gender, with or without mustaches (image set $A1$); ν_{A2} – white male faces without mustache (image set $A2$). Each set of images consists of 100 images.

A) B)

Figure 14: Two mean faces

As expected, mean face computed from first set of images (see figure 14, A) contains features of all races that were included into initial set of images. Mean face computed from second image set is a face of a white male (see figure 14, B). As a test set we used a set of 45 images that consists of males and females of different races. Images from this set are not present in set $A1$ and $A2$. The result of comparison of two bases using proposed measure is shown in the table 2.

Table 2: Comparison table

$e(\nu_{A1})$	$e(\nu_{A2})$
54.33	74.5

Since $e(\nu_{A1})<e(\nu_{A2})$ we can expect basis ν_{A1} to restore faces more precisely than basis ν_{A2}.

3 Conclusions and future researches

We have created the instrumental system for face detection and recognition tasks with estimates for speed and accuracy. Our computer face recognition system is able to handle large amounts of data due to proposed "the biggest face" approach and regions that most likely contain eyes.

The proposed measure to estimate how well basis is constructed can be used in challenging task of selection the eigenface basis for recognition purposes.

The proposed instrumental system was implemented in C# language in Microsoft Visual Studio 2008 using Intel Open Source Computer Vision Library (http://sourceforge.net/projects/opencv/), Emgu.CV library (http://sourceforge.net/projects/emgucv/) and Matlab R2009b. Examples are provided using The Color FERET Database (http://face.nist.gov/colorferet/).

One of the future research directions is to apply Bayesian face similarity (Moghaddam et al. 2002). In order to achieve higher system training speed parallelization of this process is worth researching.

Another promising research direction is to divide people into some groups using such features as gender, race, age. Each new image then firstly will be compared to individuals in groups it might belong to.

Reference

Freund, Y., R. Shapire (1996). "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting." Journal of Computer and System Sciensies. pp. 119-139.

International Civil Aviation Organization (2006). *Machine Readable Travel Documents. Doc 9303*

Lienhart, R., J. Maydt (2002). "An Extended Set of Haar-Like Features for Rapid Object Detection." Proceedings of The IEEE International Conference on Image Processing. pp. 900-903.

Moghaddam, B., T. Jebara, A. Pentland (2002). "Bayesian Face Recognition." Appears in: Pattern Recognition. pp. 1771-1782.

Osuna, E., R. Freund, F. Girosi (1997). "Support Vector Machines: Training and Applications." Massachusetts Institute of Technology.

Romdhani, S., P. Torr, B. Scholkopf, A. Blake (2001). "Computationally Efficient Face Detection." Proceeding of the 8th International Conference on Computer Vision.

Rowley, H., S. Baluja, T. Kanade (1998). "Neural network-based face detection." IEEE Transactions on Pattern Analysis and Machine Intelligence. pp 22-38.

Schneiderman, H., T. Kanade (2000). "A Statistical Approach to 3D Object Detection Applied to Faces and Cars." IEEE Conference in Computer Vision and Pattern Recognition.

Turk, M., A. Pentland (1991). "Eigenfaces for Recognition." Journal of Cognitive Neuroscience. pp. 71-86.

Viola, P., M. Jones (2004). "Robust Real-Time Face Detection." International Journal of Computer Vision. pp. 137-154.

Yang, M. (2004). "Recent Advances in Face Detection." IEEE ICPR 2004 Tutorial.