Università
Ca'Foscari
Venezia

Department of Environmental Sciences, Informatics and Statistics

Master Degree in Computer Science

Final Thesis

Face recognition on embedded devices

A proof of concept for residential application

Supervisor
Ch. Prof. Marcello Pelillo

Co-supervisor
Ing. Antonio Milici

Graduand
Marco Petricca
795687

Academic Year
2018 / 2019

Abstract

Face recognition is a well-known technique with a wide range of existing real world applications. Residential systems, like video intercom or security alarm, are instead nearly unfamiliar terrain to such methods with few commercial solutions available today in the market. A leading company in the sector has called for a research, with the purpose of assessing the feasibility of equipping their embedded video intercom systems with a feature for automatic identification and authorization of the calling subject. In this paper a combination of face detection and recognition methods on such system has been studied and evaluated, leading to a proof of concept in order to verify the feasibility assumptions, and support the claimant company decision process on investing for prototype development and final product extensions and adjustments. Early promising results suggest that the proposed system could prove usable, complying with constraints such as providing a reasonable recognition rate and execution time, running on embedded hardware and being user-friendly.

This work is dedicated to those who supported me, especially my mother.

Thank you.

Contents

1. Introduction

AI technologies are becoming increasingly present in everyday life, reaching a level of maturity and diffusion such as to leave their traditional academic applications and top technology companies boundaries. Thanks to embedded hardware evolution during the past years, it is possible to equip common devices with a range of whole new features, including AI based ones, through software development or dedicated hardware module add-ons (eg. microcontroller units available on the market). It is the case of a leading company in the industrial automation sector which has called for a research, with the aim of assessing the feasibility of equipping their embedded video intercom systems with a feature for real time automatic identification and authorization of the calling subjects, namely a face recognition functionality.

Face recognition technologies have been used until recently most prominently by government and law enforcement agencies for security reasons, with a growing commercial interest and investment by private companies as the technology has become more accurate and less costly. The result is a continuously growing range of use cases and consequent developments for consumers and businesses applications. Some examples are photograph identification in social networking, automatic organization of pictures collection, safety and security in private areas, secure physical access control to buildings or personal devices, customized and improved products, services, advertisements and other marketing purposes. While face recognition provides new potentials for safety, security and business development, it also introduces complicated issues about the nature of consumer privacy and surveillance, with companies accumulating increasingly massive amount of consumer data and a legislation not yet fully developed or worldwide standardized to address the unique risks arising from the use of these technologies. Some of those include unintended privacy or data breaches due to obsolete or vulnerable security system designs, data monetization through sales to third party, and the irreplaceable nature of the biometric data ([25]).

Generally, a face recognition system is made of a series of building blocks in charge of performing various tasks with the final goal of achieving a successful recognition of people from digital images or video frames. Those can be summarized in a *capture device* to acquire the subject images in a digital format, a *face description* algorithm to create suitable representations of faces in a computer system, an image *dataset* to store the captured faces and their description, and a *comparison* method to evaluate the subject similarities.

The recognition task begins with an image being fed as input to the system, upon which a *face detection* module searches for any face contained in it and, if found, sent to the *face recognition* module. Here facial features (eg. nose, eyes, etc.) are extracted from the detected face and compared to previous knowledge by using a suitable similarity measure. Therefore, the system verifies whether the subject belongs to one of the already known classes, or if it is unknown, and acts accordingly to the recognition results and its configuration, for example sending an authorization message to an access control module or to another application.

To maximize the performances it is crucial for such system to exploit correctly the sources of variations between faces, which can originate from the subject appearance such as eyes location, nose shape or presence of glasses, or related to external sources like illumination, pose, scale or external objects occlusion. Those external factors might have a negative impact on the detection and recognition methods performances, thus a range of preprocessing methods and operations are applied to the modules input to address this, resulting in boosted execution speed and recognition scores.

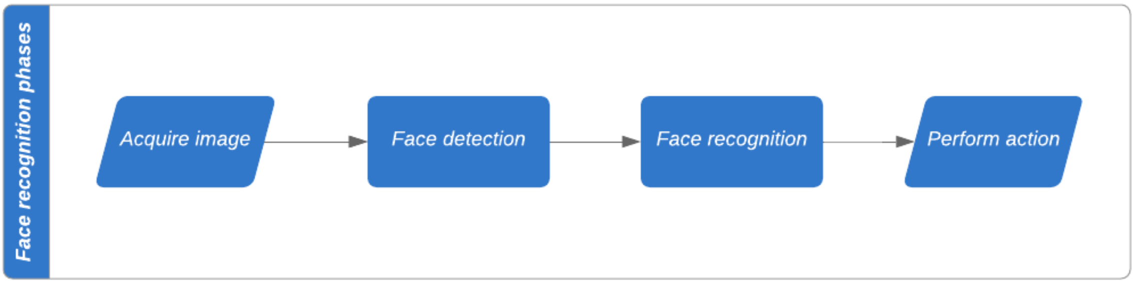

Figure 1.1 Phases of a face recognition system.

This paper treats the design, development and evaluation of a proof of concept for a face recognition system intended for residential applications, under variable but controlled light conditions. The initial concept comes from a corporate context, inheriting the characteristics objectives and schedules concreteness, whose success and consequent realization of the system depends also from a variety of non-technical factors such as:

- The feasibility of a solution respecting the imposed constraints, in the form of a working proof of concept
- The range of appliable product applications, in particular the possibility of retrofitting existing devices and installations, and draw inspiration for the design of more innovative and competitive products
- The solution sustainability in the company business ecosystem as a direct consequence of its profitability
- The observance of company deadlines for budgets and planning
- Provide an effective delivery of the solution business proposal and business cases to the steering committee, as a support for decisions on the future development.

The Research & Development department of the claimant company released a project draft, containing a series of guidelines to motivate the research and its consequent market opportunity, whose purpose is to provide current and future owners of an entry level existing video intercom system, possibly with analog capture devices, with face recognition capabilities for automatic access control purposes. The face recognition would be an optional add-on feature to install on the internal video receivers, which can be purchased separately by each tenant.

The main advantages highlighted in the project draft are summarized below,

- Infrastructure costs are kept low
- Flexibility in multi-dwelling configuration, as only tenants who wish to enable the add-on function have to purchase a high-end internal video receiver, while others can keep the entry level internal video receivers
- Data of the subjects is stored in the private internal video receiver, and not in a shared or cloud based infrastructure, allowing to overcome most of the privacy and remote connection related issues
- The tenants can access their property hands free, meaning that no additional personal identification support containing access credentials (eg. RFID smartcard) is required.

The project draft also mention the challenges concerning the face recognition solution techniques,

- The computational effort of the face recognition methods must be lightweight, to allow real time verification of subjects on a low power embedded device
- The image transport chain originating from an external entry panel may deteriorate the image quality fed to the face recognition algorithms, hence the chosen methods have to compensate for the potential noise or low image quality
- The subject face illumination may experience variations depending from daytime and installations specificity, with the face recognition algorithm being robust as much as possible to such changes.

To answer all the above mentioned topics, a typical video intercom system has been put together, to reproduce as close as possible the real application environment. With the data produced by the video intercom system, an image dataset of fourteen subjects and 780 faces has been created, and used to study and evaluate the performances of a combination of face detection and recognition methods. The result is a pipeline composed by Viola-Jones face detector and Eigenfaces face recognition, which is able to achieve good performances and comply with the company project constraints.

This paper describes in Section 2 the problem of faces recognition on an embedded video intercom system and its constraints, the related work on embedded systems in Section 3, and the explanation of the methods composing the pipeline in Section 4. The experimental system setup with the details on the generated dataset and its use as training and test set is described in Section 5, with the results achieved by the prototype system presented in Section 6. Conclusions and possible future improvements are proposed in Section 7.

2. Project description

Scope of this project is to combine Artificial Intelligence techniques belonging to the *Computer Vision* area with a video intercom system, to assess if it is possible to detect and recognise a narrow set of known faces captured from its video stream; if a subject is detected and identified successfully the intercom system performs some defined actions, typically opening the lock connected to a door or gate. Since intercoms are widely spread devices, familiar and accepted by the majority of the population, users are well aware of the presence and operating sequence of such system and are willing to be identified by clearly showing their faces at close range, thus placing this project in a collaborative environment.

The outcome of the project will allow to determine if it is possible to equip both new and selected existing products with an additional facial recognition feature, mainly aimed for residential applications. This last detail in fact limits the amount of subjects allowed to access, to generally less than five people.

When a subject presents itself in front of the intercom entry panel camera, it interacts with the device by pushing the call button of interest: this explicitly begins the identification procedure. The external device camera captures the subject image, which is encoded and transmitted to the internal video receiver, typically installed in an apartment or other private area. The latter device is capable of decode the video stream, extract a series of images, detect faces (if any) and, based on previously known subjects, assess the current user identity and allow or restrict its passage into the private area.

It is necessary to consider that during different recognition sessions the users will be very likely subjected to variable conditions of illumination, different background, distance from camera or facial expression, which could pose a challenge for facial recognition methods.

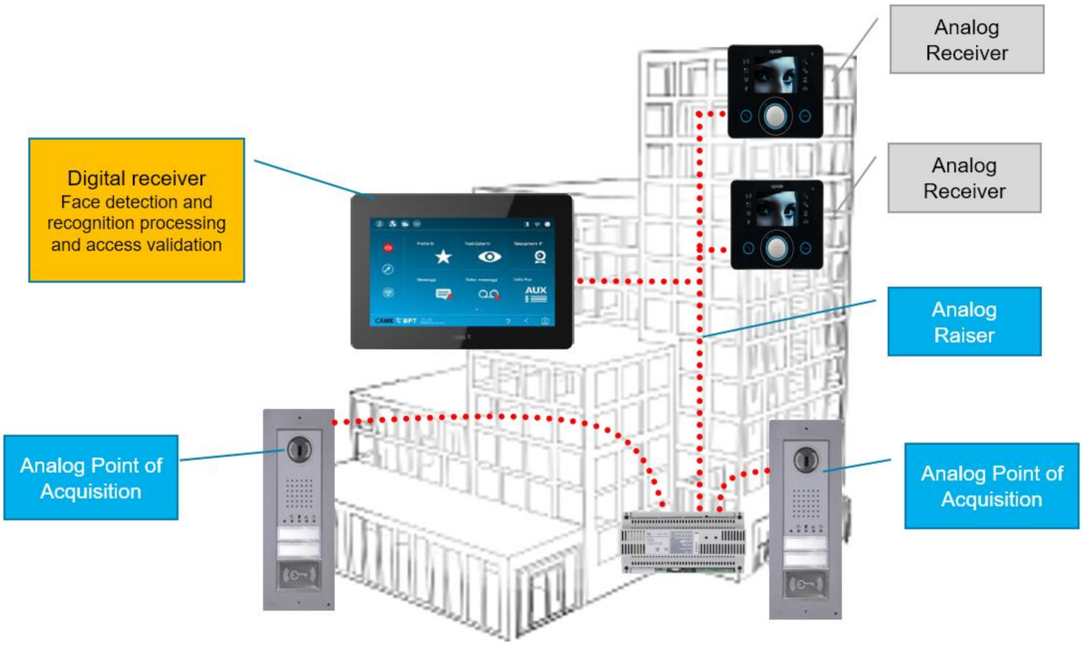
Analog Receiver
Digital receiver
Face detection and recognition processing and access validation
Analog Receiver
Analog Raiser
Analog Point of Acquisition
Analog Point of Acquisition

Figure 2.1 Example of application in a residential building.

2.1 Constraints

Even though the abovementioned collaborative application environment is favourable, there exists a number of general constraints which have to be taken into account for defining an optimal solution, several of these derived from the claimant company guidelines and others coming from the nature of the project itself.

The resulting system must be robust as much as possible to variable conditions of illumination, different background, distance from camera and facial expressions, as well as easy to use for the end customer through a user friendly interface, setup and use procedures. It also has to deliver sustainable performances in terms of speed and accuracy, and operate successfully on low power local embedded hardware. The *local* requirement had been chosen to minimize privacy issues and address any external connection (eg. Internet) issue which may arise.

Furthermore, the speed requirement relates not only to performances but also to the deadline for implementing a proof of concept for this system as well, as the applicant company has to make a decision for including a fully featured development project in its yearly budget, for equipping both new and selected existing products with a facial recognition feature.

3. Related work

Face detection and recognition topics have been extensively discussed in the literature, resulting in a vast amount of work mainly focused on improving the accuracy through new methods with growing computational complexity. Even though the recent years performance increase of embedded systems, the development of computationally expensive methods on such hardware has proven challenging, with many solutions resorting to remote processing of the recognition steps, instead of relying on local hardware capabilities.

Research and early applications of face detection and recognition methods on embedded systems are traceable back to early 2000s, and active as today.

For general purpose applications, an ARMv5TE PDA based embedded system has been built by Chu et al. ([1]), where a face annotation application used Viola-Jones face detection and ARENA face recognition. Zhao et al. ([2]) worked on a similar system using Viola-Jones and Eigenfaces with Euclidean distance methods, and running on Tiny6410 platform for portable and mobile applications. Broker et al. ([3]) described a hybrid face recognition algorithm by combining PCA and LDA methods on a Raspberry Pi 3 board. Although those systems are architecturally similar to the one proposed in this paper, none of them treats the video intercom use case specificity, as illumination for the subject is assumed to be invariable, the hardware platform used are general purpose or obsolete, and the content of those publications is more focused on describing the system development details on the hardware platform of choice.

There have been also more specific domain embedded application studies, Zuo et al. ([4]) propose a 95% accuracy LDA based near real-time face recognition system for consumer applications (eg. For Smart TV integration) where both processing efficiency and robustness/accuracy are required. Kang et al. ([5]) made a PCA-based face recognition on an ARM CPU embedded module for robot application. Günlü ([6]) proposed a 97.4% accuracy LDA based system for Digital Signal Processors (DSP) on smart cameras, with a Viola-Jones code optimization for this specific platform. Shan et al. ([7]) developed a prototype of face recognition module on a mobile camera phone, for identifying the person holding the device and unlocking the keyboard. The authors used Viola-Jones face detection and a pose variability compensation technique, which synthesizes realistic frontal face images from nonfrontal views, coupled with adaptive principal component analysis (APCA) and rotated adaptive principal component analysis (RAPCA), both insensitive to illumination and expression variations and not computationally intensive; resulting samples are used then for training with SVN or NN methods.

Publications on embedded system development and integration have been published by Sahani et al. ([8]) and Manjunatha et al. ([9]) for home security solutions, both using the Eigenfaces method on ARM11 and Raspberry Pi platforms, while Android solutions for smartphones and tablet computers have been developed by Gao ([10]) with a combination of Viola-Jones and Eigenfaces methods, and Alam et al. ([11]) with Local Binary Pattern Histogram method.

Face detection and recognition has also been studied for optimization of existing methods purposes; Bigdeli et al. ([12]) worked on improving the Viola-Jones method instructions on a FPGA embedded face detection solution, with the aim of addressing the real-time image processing requirements in a wide range of applications, and achieving up to 110 times speedup using custom instructions. Pun et al. ([13]) and Aaraj et al. ([14]) treated as well code optimization for faster execution, the first proposing a semi-automatic scheme for face detection and eye location, and the latter exploiting parallel processing on two different methods for recognition and authentication, with a combination of PCA-LDA and Bayesian approach, yielding a speedup factor of 3 to 5 times. Ramani et al. ([15]) presented the design of a domain specific architecture (ArcFace) specialized for face recognition, describing an architectural design methodology which generates a specialized core with high performance and very low power characteristics, based on Viola-Jones and PCA methods. Theocharides et al. ([16]) and Al-shebani ([17]) also worked on the design of special-purpose hardware, the first for performing rotation invariant face detection, developing a 7 Watts 75% accuracy device, and the latter for FPGA applications using KNN method with the city-block metric.

Summarizing, traditional methods such as Linear Discriminant Analysis (LDA), Eigenfaces and Viola-Jones methods are still a popular choice for embedded applications mainly due to their limited processing power requirements. Also in ([18]), Milosevic et al. studied face emotion recognition on embedded hardware (Raspberry Pi, Intel Joule, Movidius neural computing stick) with Deep Learning approaches, showing that low power consumption and short inference time requirements, are limiting factors for real-time usage of more complex methods on embedded systems.

4. Evaluated solution

Given the project and the imposed constraints, the feasibility of its solution has been evaluated through a supervised learning approach. The resulting system has to be able to distinguish between a narrow group of authorized subjects and everything else, also including generic objects, by detecting and comparing the found faces. To achieve this, it is necessary to first instruct or *train* the system to recognise the allowed subjects, by presenting a small set of representative pictures of their faces. The representativeness of the abovementioned set is defined mainly through variable illumination conditions, as the system will be used during day and night hours, and secondarily by facial expressions, as it is reasonable to suppose that users may present themselves with a non-neutral pose in front of the camera, eg. smiling or with closed eyes. Since the subjects are willing to be identified by clearly showing their faces when standing in front of the camera, it is also reasonable to assume that head and face position, their orientation and rotation are invariant. After performing the training phase, the system is able to perform recognition or *validate* users, by comparing the subject standing in front of the camera with its acquired knowledge.

To maximize the difference (or similarity) between subjects due to *facial features* and not collateral unwanted sources (eg. background), a preliminary face detection step is performed on the stream during both training and validation phases, to extract (when available) only the subject face and exclude other unnecessary areas of the image.

Considering that the training phase is expected to run for a fraction of times compared to the validation phase executions, potentially only once, a sensibly slower training phase is considered acceptable; on the contrary, validation has to be fast and lightweight.

The Viola-Jones framework ([19]) and the Eigenfaces algorithm ([23]) have been selected respectively as face detection and face recognition methods. Although they are somewhat aged, they fit well the requirements and the constraints of the project, providing a convenient approach to build a proof of concept to assess the solution feasibility and its approximate performances. Once the training phase is performed to create the main data structure known as *face subspace,* the Eigenfaces execution time to evaluate face similarity through projection onto the transformation matrix is reasonably fast. Comparable advantages are achieved by Viola-Jones face detection step.

Similar applications on embedded hardware confirm that, despite the age, both methods are still quite popular in such environment, see ([1]), ([2]) and ([3]) for insights.

Given that the final outcome of the project is provide a proof of concept for the feasibility of such a system, the selected methods are meant as a convenient starting point, considering also the deadline requirement and their understandability for a non-specialized or non-technical steering committee. Moreover, if traditional techniques proves successful, a more sophisticated approach could only improve the results.

While of great appeal nowadays, a Deep Learning approach had to be discarded due to the high amount of images needed to provide a sufficiently rich training set, and because the final user is very likely to not accept a complex or long setup routine and provide more than 5 or 6 images per subject. Such techniques are also demanding in terms of hardware requirements, and embedded systems are still usually inadequate in this respect as shown in ([18]).

4.1 Viola-Jones framework for Face Detection

Real world face images often includes unnecessary information for face recognition task, whose accuracy might be marred by. In order to minimize such collateral effects, it is a common practice to pre-process images in multiple ways before being fed to recognition algorithm. One of those steps is face detection, which locates the subject face (if any) and crop the image to the detected face boundaries, removing superfluous sections like the background. It also concur to render face recognition a much quicker step, as without knowing the position of the face in the image, it would be necessary to apply a face recognition algorithm at every pixel and scale, and such a process would be too slow.

Existing face detection techniques can be grouped in three families: *feature based*, which searches for the locations of distinctive image features such as the eyes, nose, and mouth, and then verify if they are arranged in a plausible schema; *template based*, looking for small parts of an image which match a template and can work with a range of pose and expression variability; or *appearance based* methods which scan the image using small overlapping rectangular patches in search for possible faces, further analysed by cascades of detectors.

Considering the application target, the Viola-Jones method, a supervised appearance based frontal face detector, has been chosen for the detection task due its overall accuracy, detection speed and low power demands. It is a widely used method for real-time object detection and, as Eigenfaces, the majority of the computation time is required during the training phase, while the detection phase is very fast: Paul Viola and Michael Jones could indeed implement their real-time face detector on a low power early 2000 Compaq iPAQ embedded handheld device.

Viola-Jones is a visual object detection framework composed by three main pillars: an image representation method for fast feature calculation known as *Integral Image*, an *AdaBoost* based learning algorithm for representative features selection, and a method for *combining learned classifiers* to reduce unnecessary computation on background areas of the image, and focus on more promising regions. The initial hypothesis the authors managed to demonstrate through the achieved results, was that a small amount of representative features can be combined to produce an efficient classifier for detecting faces in an image. The framework is able to achieve very good detection rates at acceptable frame rate, working only with information present in grayscale images and not relying on other auxiliary information. It was the first algorithm to introduce the concept of boosting to computer vision applications.

The building blocks for classification are three types of scale invariant rectangular-like simple features, whose values are assessed by calculating the difference between the sum of the pixel contained in the white areas and the pixel in the black areas. The main reasons for using features in place of pixels are high computational speed, and the fact that features are able to encode specific domain knowledge which is not easy to learn from a limited amount of training samples.

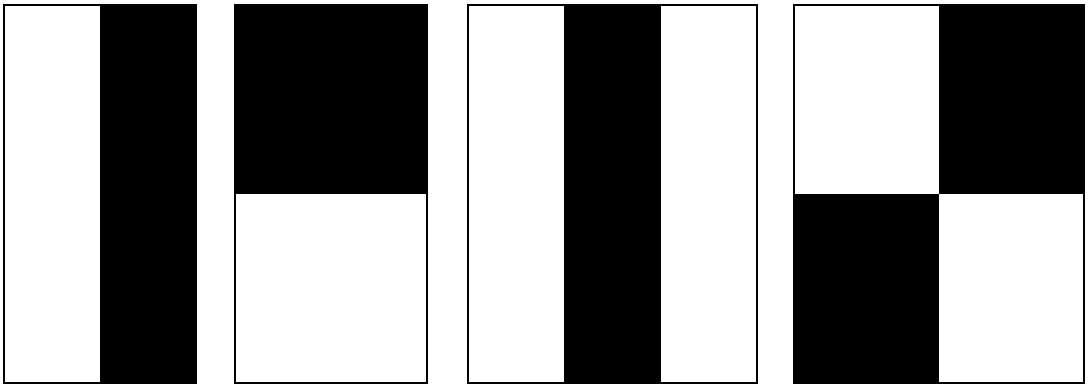

Figure 4.1.1 Features used in Viola-Jones: two-rectangle (horizontal and vertical), three-rectangle and four-rectangle.

Input images are evaluated in sub-windows of 24 x 24 pixels, and since the number of possible rectangle feature for an image is huge, it is mandatory for effective classification features to be discoverable and computable with few instructions. For the latter requirement the Integral image representation has been introduced by the authors, which can be calculated in one pass over the original image, allowing rectangular features values to be determined by using from six to nine array operations, depending on the feature structure. This balances the features simpleness with high computational efficiency.

The value of the integral image representation *ii* at location *x,y* is the sum of the pixels of the quadrangle defined from the top left point of the image *i* to the given *x,y* pixel coordinate,

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

which fast calculation is due from the recurrences below,

$$s(x,y) = s(x,y-1) + i(x,y)$$

$$ii(x,y) = ii(x-1,y) + s(x,y)$$

where *s(x,y)* is the cumulative row sum.

The other requirement, namely the ease of detection for effective classification features, is satisfied through a variant of the AdaBoost learning algorithm, for both effective features selection and classifier training, accomplished by combining a sequence of simple classification functions or *weak learners* to produce more powerful ones. The main modification, with respect to the original AdaBoost, consists on forcing the weak learners to depend on a single feature, thus each new classifier selected by the boosting process can be considered as a feature selection process. Weak learners can be seen as *decision stumps* which is the most simple instance of a *decision tree*, or as separating *hyperplanes*.

During any iteration or *hypothesis* of the learning procedure, a weak learner $h_j(x)$ consisting of a feature f_j, a threshold θ_j and a parity p_j, is trained for selecting a single rectangle feature which yields the optimal threshold classification function, or equivalently better separates labelled examples classes. No single weak learner can achieve high performances on its own, in fact their error rate is usually just better than a random selection, as it is limited to a single feature. After the feature has been learned, the samples weights are readjusted as a function of whether they were correctly classified during the previous stage. This is done to highlight incorrectly classified points, with an additional learner being trained on those updated examples. The procedure is repeated for a defined number of rounds, obtaining in the end a final weighted combination of weak classifiers with its global threshold. It has been proven by Freund and Schapire that the training error of the combined classifier approaches zero exponentially in the number of rounds ([20]). Interestingly enough AdaBoost is not suffering from *overfitting*, a typical behaviour of learning methods whose model complexity is excessive for the given data, and tends to adapt to noise rather than generalize correctly. As the complexity of the combined classifier increases with the number of rounds, the test error is found to decrease even after the training error reaches its minimum.

The classification rule for a weak learner is,

$$h_j(x) = \begin{cases} 1 & if\ p_j f_j(x) < p_j \theta_j \\ 0 & otherwise \end{cases}$$

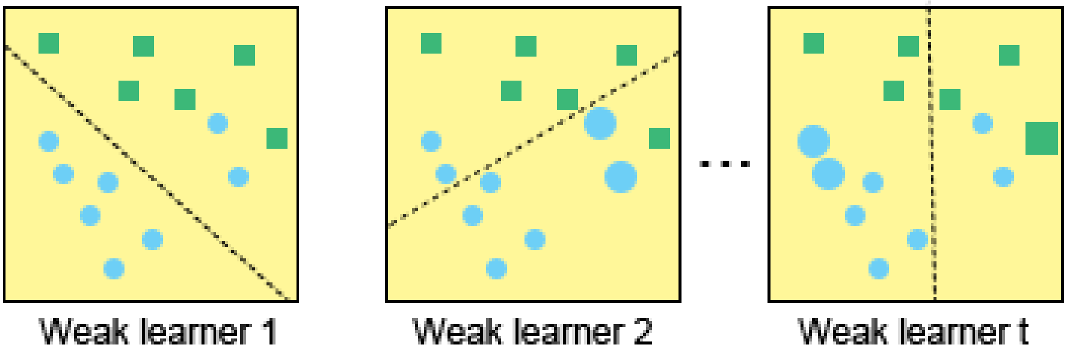

Figure 4.1.2 Weak learners iterations

The final classifier is a weighted linear combination of the weak learners, with weights inversely proportional to training errors, and yields the form,

$$H(x) = \begin{cases} 1\ if \displaystyle\sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 \qquad\qquad otherwise \end{cases}$$

where t is the iteration or hypothesis number and α is a coefficient expressing the contribution of the weak learner at iteration t in the final classifier; the value of α depends on the t classifier error rate (or goodness).

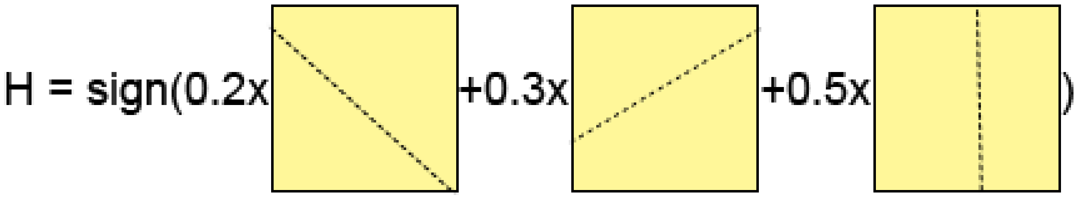

Figure 4.1.3 Final classifier composition example

It is worth to note that a detector consisting of a single strong classifier with many features would be inefficient since the evaluation time is constant, no matter the input, and its training phase could take weeks, mainly due the large amount of feature (difference of rectangles) hypotheses to examine at each stage. To increase the performances, the last section of Viola-Jones framework is a method for combining in a cascade fashion a series of growing complexity strong classifiers, with the aim of both increasing the detection rate and reducing the amount of evaluated input images sections, thus the training time, as progressing through the stages which the cascade is made of. This is accomplished by deploying simpler, thus efficient, strong classifiers in the early stages and more complex and slower strong classifiers in the following stages, with only positive detection image sections proceeding to the next classifier. A section of the input image will be marked as face only by getting through all the cascade stages.

The stages purpose and complexity is variable, and depends on their position in the cascade: instead of focus in finding faces, the early stages task is more related to reject non-faces sections of input images while preserving all promising areas, even if at the end those will result being false positives. False positives will be managed by more complex subsequent stages, which are trained with increasing difficulty examples coming from previous classifiers outputs. This architecture is made possible by lowering the initial stages simpler classifiers threshold, through an additional modification to AdaBoost which had to be introduced in Viola-Jones to accommodate such customizable thresholds: in its original version, the boosting algorithm searches for the lowest achievable error rate, while the high detection required by Viola-Jones simpler classifiers, implies a suboptimal false positive rate.

The cascade training process involves also finding an acceptable compromise between classifiers complexity and overall performances, where the number of classifier *stages*, the number of *features* per stage and the *threshold* of each stage are traded off, with the aim of minimizing the expected number of features to evaluate *N*, given the target for false positive rate *F* and detection rate *D*.

The two latter values are defined by the following,

$$F = \prod_{i=1}^{K} f_i$$

$$D = \prod_{i=1}^{K} d_i$$

where *K* is the number of classifiers in the cascade, f_i and d_i are the false positive rate and the detection rate of the ith classifier. The expected number of features to evaluate *N* can be calculated by,

$$N = n_0 + \sum_{i=1}^{K} \left(n_i \prod\nolimits_{j<i} p_j \right)$$

where p_j and n_i are respectively the positive rate and the number of features of the ith classifier.

Although finding the optimal trade-off of the abovementioned values is a difficult problem, to produce an effective and efficient classifier it is sufficient to select the minimum acceptable values for the false positive rate f_i and the detection rate d_i. During the training of the cascade layers, AdaBoost will take care of increasing the number of features until the given rates are achieved, or the number of layers of the cascade if the overall false positive rate F is not met.

The final outcome of the whole training process is an over 30 stages pipeline, where the first classifier is designed to have as low as two rectangle features, detecting almost all faces and rejecting approximately 60% of the non-faces sections, while the final stages are composed of 200 features classifiers.

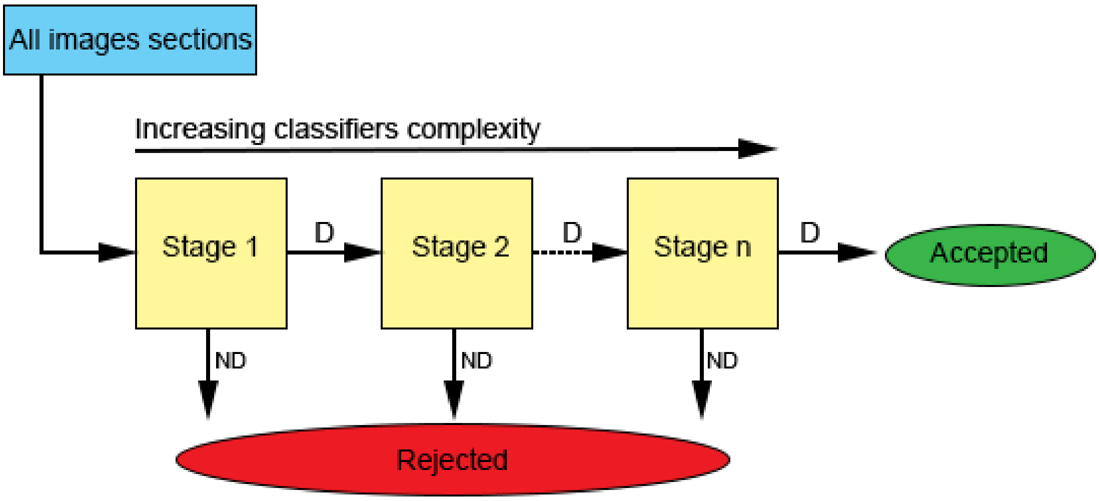

Figure 4.1.4 Classifiers cascade schema, D=detected, ND=not detected

The Viola-Jones method suffers of some known limitations, like its reduced effectiveness in detecting tilted or turned faces, its sensitivity to lightning conditions and multiple detections of the same face due to subwindows overlap. For lighting sensitivity issue, images can be pre-processed before training and validation phases by applying variance normalization, in order to minimize the impact of different lightning conditions. The computational effort necessary for this step is limited, as any sub window variance can be calculated using a pair of integral images, and during scanning the normalization can be achieved by post multiplying the feature values, avoiding to work directly on the image pixels.

The multiple detection issue can be easily addressed, as two detections belongs to the same group if their bounding regions overlap. Post processing is applied to the detected sub windows, to combine overlapping detections and yielding a single final detection. This may also contribute to the reduction of the amount of false positives.

4.2 Eigenfaces method for Face Recognition

In the recent years, the advancements of identification and security systems based on biometrics has drawn much attention, with a consequent spreading of their adoption, as they allow for convenient unique identification of individuals. Subjects are identified exploiting human body and personal behavioural characteristics, contributing to reduce the risk of unauthorized identity usage. Face recognition is one of those biometric identification technique, with a distinctive trait of not requiring the cooperation of the analysed subjects to work.

Faces are probably the most common used characteristic for identification everyday, on which human beings developed very robust recognition abilities, succeeding to memorize and recognize thousands of faces despite sensible changes (eg. aging) and visual alterations (eg. clothes, glasses, head rotation, etc.). The development of a face recognition system could prove challenging and not just for technical reasons; the natural variation in appearance and expression in faces, or the occlusion, variability of scaling, rotation, size, lighting or point of view of real world unconstrained or non-collaborative environment, or even the privacy related issues of adopting such systems in public areas are all aspects increasing the task complexity. In general, face recognition methods gives best results when they work with full frontal images of faces and relatively uniform illumination conditions.

Early applications of face recognition techniques dates back to 1966([21]), but it is not until 1991 that the first instance of automatic face recognition method “Eigenfaces for Recognition” is published by Turk and Pentland, proving the feasibility of such systems. The authors exploited the representation of faces introduced by Sirovich and Kirby([22]) to develop a supervised learning method that can locate and track faces, and recognise the subject by comparing its facial characteristics to those of already known individuals.

Images in computer systems are composed by N pixels arranged in a two dimensional matrix, and can be thought as points in an *image space*, having the same dimensions as the number of pixels. Even for simple images this space extension is massive, making the evaluation of similarities a computationally expensive and difficult task, mainly due to the curse of dimensionality issue.

Support comes from the fact that face images does not randomly distribute in the image space, occupying instead a limited subarea of the whole image space spanning the so-called *face subspace,* a lower dimensional linear subspace in the image space, created in such a way to best encode variation among face images, and preserve most of the information.

The face subspace is obtained using *Principal Component Analysis* (PCA) a dimensionality reduction method, which approximates vectors in the original space by finding a basis φ in an appropriate lower dimensional space, while keeping as much information as possible. Equivalently, assuming that the columns of the matrix ϕ_{NxM} containing *M* images are the data points where each of the *M* images is of size *N*, the aim is to obtain an orthonormal basis of size *M'* with $M' \leq M$, that produces the smallest sum of squared reconstruction errors for all the columns of $\phi - \overline{\phi}$, where the last term refers to the average column of the image matrix. The basis φ to accomplish this is obtained through PCA, and is composed by the *M'* eigenvectors of $(\phi - \overline{\phi})(\phi - \overline{\phi})^T$ that correspond to the *M'* largest eigenvalues of the images covariance matrix.

PCA calculates a best coordinate system for image compression projecting the data along the directions where the data varies the most, with the *M'* eigenvectors as output. The directions are determined by the abovementioned eigenvectors of the face images covariance matrix, which are pointing in the direction of maximum variance of the data. The resulting components are sorted corresponding to the largest eigenvalues, whose magnitude represents the amount of variance of the data along the eigenvector directions.

The first principal component given by PCA is the linear combination of the original dimensions with the maximum variance, the second and following principal components are as well a linear combination of original dimensions, which are orthogonal to the previous principal components. For defining the number of components, it is possible to choose directly the value, or define the amount of variance of the original data to keep, with the result of discarding the *n* smaller eigenvectors.

Principal Component Analysis most relevant properties can be summarized as *approximate reconstruction* of the initial distribution x, *orthonormality* of the basis found and *decorrelated* principal components.

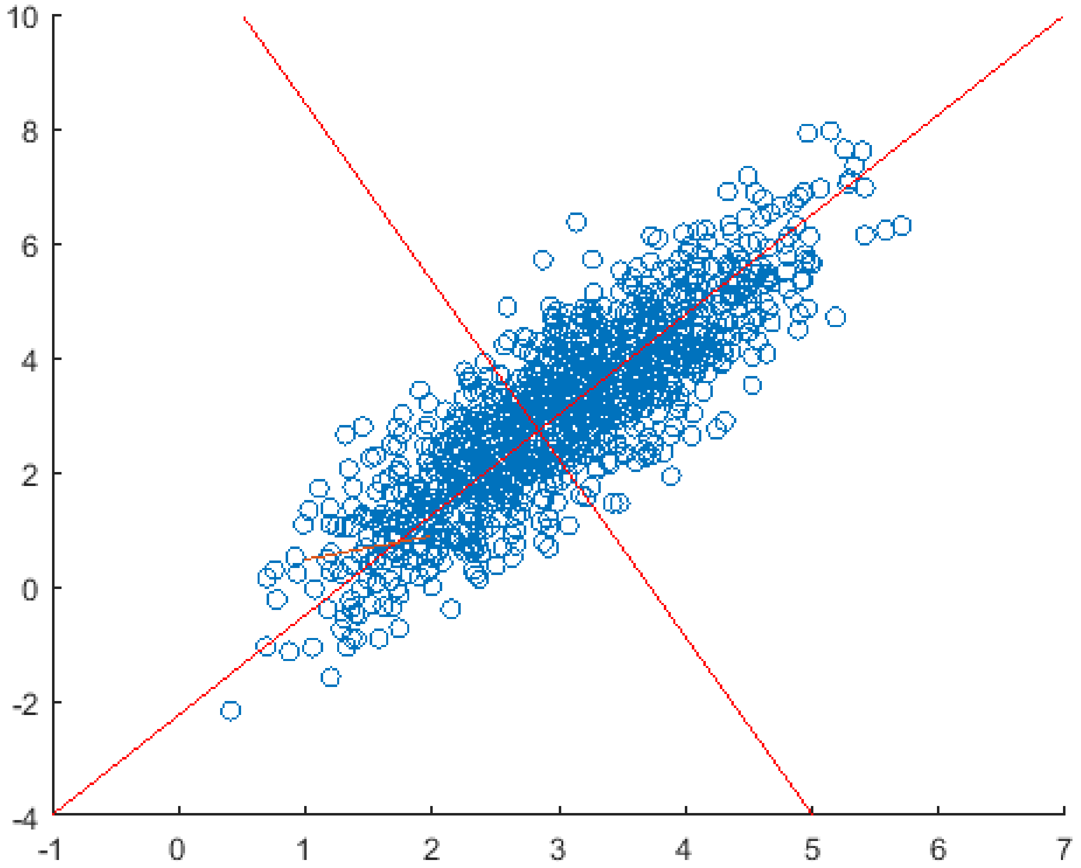

Figure 4.2.1 A multivariate Gaussian distribution and its two first principal components directions

$$x \approx \varphi_k y$$

$$\varphi_k{}^T \varphi_k = I$$

$$E\{y_i y_j\}_{i \neq j} = 0$$

Within this face recognition framework, principal components are called *eigenfaces*, as they can be visualized as an image which resembles a human face. In fact the eigenfaces encode the variations of the face images set, from the most to the least relevant, depending on the associated eigenvalue, emphasizing both the significant local and global features of a face.

Any image of the dataset can be represented by a linear combination of the calculated eigenfaces, or approximated by using a small set of those with higher eigenvalues, through a projection of the image onto the face subspace. This yields a *weight vector* for each projected image, which acts as a compact representation and can be used to compare images and discern similarities based on the distance between those weight vectors. As shown in the image below, the subject face characteristics can be easily recognised by humans through a reconstruction in a subspace spanned by the largest ten principal components (or eigenfaces) of a training set composed of three subjects and a total of 111 images.

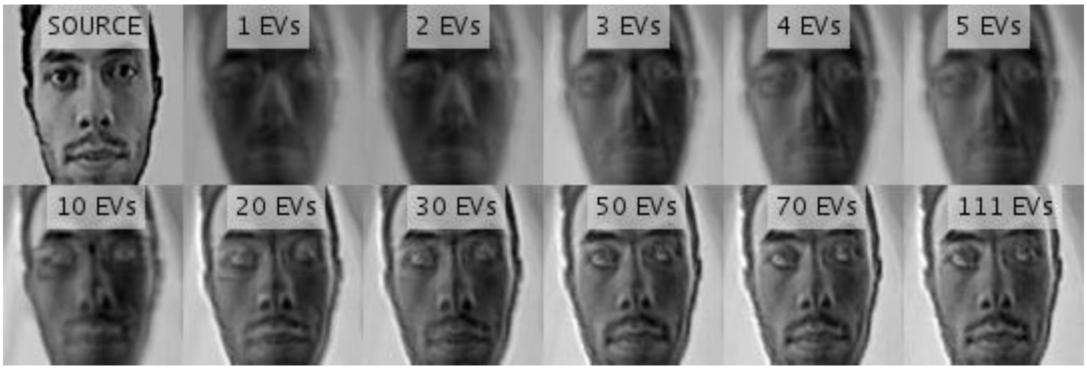

Figure 4.2.2 A face image (top left) and its approximated reconstructions

Some preliminary steps are required in order to calculate a proper face space for the given set Γ of M face images. The first one is to remove color information from the images, as it is redundant information and serves no purpose for the recognition task. Then the face distribution is centered, as PCA assumes data to be distributed as a Standard Gaussian with zero mean. The mean face Ψ is calculated averaging all M faces as following,

$$\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n$$

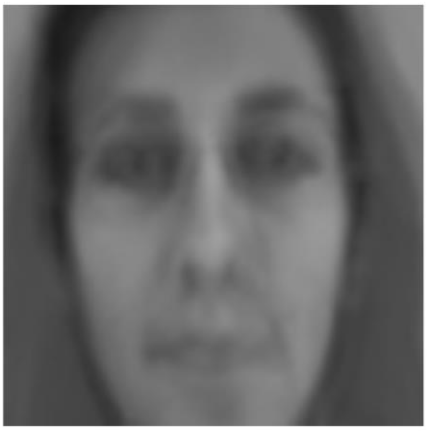

Figure 4.2.3 Example of an average face Ψ

Each of the *NxN* pixels image is then transformed from a two dimensional matrix to a one dimensional vector of length N^2, from which the average face Ψ is subtracted to center the faces distribution. The resulting vector of the image *i* is defined by,

$$\phi_i = \Gamma_n - \Psi$$

and is part of a matrix *A*, composed by all the centered training images.

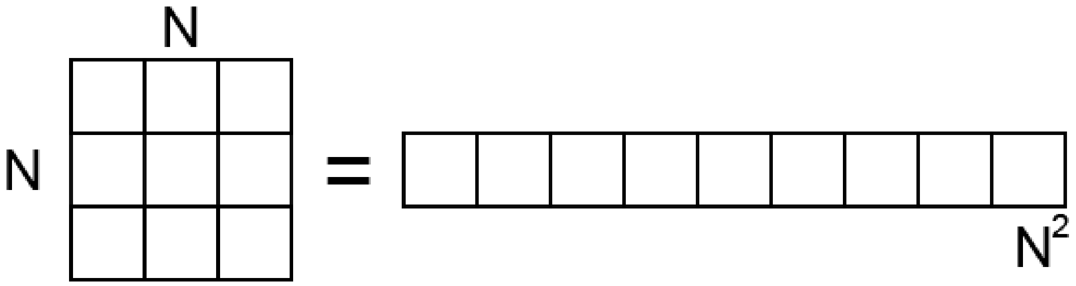

Figure 4.2.2 Face image transformation

To generate a face subspace from the set of M training images, PCA is applied on the matrix A, to extract a set of M orthonormal vectors u_n which best describes the distribution of the data, chosen such that each eigenvalues λ of the covariance matrix $C = AA^T$ is a maximum, with the corresponding eigenvectors being orthonormal. The covariance matrix C is symmetric by construction, with its eigenvectors forming a basis where any of the matrix vectors can be written as a linear combination of the eigenvectors.

Those values are defined by,

$$C = AA^T = \frac{1}{M}\sum_{n=1}^{M} \Phi_n \Phi_n^T$$

$$\lambda_i = \frac{1}{M}\sum_{n=1}^{M} (\mathrm{u}_i^T \Phi_n)^2$$

The covariance matrix C is a measure of the relation between two different images, whose size of $N^2 x\, N^2$ elements makes the eigenvectors and eigenvalues calculation a difficult task; anyway since the amount of images is much less than the dimension of the space $M \ll N^2$, many of the eigenvectors will have an associated value of zero, with only $M-1$ being meaningful. This suggests that the calculation effort can be largely reduced, and it is indeed accomplished by considering the eigenvectors v_i of A^TA instead of matrix C, such that,

$$A^TA\mathrm{v}_i = \mu_i\mathrm{v}_i$$

by the definition of eigenvector. By multiplying both sides by A,

$$AA^TA\mathrm{v}_i = \mu_iA\mathrm{v}_i$$

$$\mu_i = A\mathrm{v}_i$$

$$\lambda_i = \mu_i$$

thus $C = AA^T$ and A^TA have the same eigenvectors $A\mathrm{v}_i$ and, being A^TA of size $MxM \ll N^2x\,N^2$, the computational cost is significantly reduced. Following this result it is possible to find the *M* eigenvectors v_l of $L = A^TA$, where $L_{mn} = \phi_m^T\phi_n$, which determines linear combinations of the *M* faces of the training set to form the eigenfaces u_l defined as,

$$u_l = \sum_{i=1}^{M}\mathrm{v}_{li}\phi_i$$

with l = 1, 2, …, M.

Summarizing, instead of directly calculating the eigenvectors of the vast covariance matrix C, the eigenvalues defining the face space and the corresponding eigenvectors are obtained from a much smaller matrix L, where if λ_i are the eigenvectors of L, then $A\,\lambda_i$ are the eigenvectors for C.

The face subspace can be also calculated in an equivalently efficient and robust approach with *Singular Value Decomposition* (SVD), a more general matrix decomposition method from Linear Algebra, which for a matrix X is defined as,

$$X = UDV^T$$

where U and V matrices have orthonormal columns, and necessary eigenvalues can be extracted from the singular values of the diagonal matrix D. Matrix U serves the same function of matrix L previously described, and contains the eigenvectors to compose the face subspace.

The eigenvectors of the matrix L span a basis set which to describe face images, and are used to find the similarities between subjects. This is done by projecting a face image Γ into the calculated face subspace with a lightweight operation,

$$\omega_k = \mathrm{u}_k^T(\Gamma - \Psi)$$

where the amount of eigenvectors k defines the number of components of the face subspace. The output is a weight vector $\Omega^T = [\omega_1, \omega_2, \dots, \omega_k]$ that describes the relevance of each eigenfaces in the representation of the source image Γ

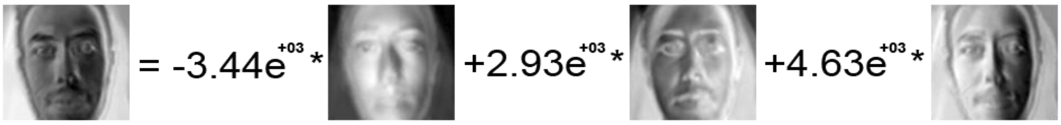

Figure 4.2.3 Eigenfaces composition of a projected face image

The weight vector Ω representing the image is compared with the previously learned faces counterparts, searching for the closest one according to Euclidian distance,

$$\varepsilon_c^2 = \|\Omega - \Omega_c\|^2$$

where *c* is the vector describing the cth face class, calculated averaging all projected images vectors of a subject (or the cth subject if classes are composed by a single image). A face image Γ is classified as belonging to class c, if the distance ε_k^2 results below a certain threshold θ_ε, otherwise it is considered an unknown non-authorized subject.

An additional threshold θ_{f} is used to detect whenever non-faces images are projected onto the face subspace, eg. face detector phase returning a false positive, as in this case the resulting reconstructed image will be likely to resemble a known face anyway. To prevent misclassification caused by this event, the *face space distance* between the mean centered original image ϕ and its reconstruction ϕ_f is calculated as following,

$$\phi = \Gamma - \Psi$$

$$\phi_f = \sum_{i=1}^{M\prime} \omega_i u_i$$

$$\varepsilon_{FS}^2 = \left\|\phi - \phi_f\right\|^2$$

where M' are the dimensions of the face subspace. If the θ_f threshold is below a defined value, the sample is considered a face whose distance from known subjects is to be evaluated, otherwise it is considered a non-face and discarded.

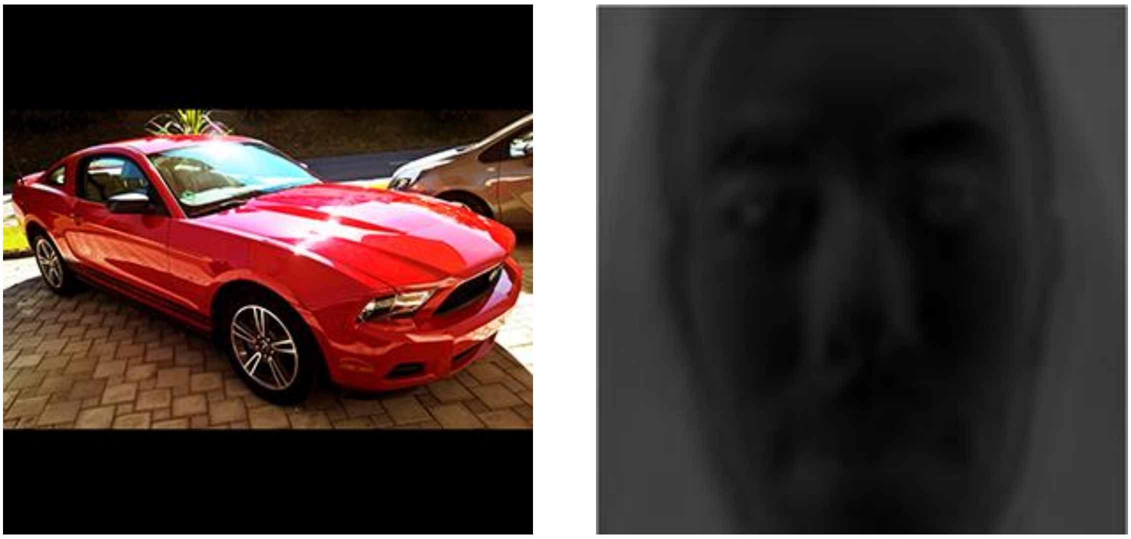

Figure 4.2.4 Non-face image (left) and its reconstruction (right)

As a result of the described procedure, by evaluating the distances it is possible to discriminate if an image never seen before could belong to the class of authorized subjects, all by using operations which mainly requires only point by point image multiplications and sums, with fast execution times. This approach moves the similarity calculations from an N-dimensional difference in image (pixel) space, to an M-dimensional difference in face space.

Furthermore there are a number of practical aspects to consider, such as the number of eigenvectors M' required to span an accurate face subspace, the distance threshold values or the presence of false positives input coming from the output of the face detection phase. The first can be dealt with by searching for the location along the decreasing eigenspectrum where the eigenvalues start to drop significantly, as smaller eigenvalues are related to fine subject details which might not be interesting for the recognition task, or through a practical approach by inspecting the recognition performances as the number of eigenvectors increases. An appropriate distance threshold can be infer by inspecting the distances belonging to the two classes of subjects, to individuate the values yielding a good separation between authorized and non-authorized groups.

For the latter, if a false positive image (eg. non face) passes all face detector classificator stages and is fed as input to Eigenfaces, it has to be managed evaluating if the image will be used for training phase, or for subject validation phase. In the training case, a false positive sample can be detected by calculating its face space distance with respect to an existing face subspace. If $\varepsilon_{FS}^2 > \theta_{\mathrm{f}}$ namely the face space distance exceeds the defined threshold, the sample is successfully detected as a non-face and treated accordingly. In the subject validation phase, the false positive sample will be evaluated similarly to the training phase approach described above, with the additional task of requesting a new image to the source video stream, an almost effortless operation as the video stream provides a more than adequate amount of images with a frequency of 25 frames per second.

The Eigenfaces method has some known limitations, some of them being PCA direct consequences: it strongly relies on the direction of maximum variance to construct the auxiliary structures, but this could not be the best method for classification. In addition it assumes an underlying Gaussian distribution of the data, and most important its linear separability. If those conditions aren't valid, the principal components may not approximate efficiently the data, resulting in a flawed face subspace.

There's also another issue directly related to PCA behaviour in following the directions of maximum variance. Suppose the set of images of a subject are taken with widely variable illumination conditions, PCA may parse this as the main source of variance, with the result of having principal components emphasizing the *intrapersonal* variability within a subject, rather than the *extrapersonal* variability between subjects.

Other limitations concern the input face images format and variability, as illumination, face orientation and alignment changes between training set samples may affect the recognition performances. Different face scaling or sizes affects the performances even more considerably, as pixel correlation between images is lost.

As a general rule it is appropriate to train the model with a collection of images characterizing the variability of conditions, and uniform the input face proportions.

5. Experimental setup

5.1 Hardware description

The feasibility of proposed solution has been evaluated through a demonstrative system using face detection (Viola-Jones) and face recognition (Eigenfaces) algorithms, composed by the following hardware:

1. DVC/IP, external entry panel equipped with VGA CMOS sensor camera, LED illuminator, ARM CPU with video encoding capabilities and call button, running a LTIB Linux distribution.
2. XTS IP 7, internal video receiver equipped with ARM CPU with video decoding capabilities, 7 inches touchscreen, running a YOCTO Linux distribution.

Both devices are POE powered and communicates through a set of proprietary and open IP based protocols. The external entry panel is responsible for digitize and, when necessary, provide additional light to brighten the environment and the subject face. Due to its wide angle lens configuration the recorded stream has to be linearly transformed with a de-warping filter according to a block stretching matrix, to adjust the frames proportions. The stream is then encoded by the DVC/IP on board CPU to 25fps H.264, and sent through RTP protocol to internal video receiver for detection, recognition and authorization phases.

On the internal video receiver side, the H.264 stream is decoded into BGRA RAW (4 bytes: blue, green, red, alpha), with its frames being sampled and used as input data for computer vision techniques during both training and validation phases. The resolution of the video stream is limited as a result of the choice of the DVC/IP entry panel device with its VGA (640x480) camera CMOS sensor; while more powerful devices were available, this choice allowed to simulate the most common application case, with low quality and possibly noisy images used as input.

5.2 Faces dataset

The face dataset has been generated by sampling the RAW stream, decoded from the received entry panel video. Two-dimension images of fourteen test subject have been gathered with different ambient lighting level, facial expression and combinations of both according to the table below; subjects are looking directly to the camera with their face perpendicular to the objective, in order to simulate the most common real world application use case. The dataset includes also variations of the same set of images, where subjects are using glasses to alter their facial appearance, having around 55 images per subject and a total of 780 pictures in the dataset.

Full environmental light, subject uniformly illuminated	**Partial environmental light, subject illuminated on one side**	**No environmental light, subject illuminated from entry panel LED**
Neutral expression	Neutral expression	Neutral expression
Closed eyes	Closed eyes	Closed eyes
Smiling	Smiling	Smiling
Looking in different directions	Looking in different directions	Looking in different directions
Repeating a predefined sentence	Repeating a predefined sentence	Repeating a predefined sentence

Table 5.2.1 Dataset structure for standard images

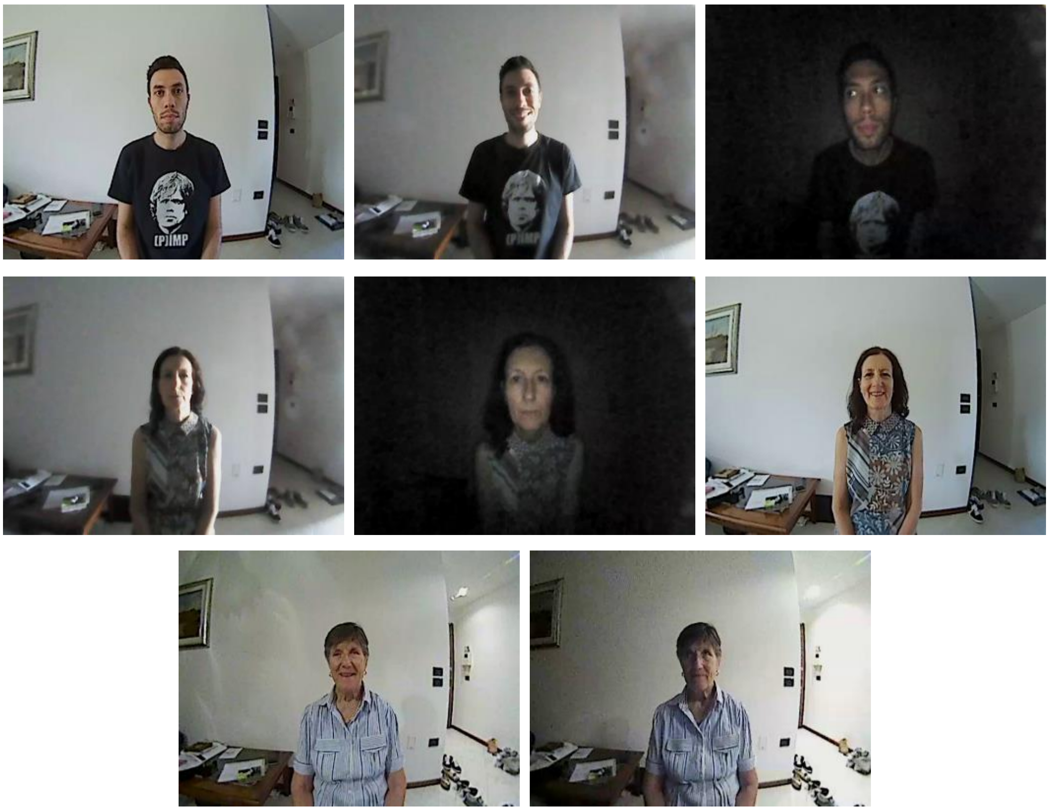

Figure 5.2.2 Some dataset sample images

The number of pictures necessary to cover the given combinations may, at first glance, appear high for a consumer user willing to generate the dataset of its family members, but in fact only two to four properly recorded session (which includes all characteristic views) per subject should be necessary, as the camera sensor sampling rate is 25 frames per second. The extension of ambient lighting changes through the day also affects the number of necessary recording sessions.

To simulate system real world usage, it has been decided to create a completely new dataset, instead of relying on one of the many already available. Moreover, one of Eigenfaces drawbacks consists in its susceptibility to face position and proportions, meaning those have to be invariant between training and test set, and the provided external entry panel camera uses a lens configuration which distorts the face appearance even after filter processing, thus results with mixed datasets may be biased.

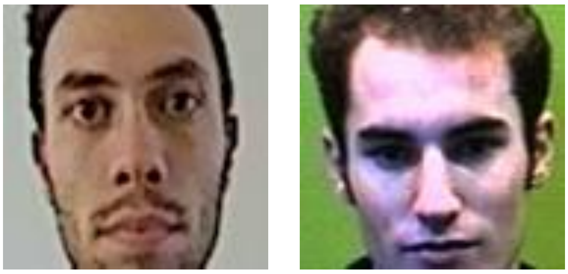

Figure 5.2.3 Proportion differences between the generated dataset and the Faces94 collection

5.3 Face recognition training set

The training set for the experimental setup is generated from the face dataset resized to 90 x 90 and grayscaled, using an increasing number of eigenvectors (from 1 to training set size), to assess their impact on system performances and the overall score. A total of 111 pictures, corresponding to 14% of the faces dataset, or 35% of the first three subjects images, have been used for training set construction, while the remaining images and subjects compose the test set for the user validation phase.

A face detection step using Viola-Jones framework is performed on the faces dataset to remove unnecessary details from the images (eg. background), since those do not contribute positively to subject identification. Since in the majority of cases subject will place itself in a central position with respect to the camera which is installed at eye level, face search area is limited to the upper central part of the input image, to minimize distracting elements such as other subjects, or face like objects. A backup search over the whole image is triggered in case no face is found in the limited area.

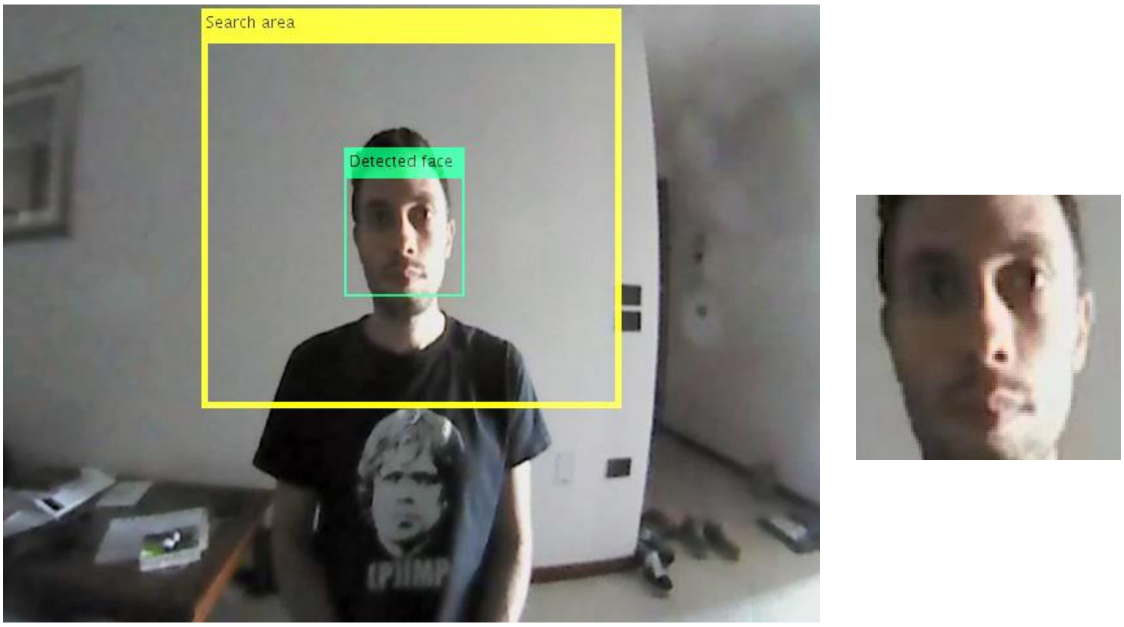

Figure 5.3.1 Face detection example, with search area and detection area (left), and detected face (right)

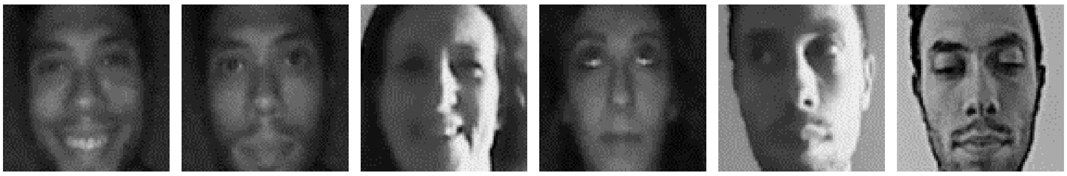

Figure 5.3.2 Training set examples

5.4 Face recognition test set

The test set is a mixture of images from the faces dataset, images of other subjects and some non-face images. A total of 668 images are present in the test set, corresponding to 86% of the faces dataset; first 121, or equivalently 19% of the test set, are standard or altered samples from the authorized class of subjects and the remaining are unknown subjects or non faces images. This partition structure of the test set has been chosen for simulating the most common real world use case, where the number of authorized subjects is far less than the other class; anyway it is necessary to have a reasonable amount of authorized samples, in order to avoid situations where the outcome of the analysis could be highly influenced by the population size.

The face detection step has been performed on the whole dataset, to simulate false positives during face detection phase.

A graphical summary of the experiment steps is reported below.

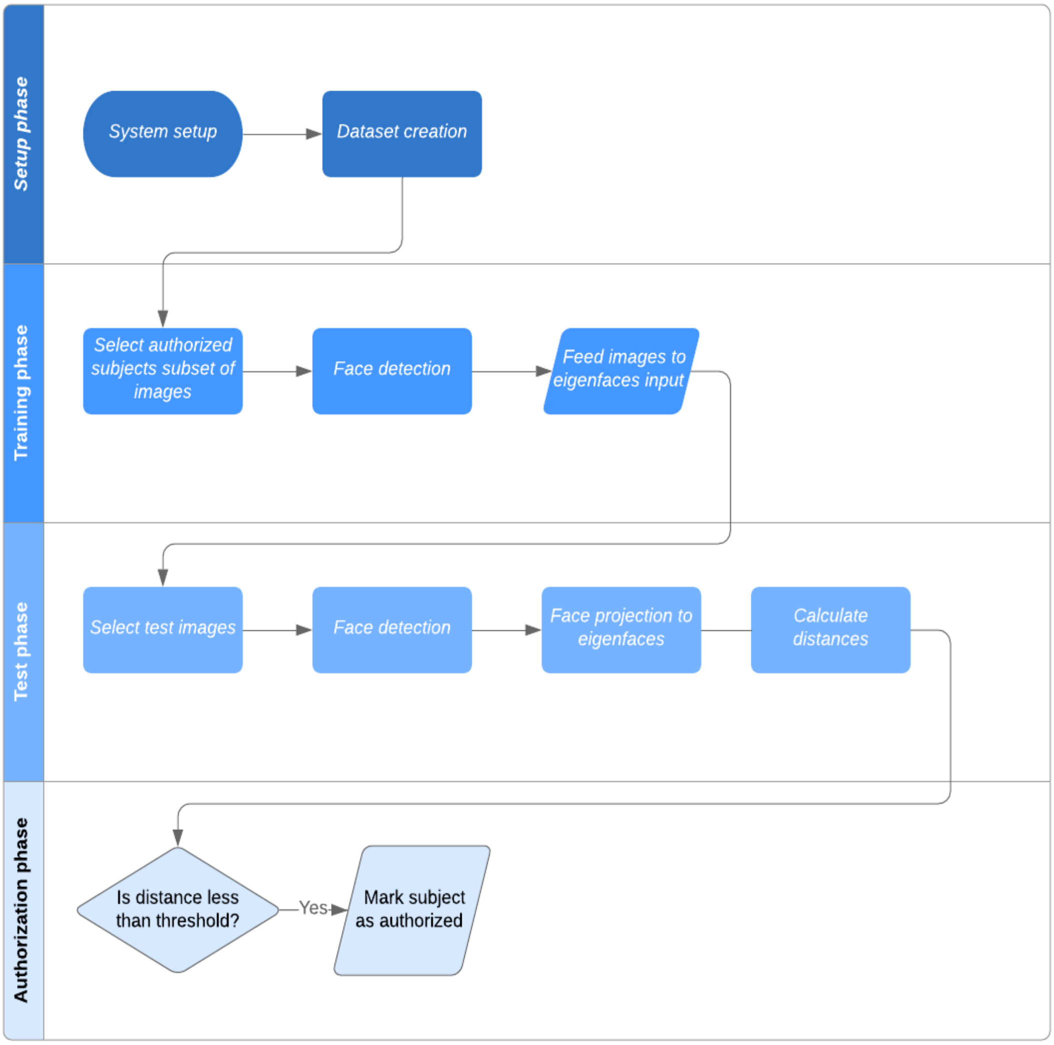
Setup phase
System setup
Dataset creation
Training phase
Select authorized subjects subset of images
Face detection
Feed images to eigenfaces input
Test phase
Select test images
Face detection
Face projection to eigenfaces
Calculate distances
Authorization phase
Is distance less than threshold?
Yes
Mark subject as authorized

Figure 5.4.1 Experiment steps

5.5 Face recognition training phase

During the training phase, images from the training set are provided to Eigenfaces algorithm, to calculate the mean face and create a face subspace, the main output of this phase. Any element of the training set can be then compressed and reconstructed, by projecting it onto the calculated face subspace.

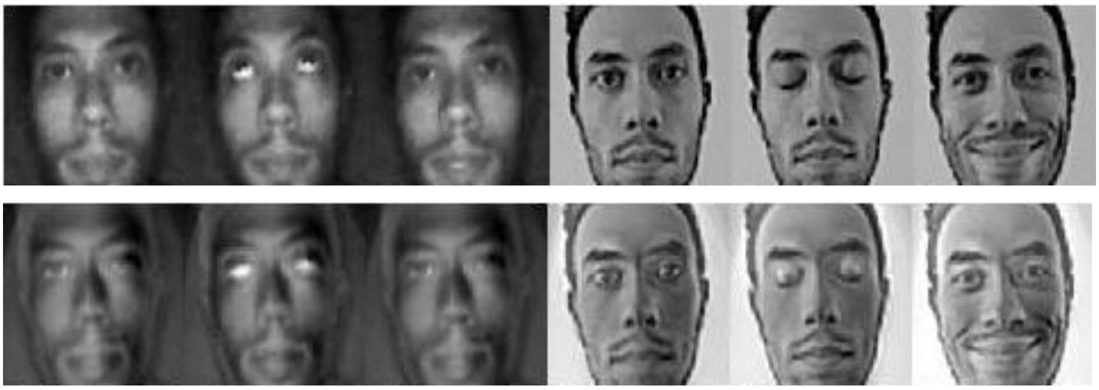

Figure 5.5.1 A subset of the training set (top), its projection and reconstruction on the face subspace (bottom)

Since the face subspace vectors have the same dimension of the input pictures, they can be displayed as images called Eigenfaces, ordered from the largest to the smallest eigenvector, hence from largest to tiniest details of training set faces.

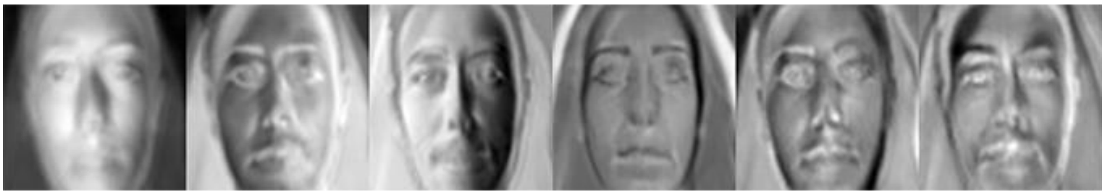

Figure 5.5.2 The first six eigenfaces

5.6 Face recognition testing phase

Once the face subspace has been created, face distance (or similarity) can be evaluated by projecting each test image on this subspace: this operation produces a set of weight per picture, which are compared to the training set weights using Euclidian distance. If the resulting values are less than the defined thresholds, the test image is associated with the closest training set image. Once evaluated the test images are considered belonging to “authorized” or “unauthorized” subjects classes.

The resulting weight are also used to calculate the distance from test set image and the face subspace itself, by evaluating the Euclidian distance between the test set images and their reconstruction.

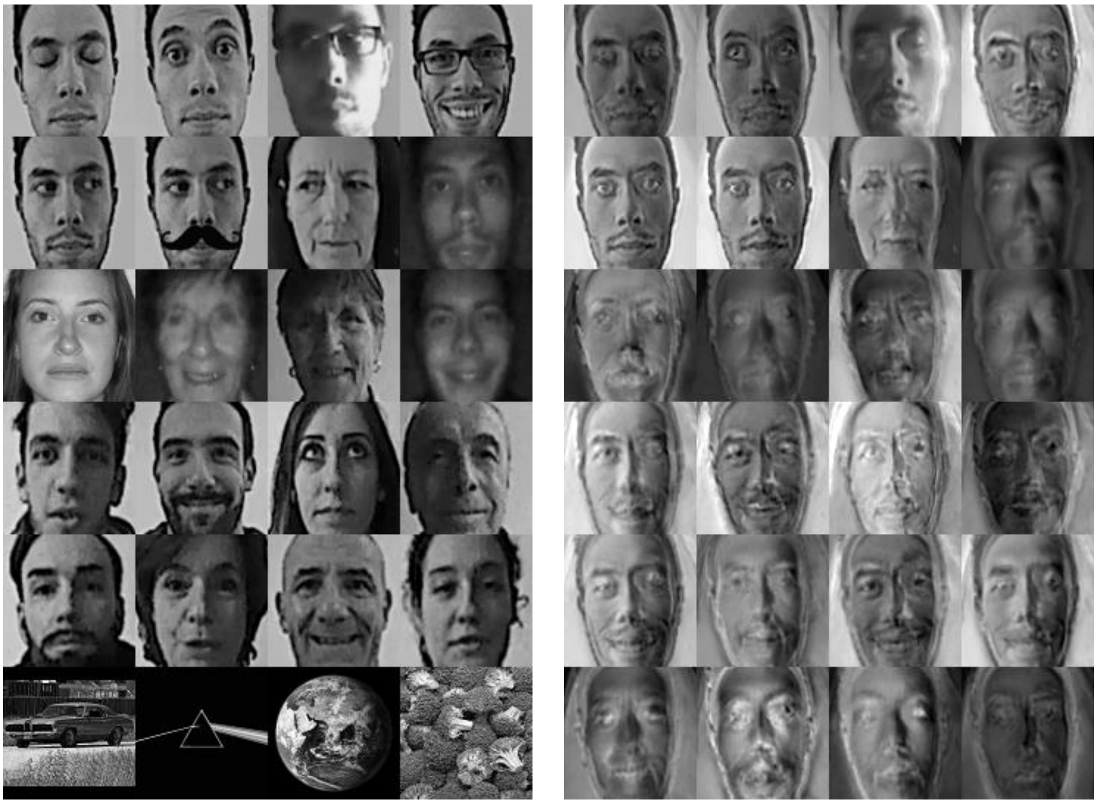

Figure 5.6.1 - Some test set images (left) and their reconstruction in the face subspace (right) using the first 30 eigenvectors

The obtained distances are used to discern each of the four possible outcomes, and act accordingly the following schema:

Number	Description	Authorize
1	Test image is a face and is a known subject	Yes
2	Test image is a face and is an unknown subject	No
3	Test image is not a face, but its reconstruction is a known subject	No
4	Test image is not a face, and its reconstruction is an unknown subject	No

Table 5.6.2 Outcomes for classification process

6. Experimental results

The application of the Eigenfaces algorithm has been tested on the described test set, assessing its performance by increasing the number of Eigenvectors used to create the face subspace, from 1 to 111 (the training set size) with multiple runs and randomized training subject images position. The number of subjects in the training set, their face position and the distance threshold are instead invariant.

A good classifier for the project intended goal is expected to avoid all False positive type of errors (authorizing an incorrect subject), while keeping the False negatives (not authorizing a valid subject) as low as possible. It is also important for the solution to be able to generalize the authorized subject facial features, in order to recognize properly those, when slightly variations or alterations are present.

As a starting point, each test image has been considered to belong to an authorized subject, thus allowing access, if its reconstruction distance from the face subspace is less than 4.000 units, and distance from the closest training sample is less than 2.200 units. Those initial thresholds have been empirically determined by considering preliminary scores. The results are shown below using various graphical methods.

The performances of the proposed solution shall be evaluated mainly in terms of number of errors on the test set, where a reconstructed face is considered incorrectly classified if:

1- The original image is a known subject, but it is either classified as unknown subject or non-face, leading to a false negative

2- The original image is an unknown subject, but it is classified as known subject, leading to a false positive

3- The original image is a non-face, but it is either classified as face, known or unknown subject, leading to a false positive

Although the focus of this chapter is on the final recognition performances, it is worth to note that the face detection phase took 1.25 seconds to process each image for a total of 835 seconds for the 668 given images, with a 99.2% detection rate of which 0.8% are false positives.

Figures 6.1a and 6.1b represent the precision of the Eigenfaces application on the given test set in terms of number of incorrect classifications or errors. These preliminary results shows that, while the algorithm performances are poor with few eigenvectors, it is able to improve in reconstructing and recognising the test set by using at least the ten largest eigenvectors, while performs similarly by increasing even more the amount of eigenvectors considered. The overall performances are anyway not acceptable as the percentage of errors is above 20% of the test set size, most of them belonging to the false positive type of error, resulting in non-authorized subject being able to access. Figure 6.1a shows the total amount of errors further subdivided in false positives and false negatives.

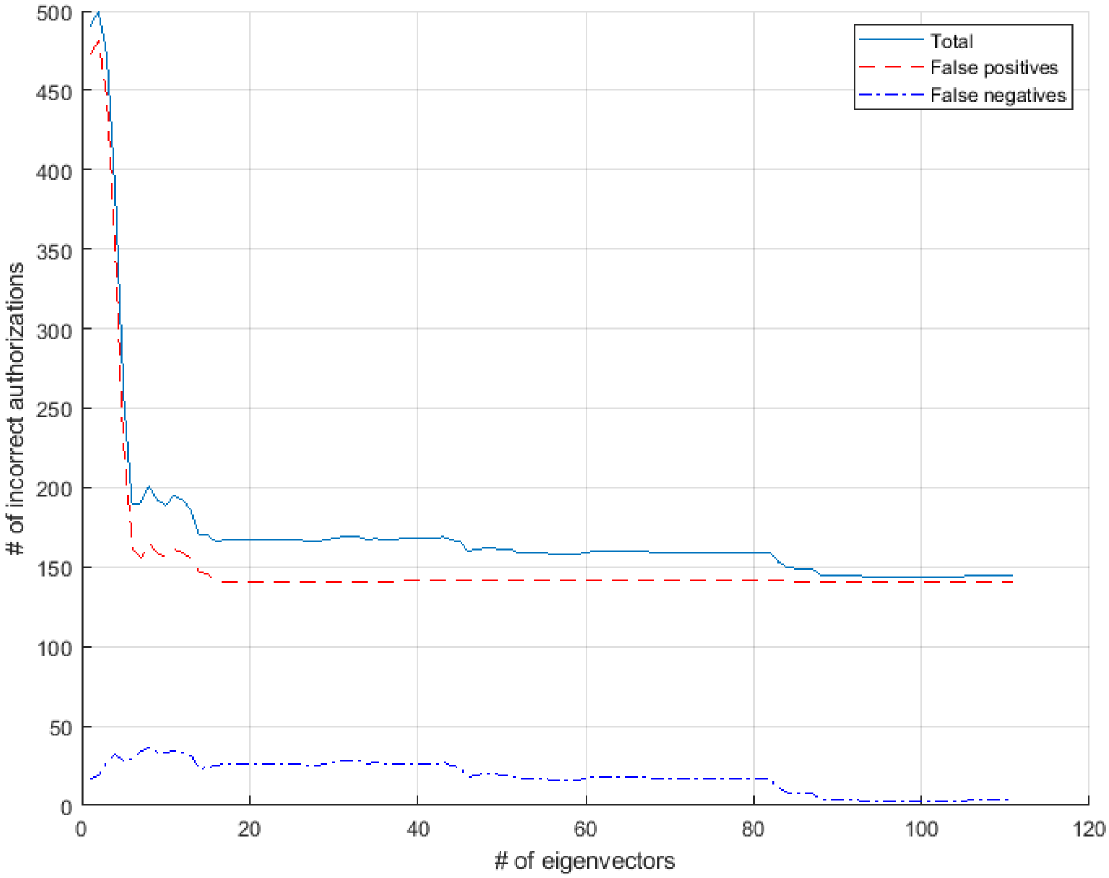

Figure 6.1a Comparison of number of errors on the test set

Figure 6.1b shows the data in terms of error percentage on the test set, where false positives percentage is calculated on the non authorized class samples and the percentage of false negatives on the authorized class samples only.

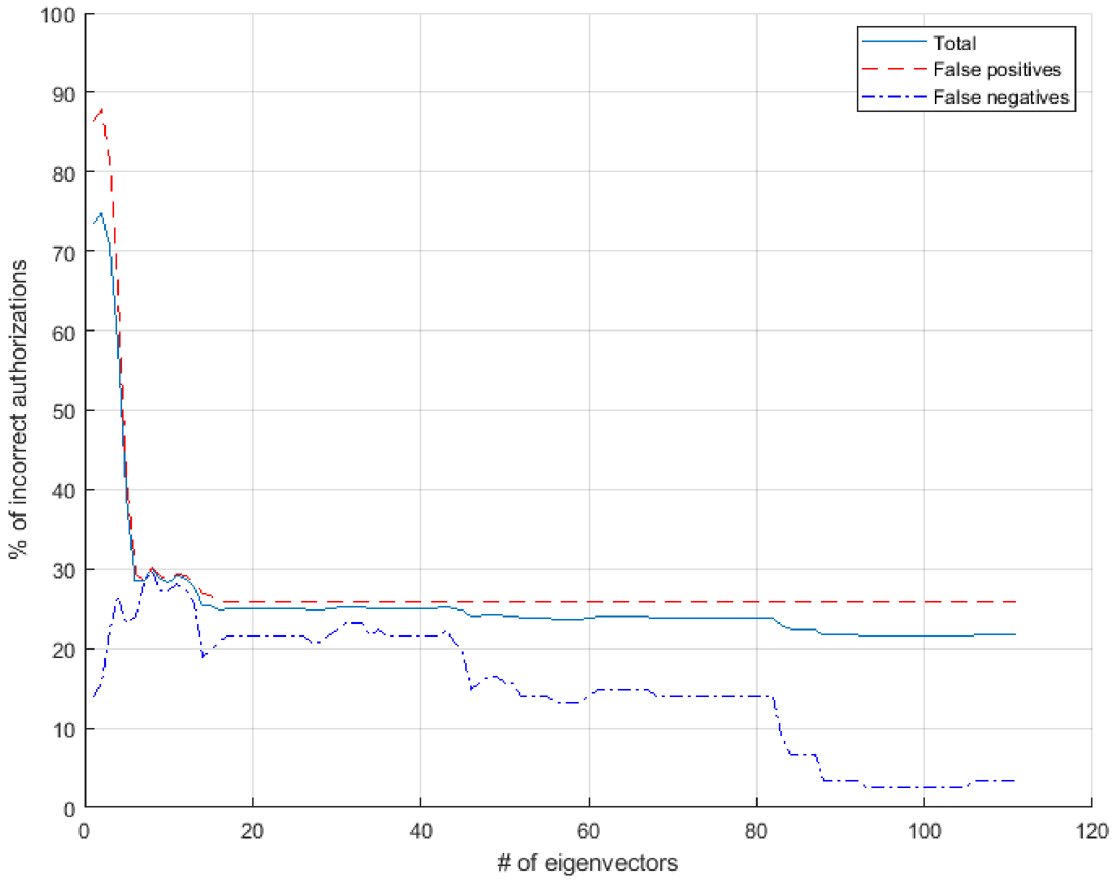

Figure 6.1b Percentage of errors on the test set

To investigate the causes of such high classification error the resulting scores of various test samples have been analysed. A subset of the test images is shown in Figure 6.2 exhibiting the details and scores of Eigenfaces method obtained using around 100 eigenvectors, namely the amount which minimizes the overall error.

Subjects test images are divided in two sections; authorized are marked by a green box, while non-authorized by a red one. The images are then grouped in blocks of two, where for each test image (left) the corresponding closest detected training sample is shown on its right side, with the distances from the closest training sample and the face subspace displayed on the bottom. The algorithm is able to classify all the provided test samples subset correctly, excluding the two highlighted in orange belonging to the low environmental light group of images. This is a known limitation of the Eigenfaces method, which is sensitive to lighting variations, and in this case it is incorrectly considering the amount of background "blackness" between images as a major source of similarity, rather than exploiting the subjects facial traits.

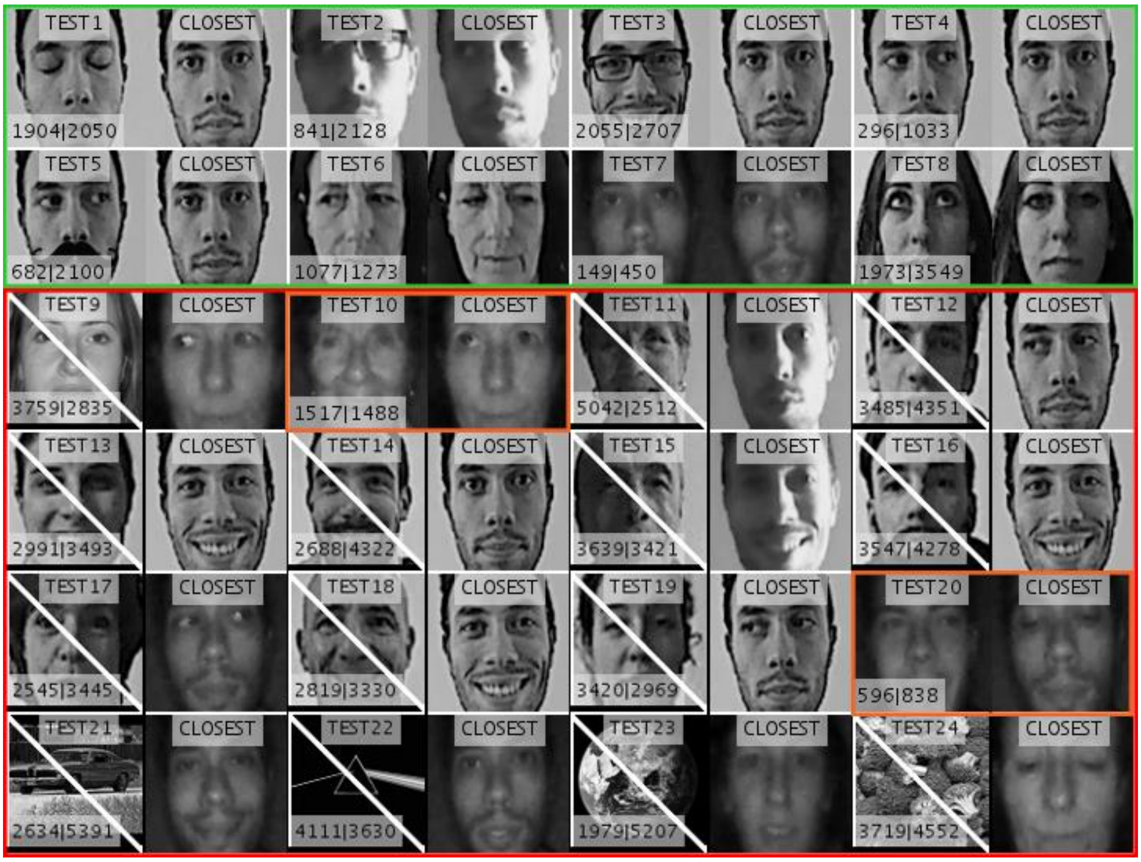

Figure 6.2 Recognition result samples details

The adopted approach for addressing this issue is composed of two improvements, image contrast enhancement and additional dedicated threshold.

Contrast enhancement consists in adjusting the images with the aim of further separating the face from the background, and highlight the face traits to make them emerge as the relevant information to consider. This is applied to both training and test set before being used as input for Viola-Jones detector, and consequently for Eigenfaces recognition phase.

The modification is not indiscriminately applied on all the dataset, as contrast is enhanced only on images having an amount of illumination inferior to a certain threshold, which is calculated by looking at the mean value of normal and low light samples in defined areas, namely the first left column and top edges. The value of this "blackness" threshold has been set to 50, as the abovementioned areas have been found to have a pixel mean value of 100 in daylight images, and around 30 in low light images. An example of the effect of its application on a face image is shown in Figure 6.3.

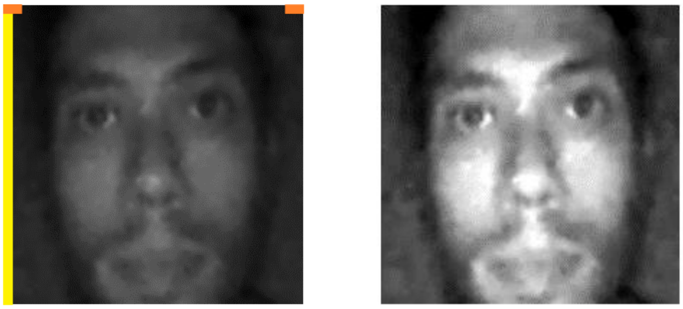

Figure 6.3 Contrast enhancement result (right) on a test sample with highlighted evaluation spots (left)

Applying the contrast enhancement successfully results in an increased distance between non authorized test samples and training samples (Figure 6.4), but it doesn't yield any direct effect on the classification performances, with percentage of errors remaining above 20%. Figure 6.5 shows indeed there's almost no difference in percentage of errors between the initial approach and the runs with contrast adjustment.

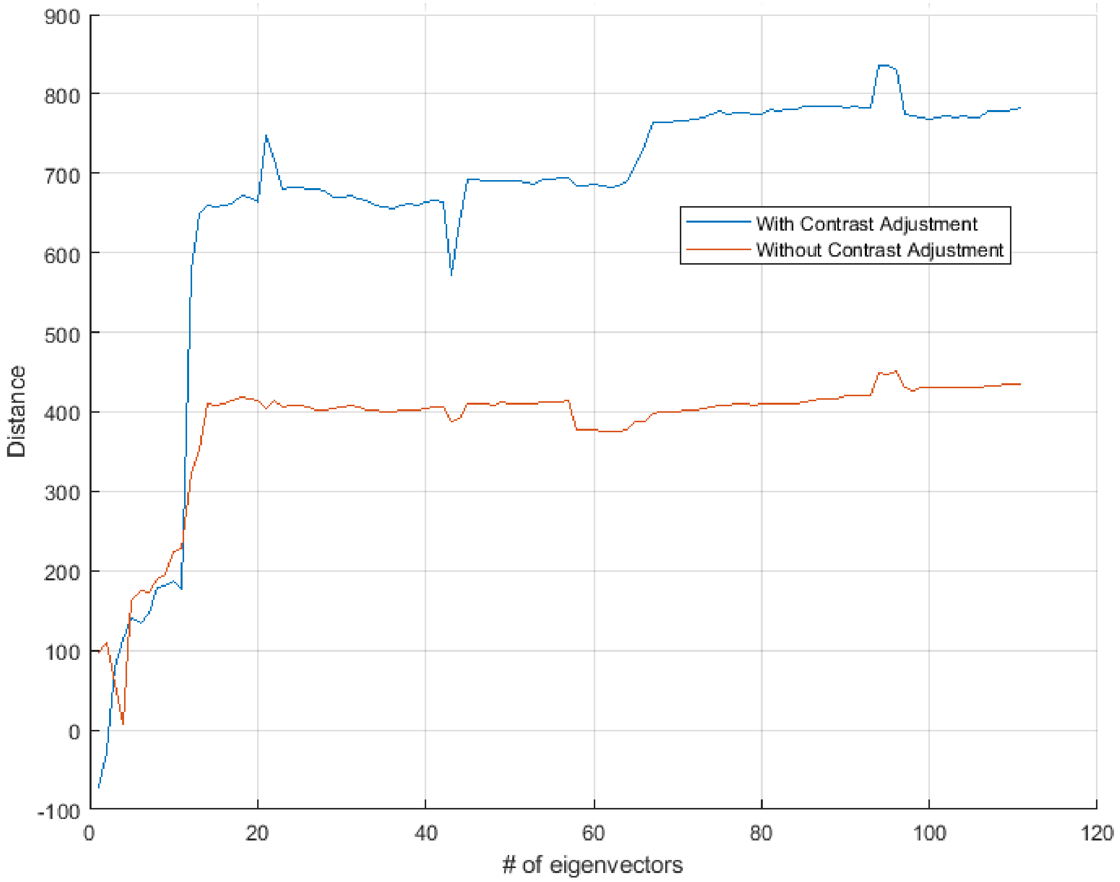

Figure 6.4 Effect of contrast enhancement on the mean distance between classes (low light samples)

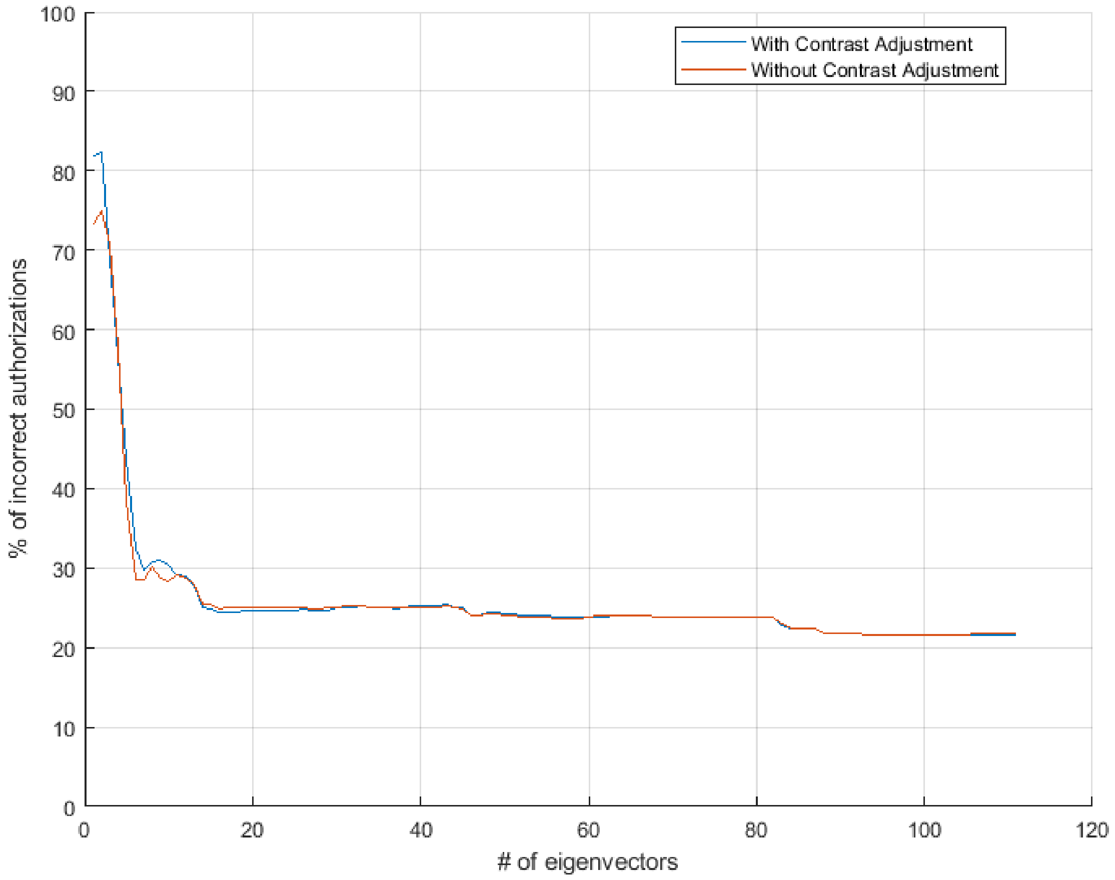

Figure 6.5 Percentage of errors on the test set with contrast enhancement

Since contrast enhancement results in a better separation between samples, it's reasonable to suppose that it can be exploited to increase performances. By turning attention to the low environmental illumination images group, it emerges that the separation between those authorized and non-authorized subjects exists, even though the distance comes with a different proportion when compared to normal illumination samples.

This concept is shown in Figure 6.6, by plotting data coming from a sample execution of the system on the whole test set; low light samples are on the left side of the graph (blue and purple asterisks), and normal light on the right (green and red circles). To exploit this, the two main subject classes are furtherly splitted, by configuring a threshold for normal illumination, and adding a lower one for darker samples.

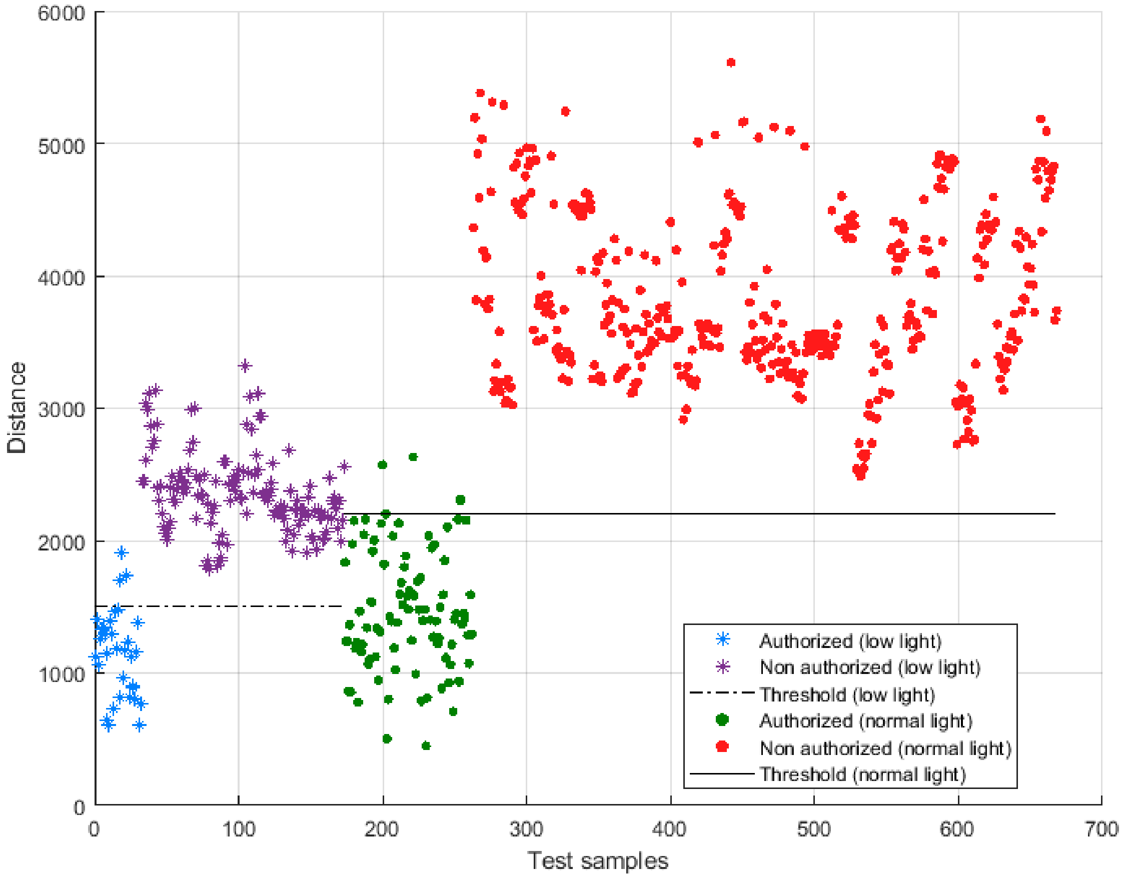

Figure 6.6 Distances from authorized class samples

The threshold for distances from authorized training samples, and the face space distance for the low illumination images group are both set to 1500 units. The resulting classification performances are sensibly improved thanks to this addition, as the system manages to separate correctly the two main classes avoiding the false positives issue caused by low light test images. Figure 6.7a illustrates the reduced number of errors of the recognition phase.

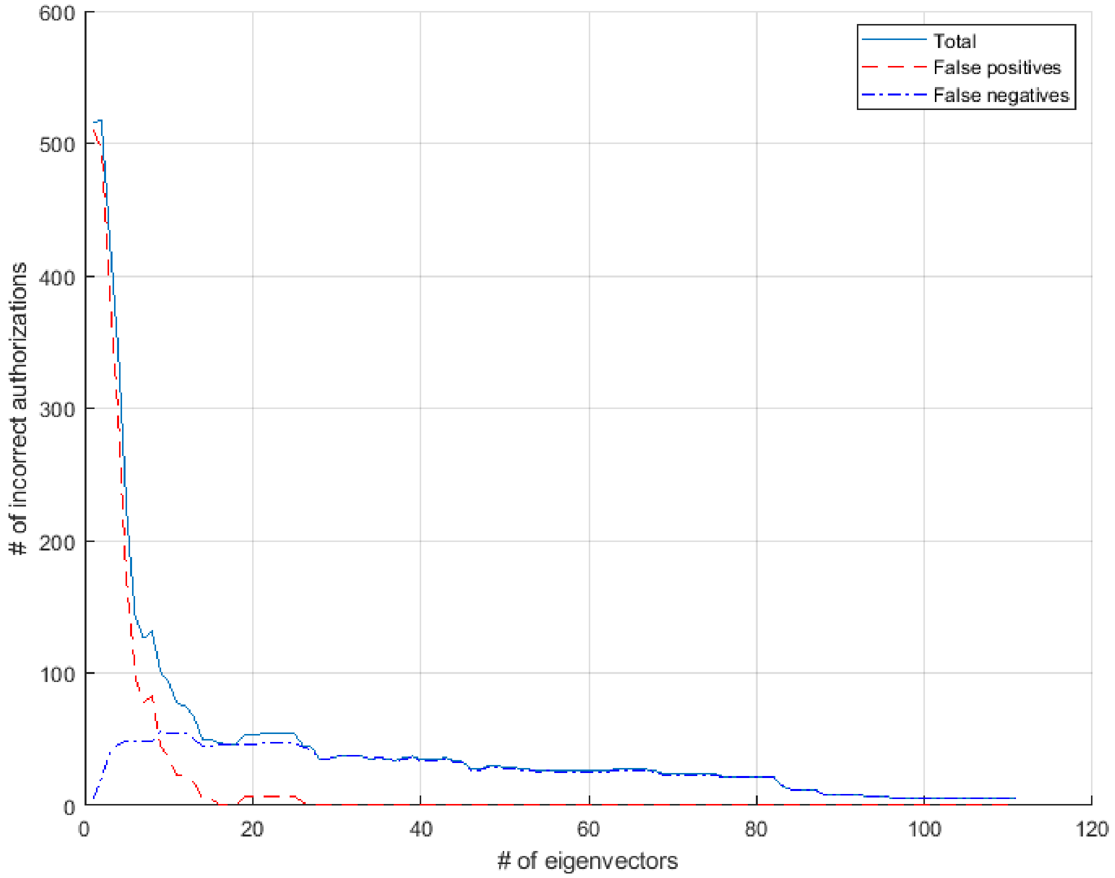

Figure 6.7a Comparison of number of errors on the test set with enhanced contrast and separate thresholds

The percentage of errors shown in Figure 6.7b has dropped from 25% to less than 1%, all due by false negatives errors which accounts for less than 5% of the authorized subject samples.

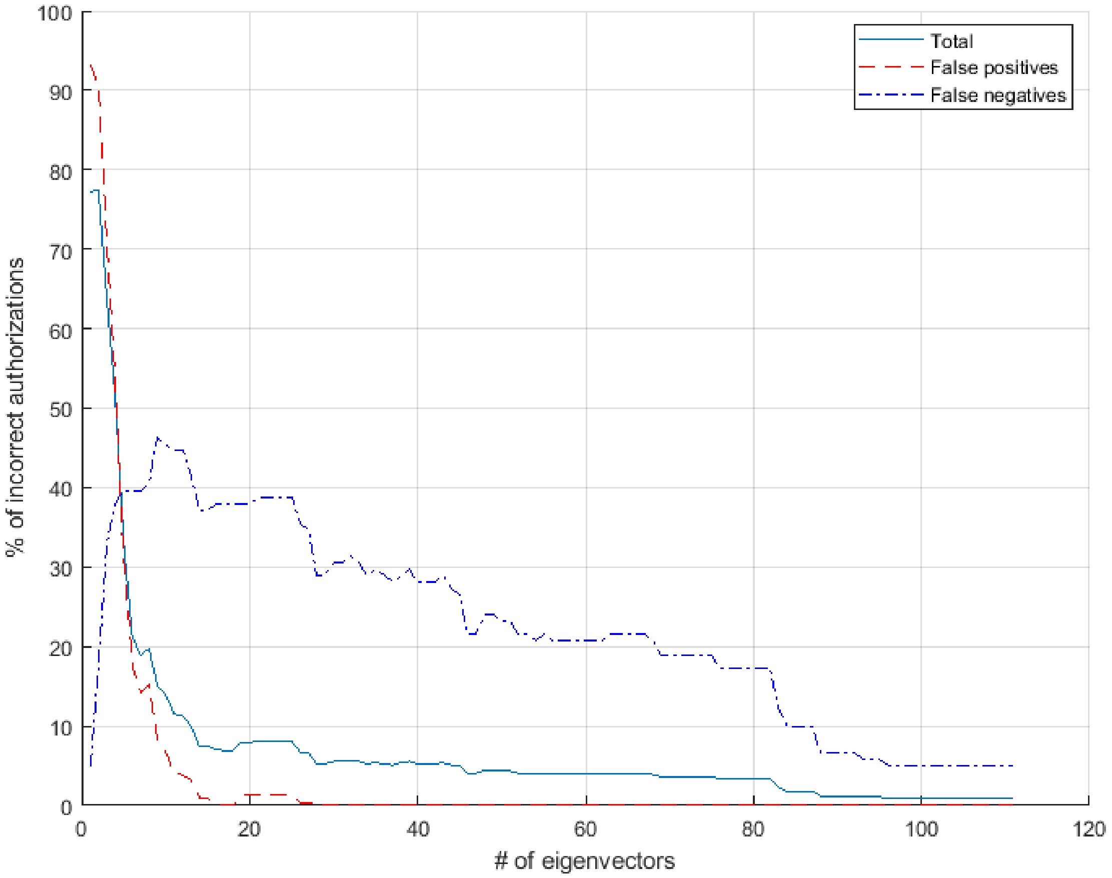

Figure 6.7b Percentage of errors on the test set with enhanced contrast and separate thresholds

Figure 6.7c is an overall comparison in terms of total number of errors between the initial approach, and its revised double threshold version.

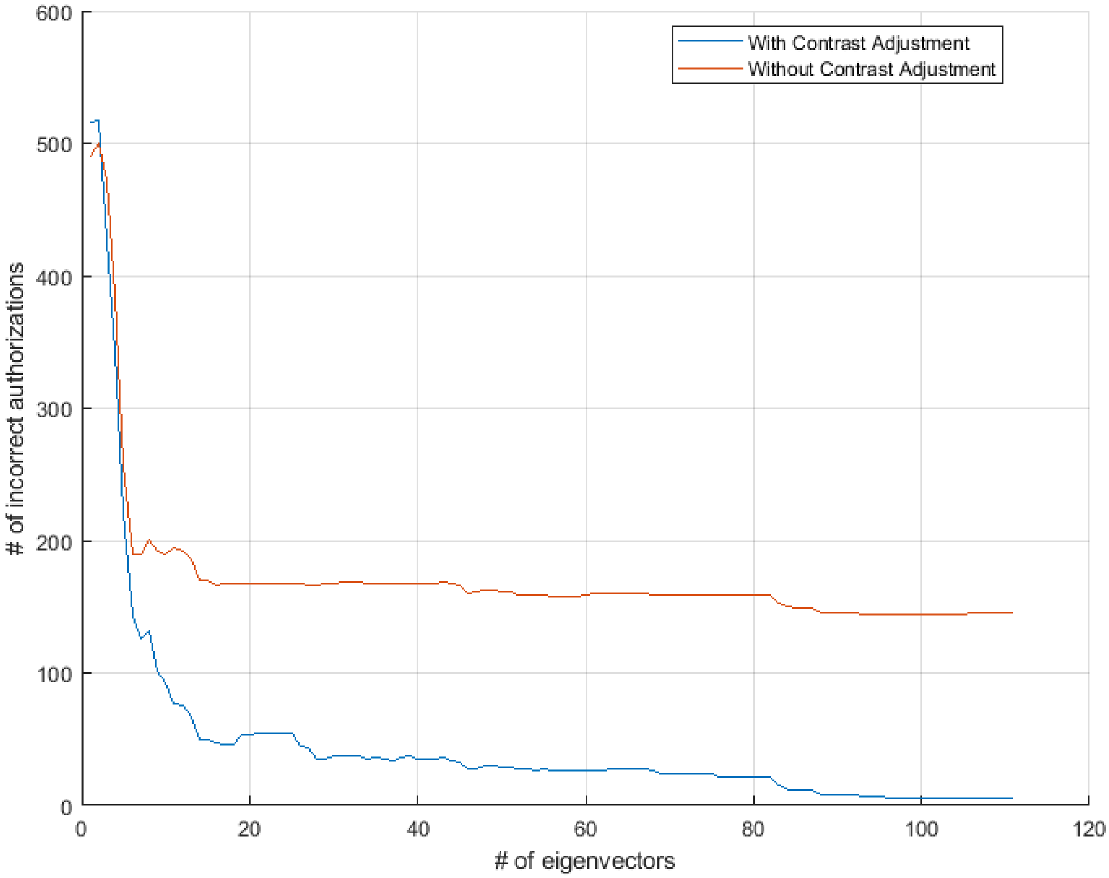

Figure 6.7c Comparison of total number of errors between the two approaches

Details on updated results on the previously selected subset of the test images when the revised approach is applied are shown in Figure 6.8.

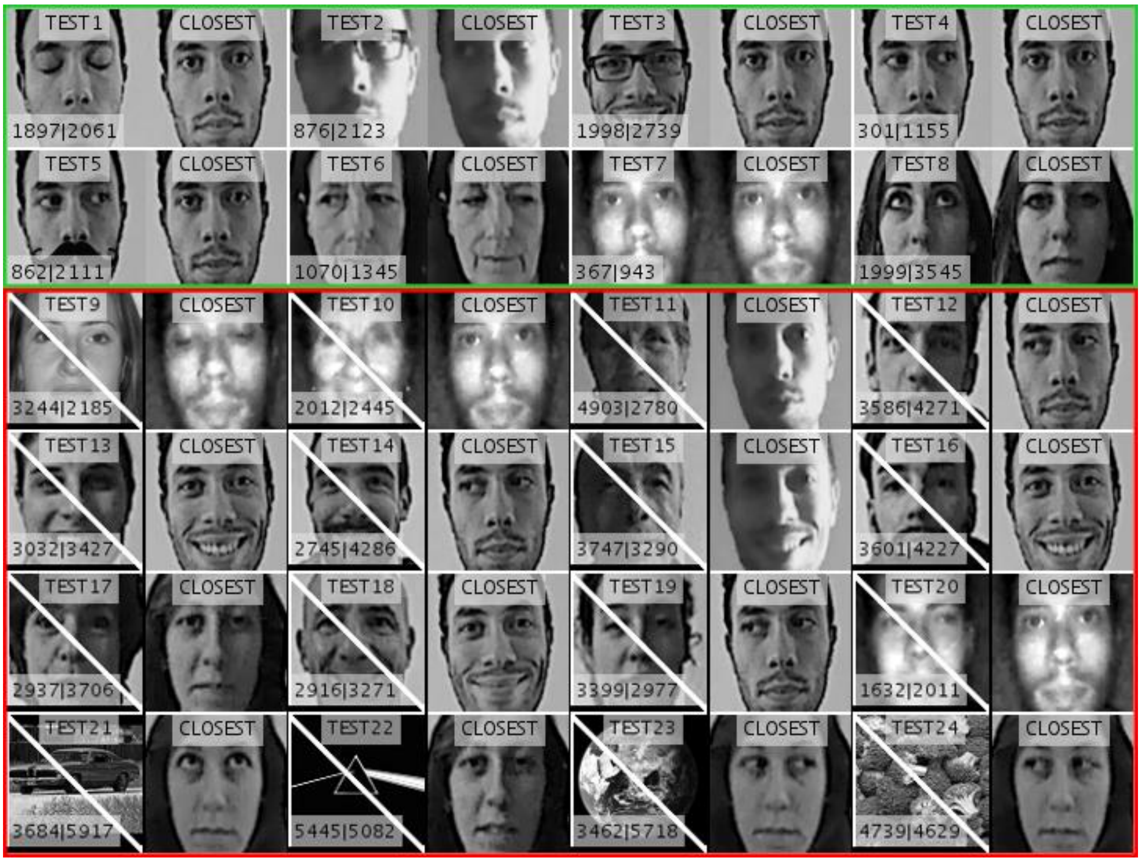

Figure 6.8 Recognition result samples details with enhanced contrast and separate thresholds

Beside the recognition performances on the test set, it is interesting to dwell on some internal results of the experiment, obtained with both contrast enhancement and separate thresholds improvements applied.

Figure 6.9a represents the mean distance between images reconstruction and the face subspace; here the number of eigenvectors directly impacts the performances of Eigenfaces method, contributing to the definition of more precise face subspace boundaries.

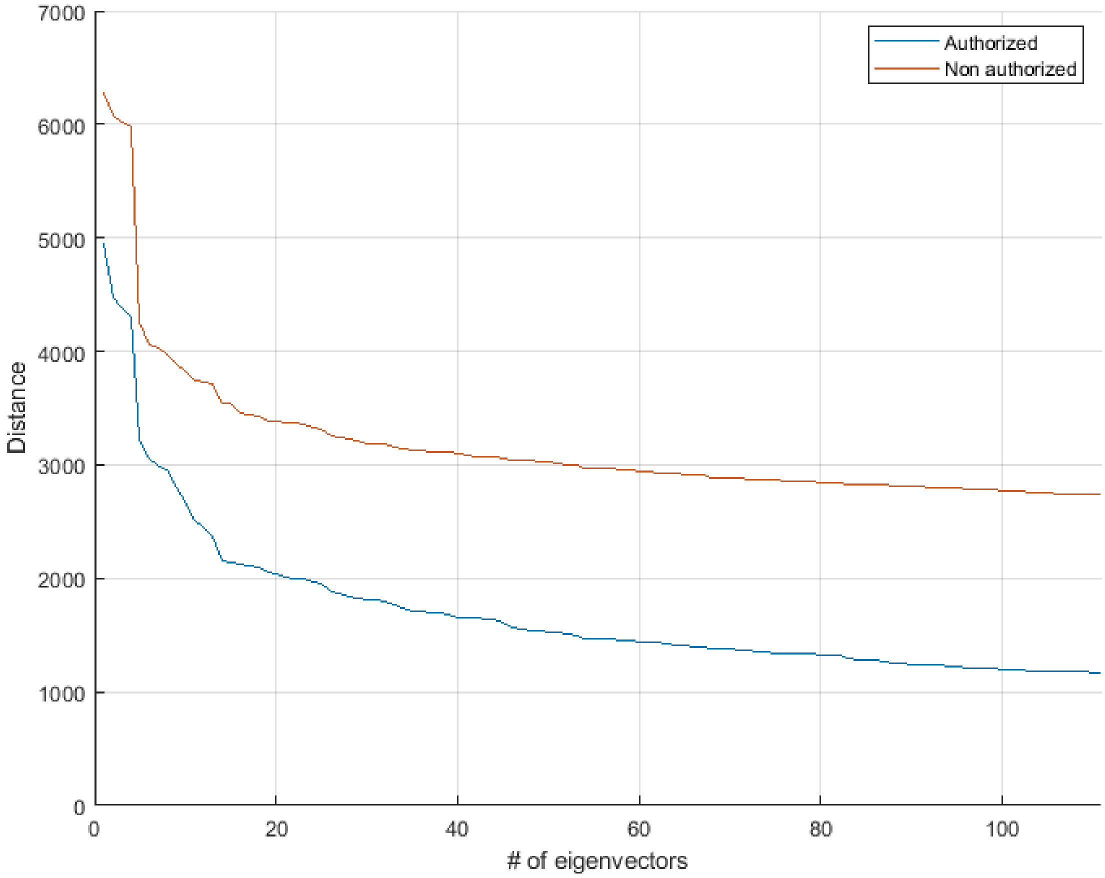

Figure 6.9a Comparison of mean distances from the face subspace

Also, the number of eigenvectors necessary to define a face subspace with good performances in approximating the authorized subjects correctly is modest. As shown in Figure 6.9b, using previously configured thresholds almost all face images are inside the face space, even with a modest amount of eigenvectors.

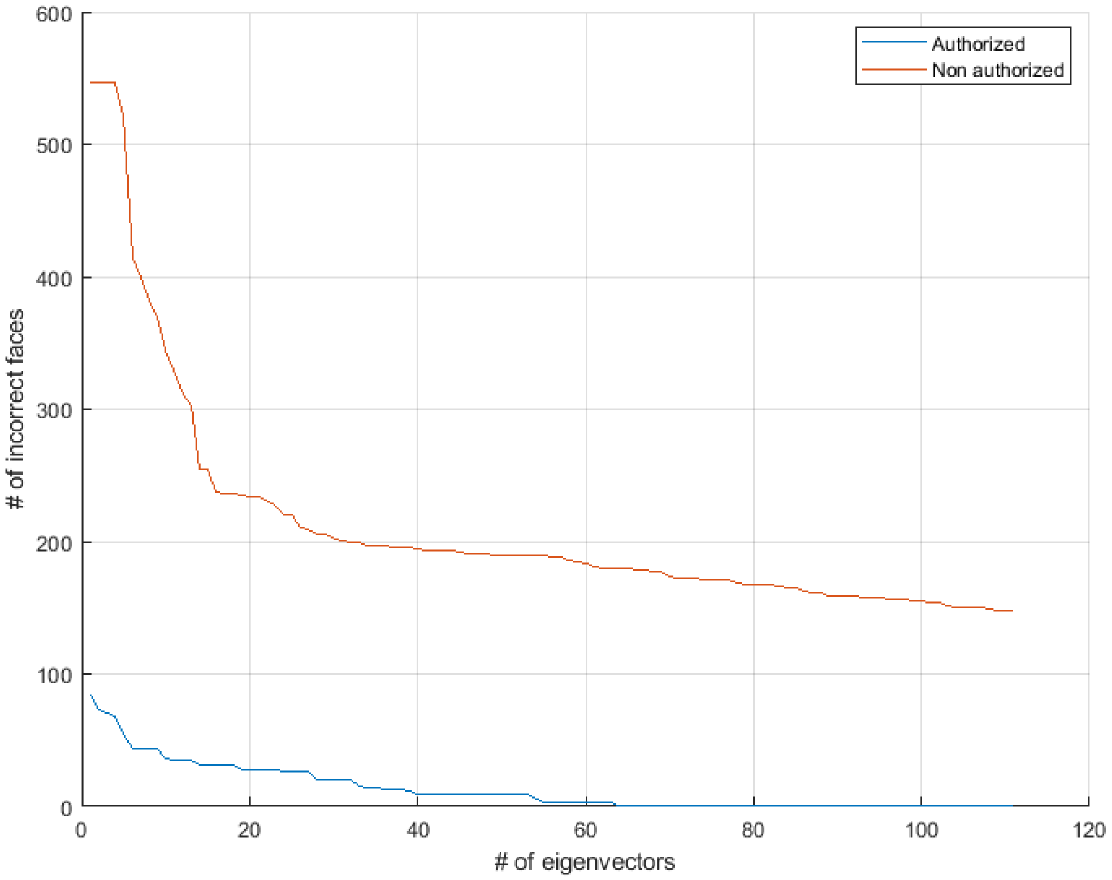

Figure 6.9b Number of faces outside face subspace for the two classes

Figure 6.10 shows how the number of eigenvectors contributes to the distance between test and training images (authorized subjects), with both classes projected on the face subspace. It is a measure of how well the Eigenfaces method performs in approximating the images on the lower dimensional face subspace, and can be interpreted as training and test errors. It is not unexpected to observe that by increasing the number of eigenvectors the error decreases, as the resulting subspace describes with increasing precision the given samples.

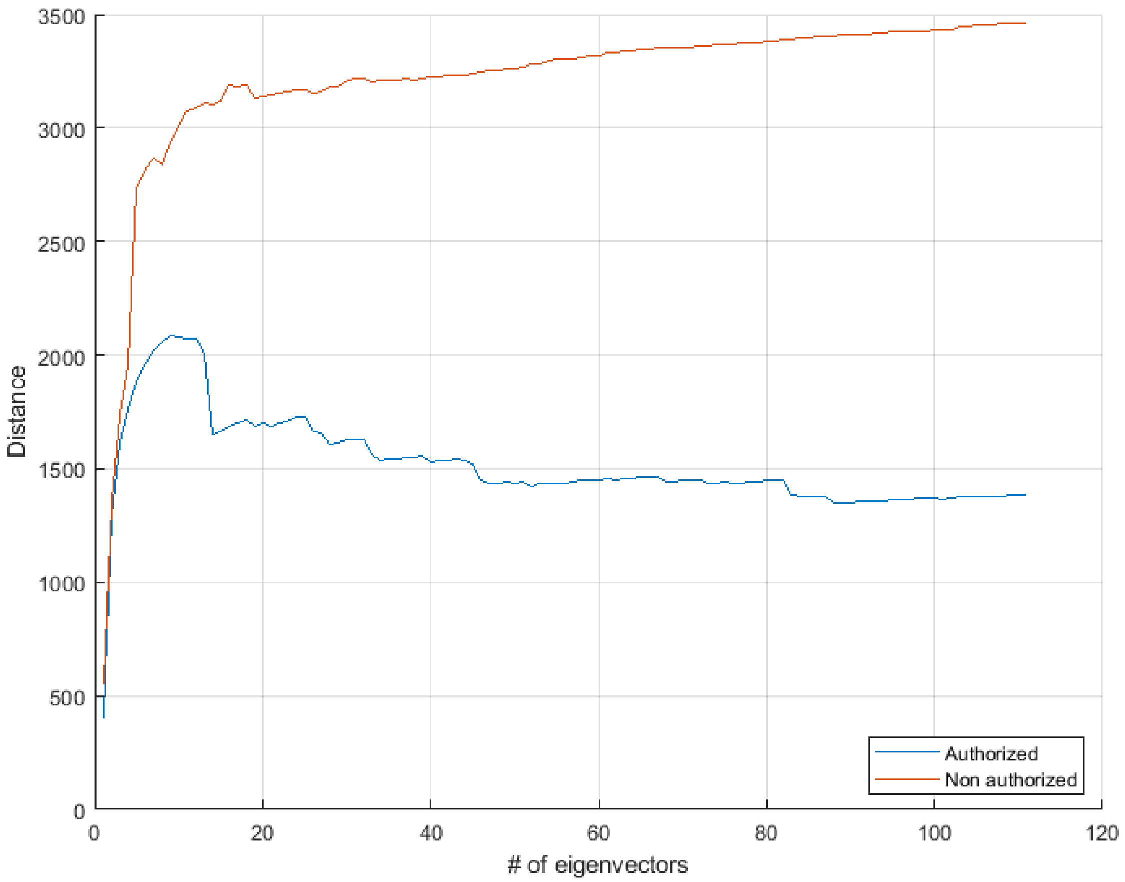

Figure 6.10 Test samples mean distance from training set samples

Anyway increasing the number of eigenvectors introduces overfitting, which can be deducted by looking at the eigenvectors images in Figure 6.11, or eigenfaces: while largest eigenvectors describes significant facial features, smaller ones shows only fine details or even noise, which generally speaking could contribute negatively to the ability of generalize to previously unseen samples. Anyway in this specific application environment this is not necessarily an inconvenient, as the number of small eigenvectors or the amount of subject specific details to consider can be seen as a tuning parameter between the generalization capacity of Eigenfaces and a more tailor-made instance of the discriminator.

It is also worth to note from Figure 6.11 that Eigenfaces seems to deviate from uniform grey where some facial feature differs among the set of training faces, as a sort of map of variations between faces.

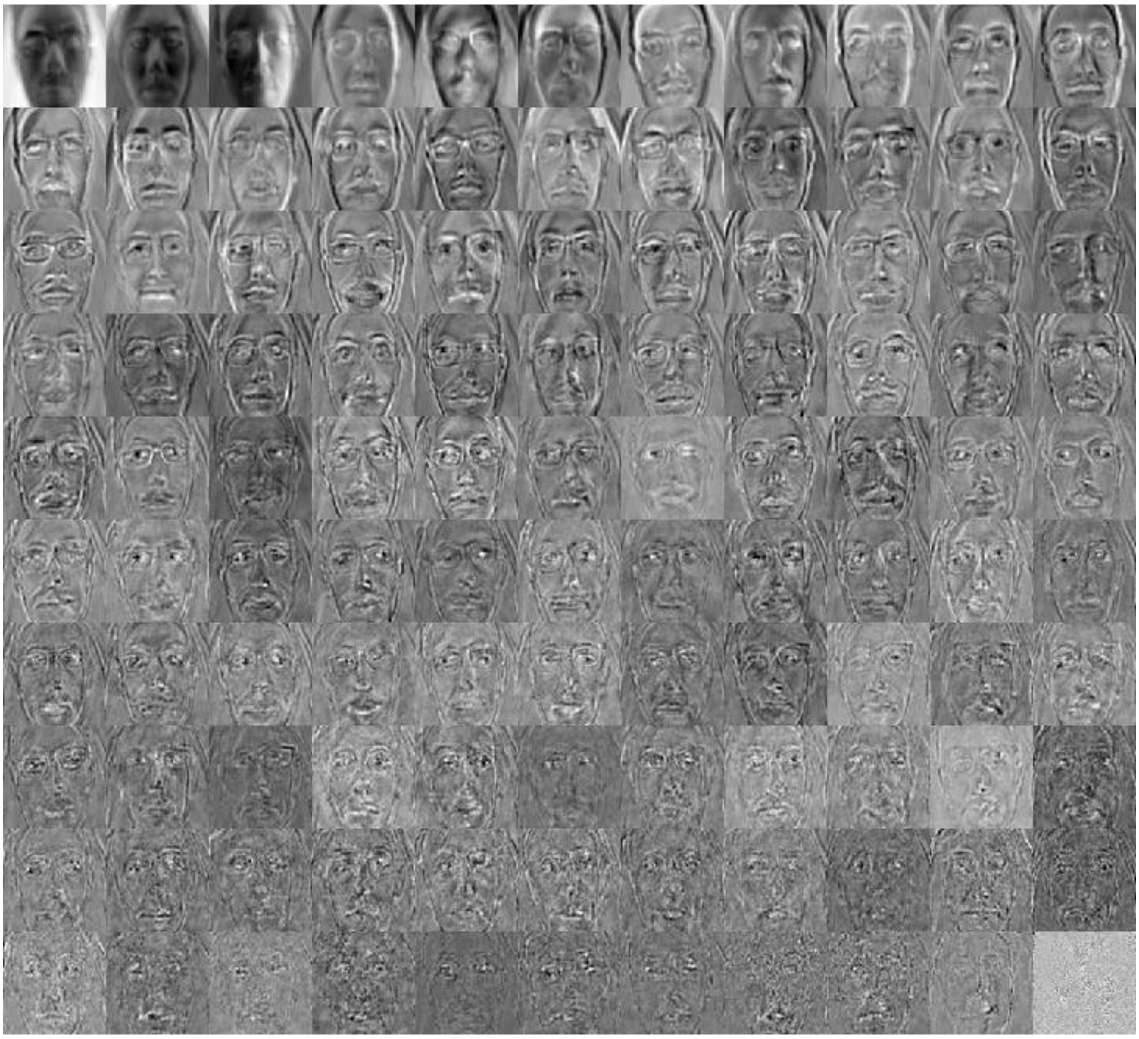

Figure 6.11 All the Eigenfaces calculated from the training set samples

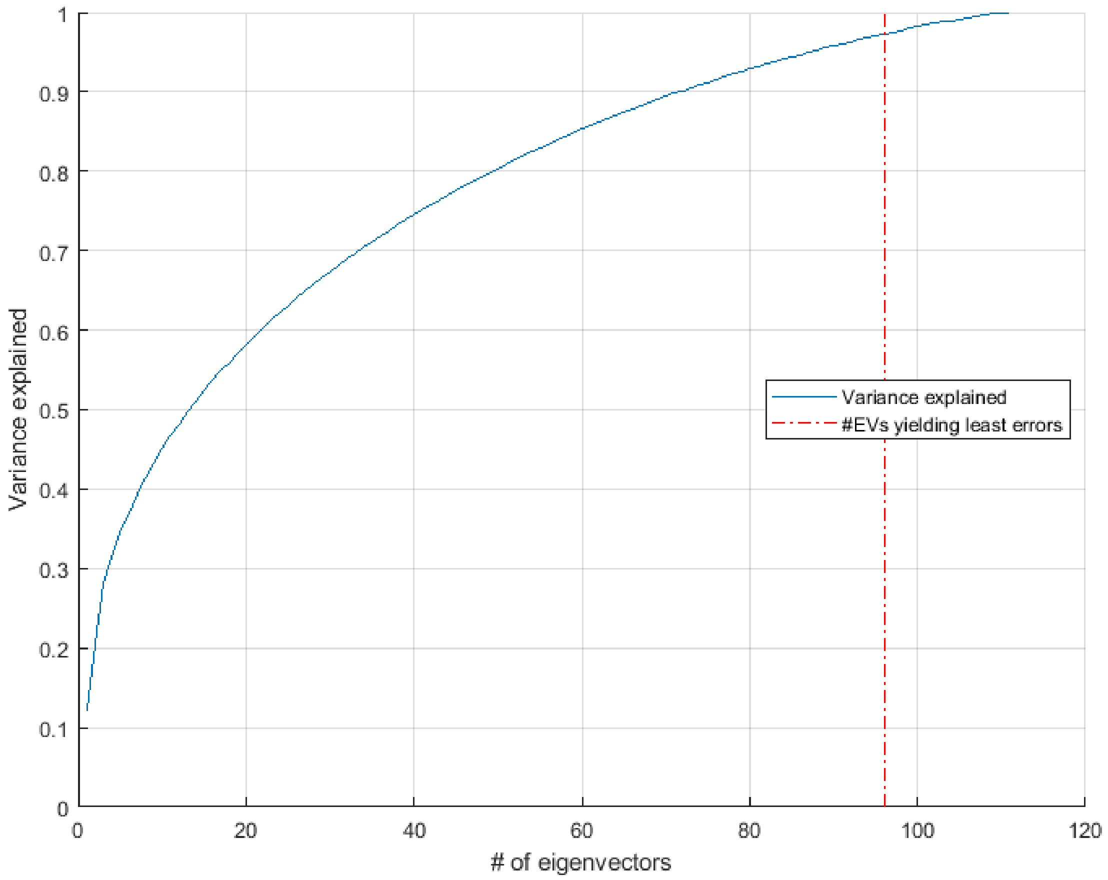

Figure 6.12 Variance explained by eigenvectors

Beside this specific application considerations, a reasonable way of searching for the optimal number of eigenvectors to describe the face space is to look for the location along the decreasing eigenspectrum, where the eigenvalues drop significantly. By looking at Figure 6.12 more than half of the training set variance is explained by the first 15 eigenvectors, while the following convey less and less variance. Their contribution is anyway remarkable with respect to the system performances also considering the expected limited impact on the system training time as shown in Figure 6.13, as the best recognition performances are achieved using around 90% (or more) of the available eigenvectors or training set.

It is also evident from Figure 6.13 that roughly from 95% to 75% of the face recognition execution time, depending on the amount of eigenvectors used, is expected to be spent in the auxiliary tasks of the training phase, such as accessing to the video stream to collect frames, or save them in image files. Increasing the number of eigenvectors has moderate effect on the system training speed, and negligible impact on the testing phase of a subject. The Viola-Jones detection time is not shown in the Figure 6.13 for scaling purposes, moreover because the typical training set for this application is expected to be much smaller than the one considered in this paragraph and it could be misleading to consider it in the below graph. The amount of time estimated for the calculations can also be used as a starting point for the claimant company, in terms of evaluating the requirements for next generation hardware design. Since the integration on the final device is not yet developed at the time of writing, the execution times were simulated on another machine with plotted values properly scaled to reproduce the internal video receiver speed. The scaling factor has been calculated by evaluating the performances of both machines CPUs in matrix multiplication operations.

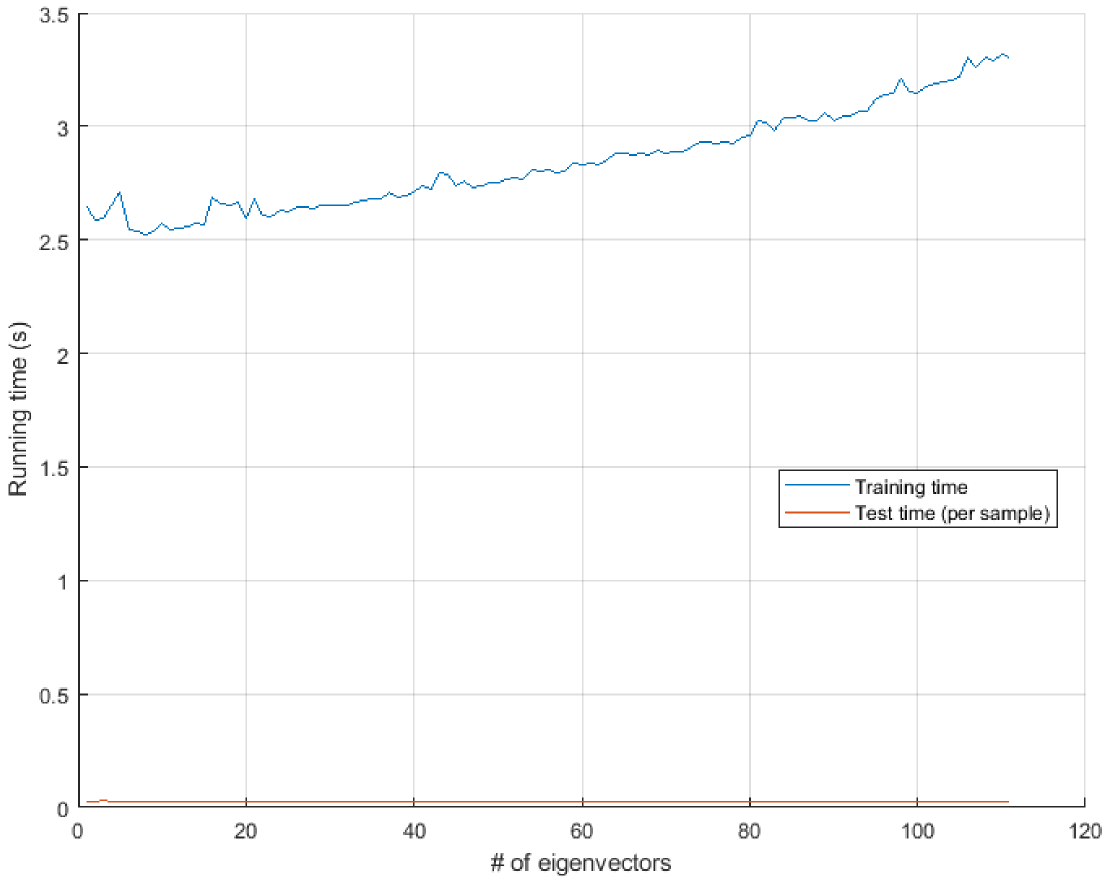

Figure 6.13 Execution time simulation

Figures 6.14 and 6.15 shows from two different perspectives how increasing the number of eigenvectors to describe the face subspace improves the ability of the algorithm to separate classes, although smaller eigenvectors don't contribute much to the recognition. In Figure 6.14 the minimum distance from an authorized subject training sample for each test sample is plotted, having on the left of the dashed line the authorized subject partion of the test set.

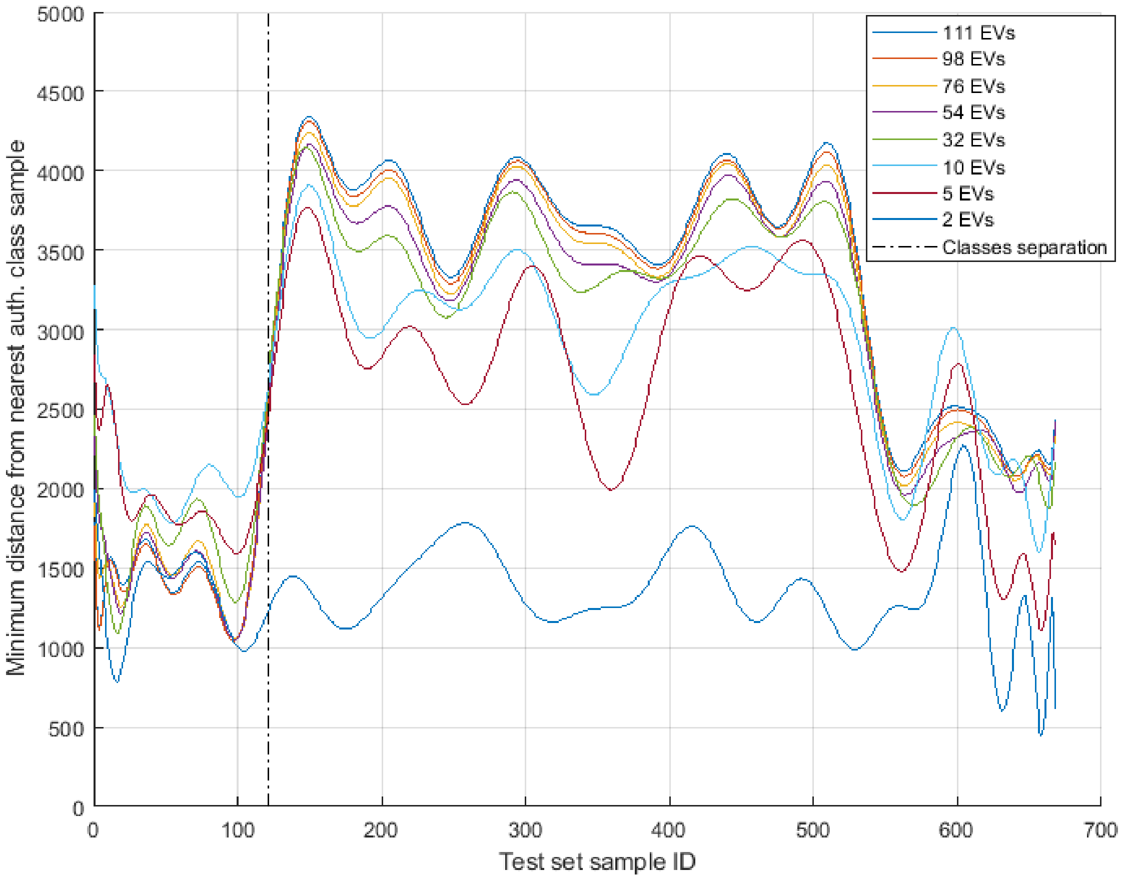

Figure 6.14 Distances trends for test set samples

In Figure 6.15 the worst samples of the two test set partitions are shown, with the largest “margin” between them corresponding to the highest amount of eigenvectors, but with an absolute value in term of errors almost identical to the one obtained with the amount of eigenvectors yielding the least errors, as was show in Figure 6.7, and by dashed red line in Figure 6.12.

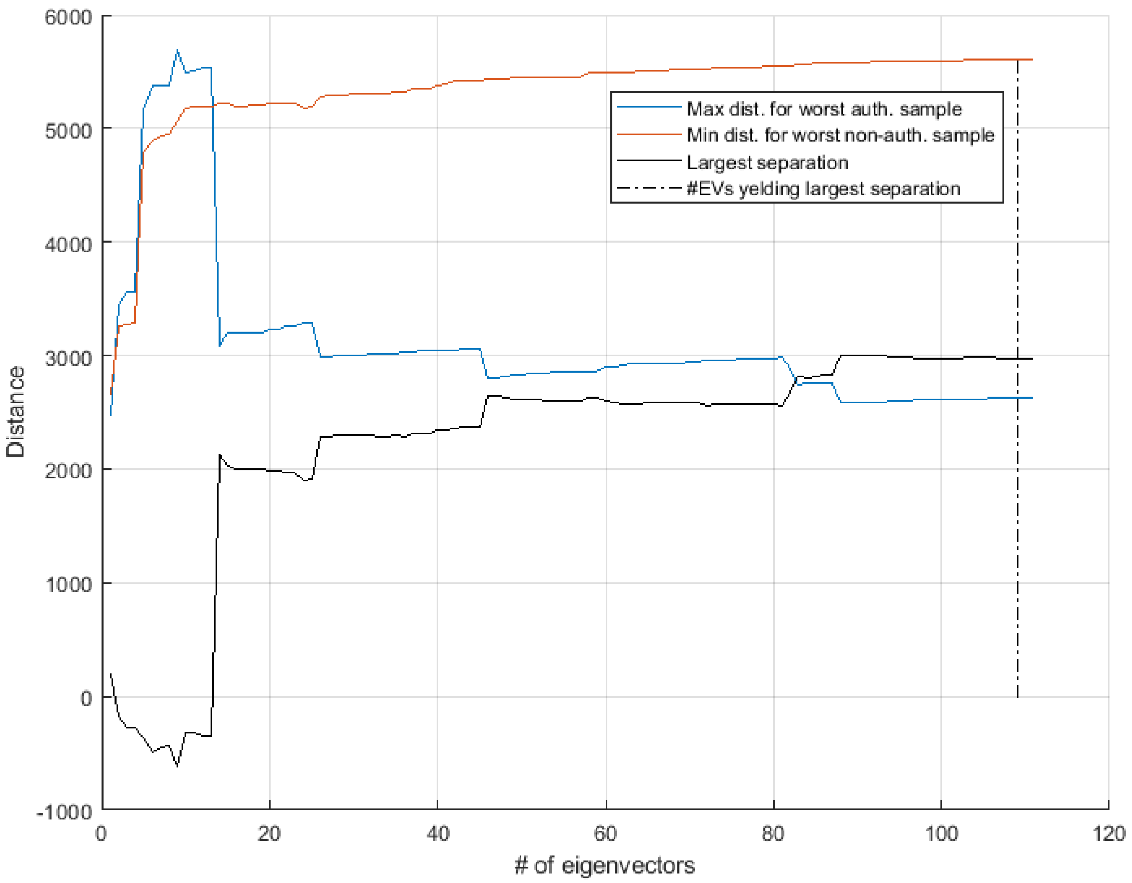

Figure 6.15 Distances between classes worst samples

7. Conclusion and future work

It has been shown through a proof of concept that it is possible to build a system capable of identify and authorize a group of subjects through facial recognition, running on embedded hardware with good performances in terms of precision, simulated execution time and speed under variable but controlled light conditions. Moreover the proposed solution answered to all of the technical requirements and constraints requested by the claimant company, as well as the majority of the other demands this analysis is meant to answer to.

The Viola-Jones and Eigenfaces algorithms were chosen as the facial detection and recognition methods, being compliant with the constraints posed by the application case, namely a reduced size training set, a lightweight calculation process and the capability to run on low power hardware. Some of the known Eigenfaces drawbacks have been addressed through the introduction of preliminary processing, in particular the face detection step using Viola-Jones method, while others have been prevented by the nature of the application type, like the frontal appearance of subjects, their willingness to be identified, and the reduced amount of subjects to identify due the residential type of application.

The system has been trained and tested on a dataset of 780 images, specifically generated for evaluating the video intercom application case, achieving an initial recognition performance of around 75%. This score has been enhanced by adding two specific improvements for low light images, namely contrast enhancement to emphasize the facial traits, and a dedicated threshold to achieve correct classification within this subclass. The final resulting pipeline is able to discriminate successfully more than 99% of the test set composed of 668 samples, where recognition errors belongs to the false negative detections type only, which compared to false positive errors is a more acceptable issue given the project environment.

There are several situations and improvements to consider for improving the proposed system and extend its suitability to a wider applications and conditions range, for example using larger datasets and cross-validation techniques to obtain a better estimate of the system performances, or extending the authentication procedure to manage the case when more than one subject is present in front of the camera. A further analysis to consider is the impact on recognition performances when complexities like occlusion are extensively present (for example due to clothing) or when a larger group of authorized subject has to be recognised.

An improved and uniform illumination can better address the low light issues of the chosen methods, especially Eigenfaces, by acting at its root cause, for example by providing the external panel with a variable power light source, regulated accordingly to the time of the day or environment light level.

Eigenfaces performances can also be improved by using a different distance concept, such as Mahalanobis distance instead of the traditional Euclidian distance, which measures the ratio between the squared distance and the corresponding variance. It has been shown by Yambor et al. ([24]) that Mahalanobis distance proves superior when 60% of the eigenvectors were used.

Experiments with more recent face recognition methods can be carried out, to determine if alternative algorithms can comply with the system constraints and improve its overall performances, or pave the way to other application cases.

Further developments of the prototype and their declinations on finished products may also help in understanding how to design a new device with native face recognition features, as for example in real world applications it is advisable to ensure the system capability of discern between real subjects presenting in from of the camera and photos of the same subjects, by introducing an identification routine or use a capture device with 3D reconstruction capabilities.

References

[1] Shih-Wei Chu, Mei-Chen Yeh, Kwang-Ting Cheng. A Real-time, Embedded Face-Annotation System. *Proceedings of the 16th International Conference on Multimedia*, 2008.

[2] Hong Zhao, Xi-Jun Liang, and Peng Yang. Research on Face Recognition Based on Embedded System. *Mathematical Problems in Engineering* 2013(4):1-6, 2013.

[3] Neel Ramakant Broker, Sonia Kuwelkar. Implementation of Embedded System Based Face Recognition System. *International Journal for Research in Applied Science and Engineering Technology* Volume 5 Issue IV, 2017.

[4] Fei Zuo, Peter H. N. de With. Consumer-oriented near real-time face recognition system variety of application environments. *IEEE Transactions on Consumer Electronics* 51(1):183 – 190, 2005.

[5] Minku Kang, Hyunjong Cho, Seungbin Moon. PCA-based face recognition in an embedded module for robot application, *International Conference on Control, Automation, and Systems*, 2009.

[6] Göksel Günlü. Embedded Face Detection and Recognition. *International Journal of Advanced Robotic Systems*, Vol. 9, 96:2012 2012.

[7] Ting Shan, Abbas Bigdeli, Brian Lovell, Shaokang Chen. Robust Face Recognition Technique for a Real-Time Embedded Face Recognition System. *Pattern Recognition Technologies and Applications: Recent Advances*, 2008.

[8] Mrutyunjaya Sahani, Chiranjiv Nanda, Abhijeet Kumar Sahu, Biswajeet Pattnaik. Web-based online embedded door access control and home security system based on face recognition. *International Conference on Circuits, Power and Computing Technologies,* 2015.

[9] R. Manjunatha, R. Nagaraja. Home Security System and Door Access Control Based on Face Recognition. *International Research Journal of Engineering and Technology*, 2017.

[10] Weihao Gao. Face Recognition Application Based On Embedded System, 2014.

[11] K. Tanveer Alam, Mahammad D.V, B. Rama Murthy, Sujay Dinakar. Design and Development of an embedded based Facial Recognition System using UDOO Android, *Journal of Electronics and Communication Engineering* Volume 10, Issue 4, Ver. II, 49-54, 2015.

[12] Abbas Bigdeli, Colin Sim, Morteza Biglari-Abhari and Brian C. Lovell. Face Detection on Embedded Systems. *Proceedings of the 3rd international conference on Embedded Software and Systems* 295–308, 2007.

[13] Kwok Ho Pun, Yiu Sang Moon, Chi Chiu Tsang, Chun Tak Chow, Siu Man Chan. A face recognition embedded system. *Proceedings of the SPIE*, Volume 5779, p. 390-397, 2005.

[14] Najwa Aaraj, Srivaths Ravi, Anand Raghunathan, and Niraj K. Jha. Architectures for Effcient Face Authentication in Embedded Systems, *Proceedings of the Design Automation & Test in Europe Conference*, 2006.

[15] Karthik Ramani, Al Davis. Application Driven Embedded System Design: A Face Recognition Case Study. *International conference on Compilers, Architecture, and Synthesis for Embedded Systems*, 2007.

[16] T.Theocharides, G. Link, N. Vijaykrishnan, M.J. Irwin. Embedded Hardware Face Detection. *IEEE International Conference on VLSI Design*, 2004.

[17] Qasim Hasan Mezher Al-shebani. Embedded door access control systems based on face recognition, 2014.

[18] Jelena Milosevic, Dexmont Pena, Andrew Forembsky, David Moloney. It All Matters: Reporting Accuracy, Inference Time and Power Consumption for Face Emotion Recognition on Embedded Systems, 2018.

[19] Paul Viola and Michael Jones. Robust Real-time Object Detection. *Second international workshop on statistical and computational theories of Vision-modelling, learning, computing, and sampling*, 2001.

[20] Yoav Freund and Robert E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. J*ournal of computer and system sciences*, 55:119-139, 1997.

[21] Woodrow Wilson Bledsoe. The model method in facial recognition. *Panoramic research Inc.*, 1966.

[22] Lawrence Sirovich and M. Kirby, Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, 1987.

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

[24] Wendy S. Yambor, Bruce A. Draper and J. Ross Beveridge. Analyzing PCA-based Face Recognition Algorithms: Eigenvector Selection and Distance Measures. *Empirical Evaluation Methods in Computer Vision 2002*, pp. 39-60, 2000.

[25] Elias Wright. The future of Facial Recognition is not fully known: developing Privacy and Security Regulatory Mechanisms for Facial Recognition in the Retail Sector. *Fordham Intellectual Property, Media and Entertainment Law Journal*, vol XXIX(2), pp. 611-685, 2019.