

Development of a real-time face recognition system for access control

Desmond E. van Wyk, James Connan

Department of Computer Science

University of the Western Cape, Private Bag X17 Bellville, 7535, South Africa

Telephone: +(27) 21 959-3010, Fax: +(27) 21 959-3006

Email: 2645028@uwc.ac.za, jconnan@uwc.ac.za

Abstract—Face recognition is an active field of research with many applications. This paper discusses the development of a face recognition system that operates in real-time. For our system we make use of Multi Layer Perceptron Neural Networks and use the backpropagation weight update rule for training. We train one network per person to perform face recognition. Evaluation of our system is performed with respect to false acceptance and false rejection errors to compute the equal error. One of our goals was to perform as little image pre-processing as possible and to still obtain a low equal error.

Index Terms—Face Recognition, Artificial Neural Networks, Access Control

I. INTRODUCTION

FACE recognition is an active field of research and has increased significantly since the early 1990s. This is mainly due to the fact that government agencies and businesses have realized the vast range of commercial applications that one can provide with face recognition. These applications fall in many areas, such as: entertainment, smart cards, information security and law enforcement [6].

Since face recognition research is so active, there are a multitude of techniques and algorithms to perform face recognition [5], [6]. A very important distinction that one should make is between three-dimensional (3D) and two-dimensional (2D) face recognition methods. In 3D face recognition a 3D model of a face is used in the recognition process. The 3D model is usually constructed from a set of 2D images or by means of a 3D camera. In 2D face recognition a 2D image is used in the recognition process. Within 2D face recognition research one must make further distinctions between still and video based face recognition methods. This is mainly due to the fact that still images of faces are captured under controlled conditions such as in a laboratory and are of much higher quality than images captured with a video camera [6].

A face recognition system is in effect a biometric system. With respect to face recognition, the face is considered to be

the biometric because it may be used to uniquely identify an individual [2]. This paper addresses the implementation of a face recognition system that may be applied to the task of access control and overlaps with the areas of smart cards, information security and law enforcement. An important question that arises is which technique will be the most suitable for video-based face recognition, since for the task of access control a camera that captures a live video stream is normally used.

This paper takes a software engineering approach to how we implemented a face recognition system. In Section II we look at the requirements for such a system and in Section III we analyse those requirements. In Section IV we do further analysis to discover users and interfaces of such a system. In Section V we discuss the high level of our system. Section VI provides the different stages of development along with implementation details. In Section VII we discuss some of the testing we performed and give our conclusions in Section VIII.

II. SYSTEM REQUIREMENTS

A. Requirements for a system

The foremost expectation for the face recognition system is that it must have a high degree of accuracy when recognizing people. The next highest expectation for the system was that people should be notified when the system recognized them. It must also be possible to easily add and remove people from the system that should be recognized. The system must be able to recognize a person under the following conditions and circumstances:

- Frontal face poses.
- Minor variations in lighting conditions.
- Minor variations in facial expressions.

B. Not required from a system

The solution system was not expected to accurately recognize a person and indicate who that person was under the following conditions and circumstances:

- Non frontal face poses.
- Extreme variations in lighting conditions such as darkness or too much light under which it will be very difficult to recognize a person.
- Major variations in facial expressions.

J. Connan is with the Department of Computer Science, University of the Western Cape, Private Bag X17 Bellville, 7535, South Africa (phone: +(27) 21 959-3010; fax: +(27) 21 959-3006 ; e-mail: jconnan@uwc.ac.za).

D. E. van Wyk is a M.Sc student in the Department of Computer Science, University of the Western Cape, Private Bag X17 Bellville, 7535, South Africa (e-mail: 2645028@uwc.ac.za).

A solution system was also not expected to be a complete access control system and only needed to recognize people and indicate if a person was recognized or not.

III. REQUIREMENTS ANALYSIS

A. Requirements interpretation

Access control systems are considered to be mission-critical real-time systems and thus must operate correctly under many different situations and circumstances. For a fully automatic face recognition system, face detection and face localization are very important and the very first steps to developing such a system [6], [7]. The single most dominant problem with a face recognition system or other biometric systems is accuracy and they do not perform well under the many different situations and circumstances that are encountered in day-to-day life [2]. Thus accuracy should be the main focus of the solution system.

Zhao et al. [6] suggest that one must evaluate the accuracy of the various techniques and algorithms depending on the task to which they are applied. With respect to access control, the accuracy of these systems can be tuned by balancing two competing types of errors: false rejection errors (FREs) and false acceptance errors (FAEs) to obtain the equal error (EE) [2], [6]. A false acceptance error occurs when a person is mistakenly positively recognized [1], [6]. A false rejection error occurs when a person that should be positively recognized is not accepted or recognized at all [1], [6]. For face recognition systems there are many contributing factors that can lead to these two errors and the most important ones are:

- Variations in lighting conditions.
- Variations in face pose.
- Variations in facial expressions.
- Total or partial occlusion of a face [1], [6], [7].

Other factors that can also significantly affect the recognition accuracy and that should also be considered are:

- Quality of face images.
- Image resolution and face area size in images.
- Color information in face images [6], [7].

Aside from the factors mentioned above that could lead to recognition errors, it is known that face recognition systems could be easily deceived by using a face mask or face image of a person that will be positively accepted. There are however methods of making a face recognition system more robust so that it is not easily deceived. One way of doing this is to make use of a face thermogram or output from an infrared camera as described in [1]. Although a face thermogram or output from an infrared camera can be used as mentioned in [1], the underlying techniques and algorithms for performing face recognition are still the same.

B. Existing solutions

Since face recognition is such a highly researched field, there are many different techniques and algorithms for performing face recognition. As mentioned before, there are many different areas in which face recognition can be applied and we would like to refer the reader to Zhao et al. [6] for a

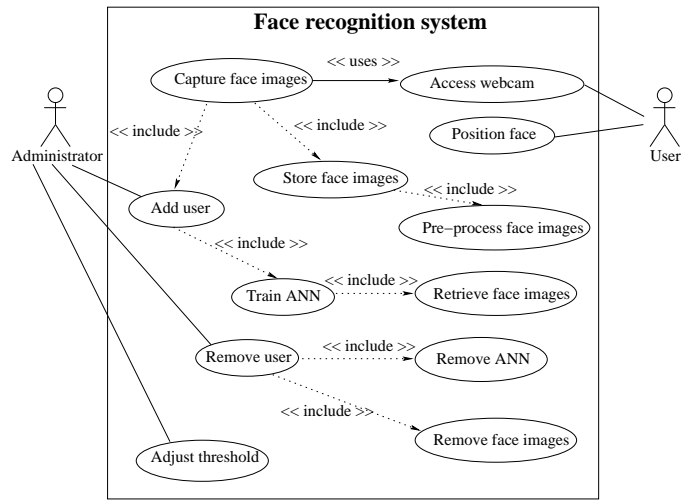


Fig. 1. User addition and removal scenario.

comprehensive survey on face recognition algorithms. Some of the techniques and algorithms discussed in [5] for performing face recognition fall under the following tasks that include:

- “document control”
- “access control”
- “database retrieval”

The research performed in [1] and [5] suggests that certain techniques and algorithms are best suited to certain tasks. For this very reason, the implemented system makes use of Artificial Neural Networks (ANNs) to perform face recognition, since they are best suited to the task of access control as discussed in [1] and [5]. This is due to the fact that for access control systems, video cameras are normally used which provide images of poor quality that contain a lot of noise [6]. Artificial Neural Networks are known to be robust to noise and thus techniques that make use of them would make the ideal choice for real world applications [1][3].

IV. FURTHER ANALYSIS AND USERS’ INTERFACE

Further analysis showed us the different types of users and ultimately gave us hints on the user interface requirements. Two use case scenarios were depicted to aid in visualizing how users interact with the system. The first scenario is called the *User addition and removal scenario* and is depicted in Fig. 1. In this scenario there are only two types of actors or users that will interact and use the system. These are mainly an Administrator and a User. The Administrator is the system user that will add and remove other Users that must be recognized by the system through a graphical user interface. The User is the system user that is to be added for recognition by the Administrator and interacts with the system through a webcam.

The second scenario is called the *User Face Recognition Scenario* and is depicted in Fig. 2. In this scenario, the system is used to recognize a User that was added, to be recognized by the system by using a webcam and a graphical user interface that contains a live video window. The window is used by the

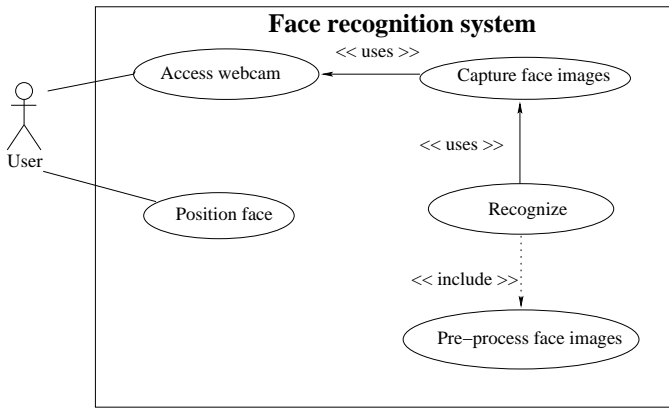


Fig. 2. User face recognition scenario.

User to help position the Users face for the images that are to be captured by the system.

V. SYSTEM DESIGN

The high level design of the system is depicted by a component diagram in Fig. 3. The system consists of both hardware and software components. The hardware components include the webcam and Personal Computer. The web cam is connected to the Personal Computer component through a universal serial bus (USB).

The main application functionality all resides within the Personal Computer component, which are all software components. The software components include the Face Recognition Application component which is dependent on the Image Capture Processing component, Artificial Neural Network component and Graphical User Interface component. The rest of the system design and implementation was broken into four stages as can be seen in Fig. 4. Each stage is discussed below with implementation details of some of the components.

VI. SYSTEM IMPLEMENTATION

A. Stage 1

At this stage of our implementation we focused on capturing images from a webcam between 25 and 30 frames per second.

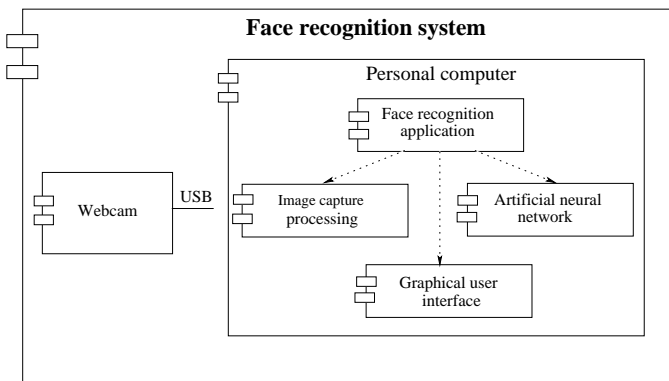


Fig. 3. The system high level design.

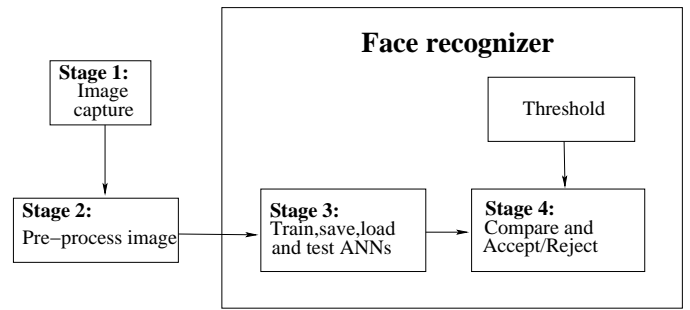


Fig. 4. The system implementation stages.

Images are captured from a raw Red Green Blue (RGB) video format that has 24 bits per pixel and dimensions of 320×240 pixels. These images are cropped in memory to have dimensions of 150×200 pixels that contains the face of a User.

B. Stage 2

At this stage we developed and applied image pre-processing operations. We first applied a grayscale transform on the images before resizing them to have dimensions of 15×20 pixels. The pre-processing is performed to reduce the amount of image data and thus ultimately the size of the ANNs used.

C. Stage 3

At this stage ANNs are trained, saved, loaded and tested to provide an output that depends on the image data that they were given. The ANNs used are known as Multi Layer Perceptron Networks (MLPs) and we use the backpropagation weight update rule for training. The ANNs we employ make use of sigmoid threshold units because they use the sigmoid activation function as illustrated in Fig. 5 [3].

The sigmoid threshold units' inputs are indicated as x_1 to x_n and the weights are indicated as w_1 to w_n . The sigmoid threshold unit includes a bias input x_0 that is always 1 and that has an associated weight w_0 [3]. The output of the sigmoid threshold unit is computed by first calculating the sum of the product of the inputs and their weights. The sum of the product of the inputs and their weights are then used in the sigmoid activation function. The sigmoid activation function is also known as a squashing function and forces the output of the unit to fall between 0 and 1 [3].

D. Stage 4

At this stage the output of an ANN is compared with an acceptance threshold value used by our system. If the output of an ANN is less than the threshold value, then the User is rejected otherwise the User is accepted [1][3]. The next section provides details of the testing that we performed.

VII. TESTING AND RESULTS

A. Performance testing

We tested the components during the various implementation stages of our system to ensure that they operate correctly

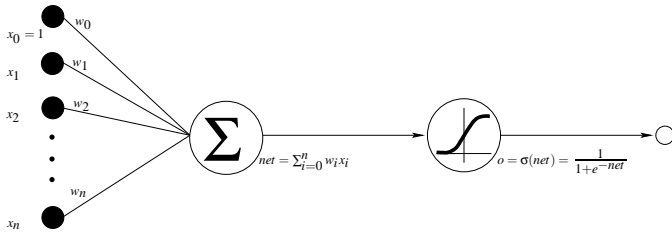


Fig. 5. The sigmoid threshold unit taken from [3].

and in real-time. Thus we first performed tests on the image capture and pre-processing operations. These tests showed us that we needed to keep the image processing to a minimum to ensure real-time operation. One of these tests included the resizing of an image with dimensions of 150×200 pixels to have dimensions of:

- 120×160 ,
- 60×80 ,
- 30×40 and
- 15×20 pixels respectively.

The results of the above tests in Table I led us to use images with dimensions 15×20 pixels in our system. The 15×20 images yielded not only fast ANN training and testing times but also acceptable recognition accuracy. Two other reasons for this decision was 1) we still need to add face detection and tracking which are expensive operations and 2) we make room to add additional image pre-processing operations to enhance facial images before recognition [6].

B. Correctness testing

The most important testing with respect to the correct operation of our system was that of face recognition. To ensure that our system could in fact recognize faces by training and

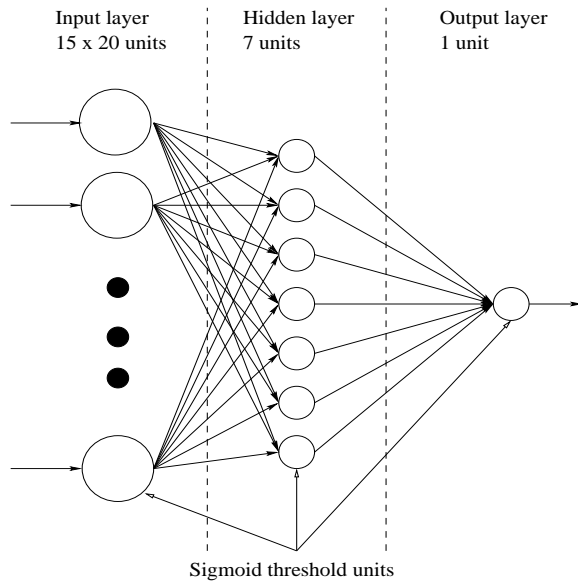


Fig. 6. The ANN architecture with sigmoid threshold units, adapted from [1] and [3].

TABLE I
RESULTS FOR IMAGE RESIZING OPERATIONS

Resize dimension	Time (ms)
120×160	11.433
60×80	1.695
30×40	0.135
15×20	0.043

testing ANNs, we performed offline testing by using a face database. The face database we used is the same one used in [1] and is known as the Olivetti Research Ltd (ORL) database [4]. The ORL database has a total of 40 subjects denoted by s1 to s40 with 10 facial images per subject. For a detailed description of the ORL database we refer the reader to [5].

We performed many tests of which we include the description and results of only two tests. The evaluation criteria we used for these tests, was the computation of the FAEs, FREs and EE since our goal was to develop the system to perform access control.

The results of the first test, which we will refer to as T1, are displayed in Fig. 7. We trained an ANN only for subject s1 in the ORL database. Only the first five images of subject s1 were used as positive training samples to train an ANN and the latter five were used for testing the trained ANN. The first images of subject s31 through s40 were used as negative training samples. Subject s11 through s30 of the ORL database were used as unseen or alien people upon testing the ANN for subject s1. The percentage of False Acceptance Errors (%FAE) and percentage False Rejection Errors (%FRE) were computed by varying the acceptance threshold used when testing the ANN. There are various parameters that can affect the result of the test such as the image dimensions used and the parameters when training the ANN. The images of the ORL face database were resized from dimensions of 92×112 pixels to 15×20 pixels in memory to stay consistent with the rest of our system design decisions. The ANN that was trained for subject s1 has the following parameters:

- Learning rate = 0.35
- Number of input units = 15×20
- Number of hidden units = 7
- Number of output units = 1
- Number of weight update cycles = 500

It should be noted that most of these parameters were obtained by experimentation. The EE is the point where the %FAE equals the %FRE. As can be seen from the results in Fig. 7, the EE equals 0% where the acceptance threshold is 0.8.

The results of the second test, which we will refer to as T10, are displayed in Fig. 8. We trained ANNs only for subject s1 through s10 in the ORL database. Only the first five images of each subject were used as positive training samples to train the ANNs and the latter five were used for testing the trained ANNs. The first images of subject s31 through s40 were used as negative training samples. Subject s11 through s30 of the ORL database were used as unseen or alien people upon testing the ANNs. We used the same parameters for the

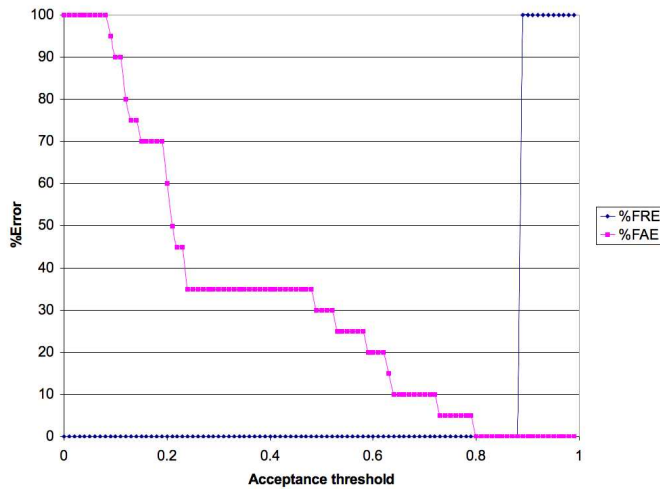


Fig. 7. Result for T1.

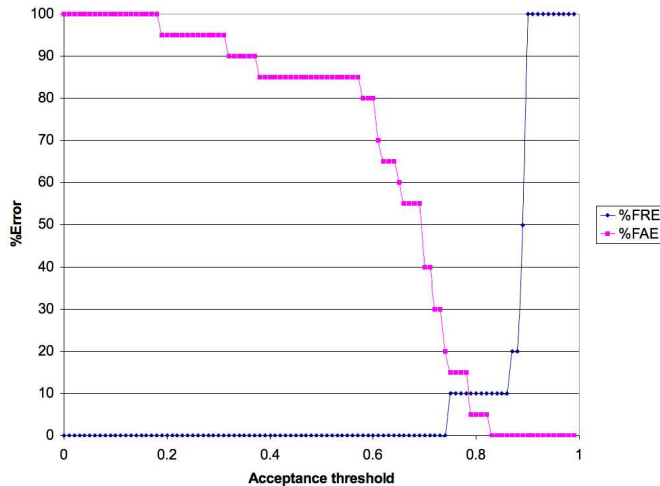


Fig. 8. Result for T10.

ANNs as for the ANN in T1. The EE for T10, as can be seen in Fig. 8, was found to be 10% where the acceptance threshold is 0.79. The results obtained from tests such as the ones described above suggest that the technique we employ is highly usable for an end system. We do however recognize that more testing and experimentation should be performed to guide us in selecting the appropriate parameters for both the image dimensions we use and for the ANNs.

VIII. CONCLUSION

In this paper we discussed the development of a face recognition system that may be applied to the task of access control. We noted that face recognition systems for access control normally use video cameras that deliver image data of poor quality due to noise. The technique we implemented use ANNs which are robust to such noise. Much of the focus for the implemented system was also on its real-time behaviour while performing continuous face recognition. This

in effect has shown us how scalable the system is with respect to the number of users and the ANNs used in the face recognition process. A major problem for the above mentioned technique was finding parameters to train the ANNs. Thus more experimentation could be performed not only to acquire better parameters to train ANNs but also their architecture. The system can easily be extended to include face detection and tracking and to avoid continuous face recognition. The goal for an access control system that use face recognition should be a %FAE of 0%. This implies that one must be strict when selecting an acceptance threshold and how Users interact with the system when they are added to be recognized.

ACKNOWLEDGMENT

We wish to thank Mr. R. Dodds and the many anonymous reviewers for their suggestions and criticism that helped us to improve this paper.

REFERENCES

- [1] Bryliuk, D. and V. Starovoitov. 2002. Access control by face recognition using neural networks and negative examples. The 2nd International Conference on Artificial Intelligence, pp. 428–436.
- [2] Libin, P. 2005. (2006, March). A practical summary of the advantages and drawbacks of today's biometric systems for mainstream customers. [Online]. Available: <http://www.assaablofuturelab.com/399.epibrw>
- [3] Mitchell, T.M. 1997. Machine Learning. McGraw-Hill.
- [4] Samaria, F. and A. Harter, Parameterisation of a stochastic model for human face identification 2nd IEEE Workshop on Applications of Computer Vision December 1994, Sarasota (Florida).
- [5] Starovoitov, V.V., D.I. Samal and D.V. Briliuk. 2002. Three approaches for face recognition. The 6th International Conference on Pattern Recognition and Image Analysis, pp. 707–711.
- [6] Zhao, W., R. Chellapa, P. J. Phillips and A. Rosenfeld. 2003. Face Recognition: A Literature Survey, ACM Computing Surveys, 35(4):399–458.
- [7] Yang, M., D.J Kriegman and N. Ahuja. 2002. Detecting Faces in Images: A Survey. IEEE Trans. Pattern Analysis and Machine Intelligence, 24(1):34–58.



Desmond van Wyk is currently an Telkom Centre of Excellence M.Sc student at the University of the Western Cape. Currently he is doing research on sign language synthesis and novel communication applications for the Deaf and hard of hearing.



James Connan heads up the South African Sign Language (SASL) research group. He has a wide range of interests that include: databases, computer vision and machine learning.