BIOMETRIC AUTHENTICATION SYSTEM USING MULTI-AGENT TECHNOLOGY IN BORDER CONTROL

Susara S Thenuwara

179418B

Degree of Master of Science in Artificial Intelligence

Department of Computational Mathematics

University of Moratuwa

Sri Lanka

January 2020

BIOMETRIC AUTHENTICATION SYSTEM USING MULTI-AGENT TECHNOLOGY IN BORDER CONTROL

Susara S Thenuwara

179418B

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Artificial Intelligence

Department of Computational Mathematics

University of Moratuwa

Sri Lanka

January 2020

Declaration

I declare that this is my own work and this thesis does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to the University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or another medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Name of Student

Susara S Thenuwara

Signature of Student: S.S. Date: 28/01/2020

The above candidate has carried out research for the Master's Dissertation under my supervision.

Name of Supervisor

Prof. A.S. Karunananda

Signature of Supervisor: Asoka Date: 28/01/20

Acknowledgments

I would like to express my sincere gratitude to my supervisor Prof. A.S. Karunananda for providing his invaluable guidance, comments, and feedback throughout the entire project. I would also like to acknowledge him for constantly motivating me to work harder to make this project a success.

Furthermore, I would like to thank my parents, fellow colleagues, and all the lecturers of the Department of Computational Mathematics for the help and support they have given me over the course of this project.

Abstract

In today's world security is the most important aspect. Border criminals, frauds, unauthorized immigrants are burning issues in Sri Lanka within the last few years due to the lack of proper identification system i.e. duplicate passports, fake identity, etc. Therefore, the efficiency and accuracy of the traditional authentication system are not good enough to overcome this disaster. Furthermore, cryptanalysis and brute force attacks are dramatically strong with uncontrolled demanding of computational power. In fact, the efficiency and accuracy of the authentication are not enough to cater the future authentication systems by comparing traditional user authentication techniques. Therefore, Biometrics is the ideal solution for authentication as it has advantages over conventional systems. Second, it's not important to recall a biometric and it can't easily be lost. It makes the client much smoother. In addition, it is not easy to stolen or loan a biometric to a relative.

Biometrics can provide greater security and comfort than traditional methods of human identification. Even if we don't want to replace a conventional method with a biometric one, we are certainly future consumers of these systems, which will even be mandatory for new passport models. Therefore, it is important to be familiar with biometric security engineering possibilities. The most common way for people to perform biometric authentication maybe facial recognition. Face recognition can be based on single images, multiple still images or video sequences. Although most of the efforts have been traditionally dedicated to the former, the latest is rising, probably due to the lowering of prices in devices for image and video acquisition. The system was designed using the technology of multi-agents and it has two phases biometric capturing phase at VISA granting process and biometric recognition phase at border points. The product's longevity is maintained through the use of biometric and nonbiometric techniques. This solution has been implemented as a multi-agent program with a hybrid approach that identification of faces and validation of fingerprint. The system has been tested in a border protection environment which is a more time-critical real-world application and notices the 87% accuracy in the recognition phase. The proposed system evaluated with traditional border management software and time calculated for each participant. Those participants have given informed consent participation at the evaluation.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Prolegomena

The term “biometrics” is derived from the Greek words, “bio” is life and “metrics” is to measure and it is a gift from nature to human been [1]. Due to significant advances in computer security, biometric authentication systems have only become available in the last few decades. Biometrics is the science of identifying or verifying based on physiological or behavioral characteristics of the human being. Physiological characteristics include DNA, retinal pattern, facial appearance, fingerprints. Behavioral characteristics are actions carried out by a person in a unique way such as signatures, voiceprints, and gait. The face is one of the oldest and most important representations of a character used for human identification. Human beings have used faces since the beginning of civilization to distinguish recognized (familiar) and unknown (unfamiliar) people. Face appearance is biometric which is used every day by everyone as a primary means of recognizing other humans.

During the last few years, unauthorized immigration and emigration dramatically increased. Consequently, drug frauds, crimes are on the rise in an uncontrollable level. Hence, these threats have seriously affected many key sectors in the country i.e.: national security, finance, tourism. On the other hand, the Ministry of foreign affairs in Sri Lanka stated: “Illegal immigrants are burning issue in Sri Lanka” [2]. The main reason for the issue can be identified as a lack of proper authentication or identification system. Unauthorized VISAs, fake and duplicate identities were dramatically increased in an uncontrollable way. Normally biometric capture at the VISA granting process and the same applicant should be present at the airport with his/her passport during the time of arrival and departure [3]. Efficiency and the usage of those techniques cannot be satisfied in the Sri Lankan context. There are numerous solutions for authentication for border control authentication has been crucial for national security. Many countries' large volume of research was conducted to issues of this has resulted in numerous researches related to border control authentication due to very nature, border control authentication is done by more than one authentication method where different officers are involved. If we consider financial sector, several banks have already utilized biometric technology Wells Fargo BBVA, Compass HSBC USA Citigroup and other international banks as well biometric technology is changing the financial sector and improving the way that we conduct our financial activities if your financial institution hasn't looked at biometric technology then now might be the time to make a suggestion for improvement. This thesis has research into the face biometric authentication solution in MAS. This solution has been tested in a real-world setup and observed a very encouraging result. This area has been used in many domains including border control, VISA process, and criminal identification. However, authentication entirely through biometric measures is not adequately accurate in many instances and multi-model biometrics is not effective for a time-critical environment. The iris pattern recognition and DNA are cost and time-consuming biometrics with comparing others. Table 1.1 shows the difference between face recognition and faces biometric recognition systems.

Table 1.1 : The Difference Between Face Recognition and Face Biometric

The classical face recognition system	Face biometric recognition system
The location of capturing and recognition is not important.	The special counter/booth for capturing and recognition
Huge noise and different lighting condition in capturing and recognition phases	Less noise and similar lighting condition in capturing and recognition phases
Multiple faces in image many	Only a single face image at a time
More training images	Less training images
Facial expressions, angles are obstacles in the recognition phase	Fewer obstacles in the recognition phase
Find the identity from a large population	Find identity from a small population
A lot of applications are based on internet	Isolated application due to security reason

Source: https://www.sciencedirect.com/topics/engineering/biometric

1.2. Aims and Objectives

The aim of this project is to develop a multi-agent system for face biometric authentication in the time-critical application. To achieve this. aim the following objectives have been identified.

The objectives of this project are listed below in their chronological order.

- The critical study of Multi-Agent Systems and how it has been used in different real-time applications.
- The critical study of the literature on how to use a Multi-Agent based approach to solve the face biometric authentication problem.
- Development of a fully functional solution using Multi-Agent technology as the core of the system.
- To improve the robustness of the existing authentication system at the border control.
- Identify the duplicates, illegal entries, fake identifications in the current system.
- Writing a research paper and a conference paper on the project.
- Producing the final documentation.

This thesis has research into the face biometric authentication solution together with MAS. This solution has been tested in a real-world setup and observed a very encouraging result. Furthermore, the system has been evaluated and analyzed with the traditional system.

1.3. Background and Motivation

The surge in biometrics adoption generates a booming industry for biometric manufacturers, with market size figures expected to rise from $10.74 billion in 2015 to $32.73 billion by 2022. The estimated double-digit growth is due to growing concerns about safe authentication and penetration into a wide range of industries, including healthcare, financial services, and travel. Biometrics uses are extensive and benefit not only business organizations but also government agencies and service providers. For identification or monitoring purposes, these organizations shift to biometrics. Hong Kong and India, for example, are using fingerprints for border control and welfare programs. Walt Disney World Florida theme park takes full advantage of biometrics, using mobile technology and fingerprinting to create a magical environment for its visitors.

It is much easier to use a fingerprint or iris scan than a password, particularly a long one. Recognizing a fingerprint and allowing a user to access the device requires only a second (if so) for the most popular smartphones. Ultrasound scanners will soon become popular as they can be placed directly behind the screen by manufacturers without requiring any additional real estate on mobile. In this research author's main focus of higher accuracy and cost-effective biometrics such as face image and fingerprint.

1.4. Problem in Brief

As identified in the literature review, accuracy and robustness of the traditional face biometric system and the manual identification is not up to standard for authenticating at the border control like time-critical applications. On the other hand, we cannot rely on the face image embedded in the passport to verify the identity. Normally face biometric captured by the VISA granting process and it will be updated frequently. In such a situation there are no robustness systems to compare a person who presents at the border and the updated biometrics. This issue arises in both immigration and emigration.

Following mentioned are the research challenges that will be considered in this research:

1.4.1 Functional Issues

1) Limited data issues

There are only a few face images are captured in each individual at the biometric acquiring phase. Therefore, these less training data is not applicable to machine learning systems in time-critical applications.

2) Rely on paper-based identity

Passport identity is not up to date by comparing biometric and duplicate and fake passports are big trouble in the border control process.

3) Complexity

There are different face biometric algorithms in the conventional system. They have their own limitations and possibilities in a different application. Due to this complexity, the accuracy and robustness of the conventional system are significantly low.

1.4.2 Technological Issues

1) Distributed

Biometrics are captured by many locations within a single tour itinerary, for example, traveling to many countries on the same journey.

1.5. Approach to the proposed System

In order to address the challenges and the opportunities mentioned in the previous section, the solution proposed in this study is to implement a Multi-Agent based system, (henceforth referred to as BMAgent system) to improve the accuracy and robustness of face biometric authentication. This BMAgent system runs in collaboration with 6 agents and the main ingredient is face-biometric. This solution can be used in time-critical applications. There are three main parts of the proposed authentication system.

a. Phase1: Biometric Detection Phase
b. Phase2: Biometric Recognition Phase

Different kinds of agents are involving in the above three phases to gain a higher accurate result. "Authenticated or not" can be seen as the output of the proposed system. After detecting the face from the given image using a face detect agent recognize agents will activate accordantly with cyclic behavior. Multi-agent coordination done within this period and choose the optimal result with the same.

Finally, the system compares with eborder software which currently uses at the airport with 2400 authentic biometric samples. Contract-net protocol used as a communication protocol for both phases for a task-sharing protocol in multi-agent systems.

1.6. Resource Requirements

The following are the resource requirements that would be necessary for continuing work on this project.

1.6.1. Computer Hardware Requirements

- PC/Laptop with Intel i5 or i7 Processor, minimum 8GB of RAM
- Printer; for documents etc.

1.6.2. Software Requirements

- Software is expected to run on platforms above Microsoft Windows 7
- Microsoft Visual Studio or NetBeans IDE/Text Editors
- Appropriate Database Servers (Microsoft SQL/MySQL/Object DBs)
- Microsoft Office; to produce documentation, presentations, and charts, etc.

1.6.3. Other Requirements

- Comprehensive functional knowledge related to Constraint-Based authorization in biometrics.
- LogiTech HD Web cam used for face biometric capture, this web camera output high definition 1700*600 dpi images 12fps rate.
- Arduino Optical finger print scanner used for fingerprint biometrics with 5600 baud rates.

It should be mentioned that the above-mentioned resources and any additional resources can be acquired by the Author himself while expecting the required supervision and guidance from the supervisor and the department in general.

1.7. Structure of the Thesis

The structure of the thesis is as follows: Initially, the background and the proposed solution of the study will be briefly discussed in the Introduction chapter. Afterward, the literature review chapter will discuss the history, the state of the art as well as the future trends of Multi-Agent Systems, and its application on dynamic constraint-based face biometric. The technology chapter will elaborate on the different technologies identified and utilized in the proposed system including three phases and the Approach chapter will provide an overall image of the proposed solution for the dynamic constraint-based authorization system.

The Design and Implementation chapters will go into further details on how the system has been designed and implemented using the proper architecture and technologies according to the approach discussed in the Approach chapter. Furthermore, the Evaluation chapter will elaborate on the performance of the system compared to a dataset that has been acquired from a conventional authentication system that uses both manual and dynamic processes for real-time authentication and the Conclusion chapter provides the final verdict on the success of the project and provides its advantages as well as the technical issues and opportunities.

1.8. Summary

In this chapter, the research study that is discussed in this report has been introduced by illustrating its background and motivation, the problem that is addressed in brief, the proposed solution and the resources that are necessary for the completion of this study. Finally, the overall structure of the thesis is also discussed. In the next chapter, the literature associated with the Multi-Agent technology and Multi-Agent based face-authentication systems will be discussed and subsequently, the identified functional issues and technologies will discourse.

Chapter 2

Biometric Review

2.1. Introduction

This chapter will focus on the literature associated with Multi-Agent technology and how it has been used to solve authentication problems in different domains. Initially, a brief history of Multi-Agent technology, which is the main technology used in the BMAgent system, will be discussed. Afterward, a critical review of various studies that have been done with regards to face biometric using Multi-Agent technology in various domains will be discussed. Many researchers have been conducted on face recognition or authentication in still face images and one algorithm or comparing with the latest algorithms. They have used popular techniques like principal component analysis (PCA), linear discriminate analysis (LDA) and backpropagation neural network which can be considered as higher accurate in face biometric domain. They use those technologies in different applications. In the following section, the authors review the usage of the above-mentioned techniques and how their contribution and limitation affect the final solution. Finally, the functional issues, as well as the technical issues and the opportunities that will be identified and addressed in this study, will be discussed.

2.2. Application of Multi-Agent Technology in biometric authentication

Biometric software implementations in recent years are not limited to high-security border control and national security situations but are also limited to day-to-day applications for civil and eCommerce. Many specific biometric attributes, such as fingerprint, eyes, tone, gait, retina, iris, hand morphology and vein patterns, are available for personal identification. Nevertheless, recognition based on any of these approaches may not be sufficiently robust or may not be applicable to a specific user group in a particular situation or instance. They can be discussed in two different categories, functional and technological when it comes to the main identified issues related to this research. The operational aspect addresses process-related issues based on authorization and the technical dimension focuses on perceived opportunities and problems related to the Multi-Agent approach. In the past, there is no research undertaken for multiple biometrics using multi-agent technology however there are many types of research on face recognition using multi-agent technology. Agent-based biometric systems use the computational notion of intelligent autonomous agents that assist the users and act on their behalf to develop systems that intelligently facilitate biometrics-enabled transactions, giving them the ability to learn from the users and adapt to application needs, thus enhancing recognition performance and usability.

2.2.1 Face recognition using multi-agent technology

Rajiv Gupta discussed the face recognition system using multi-agent technology [4]. Multi-agent-based computing model the use of intelligent agent methodology enables efficient control of the challenge posed by remote access facial biometrics. Intelligent autonomous agents and multi-agent systems represent a vibrant and rapidly expanding field of research. Innovative multi-agent architecture is used in complex environments to solve the issue of distributed face recognition. The architecture is based on a multi-agent system framework proposed for the distributed face recognition tasks by Intelligent Systems Group at the University of Canberra. The architecture consists of a combination of multi-layered structural and functional models in a distributed network-oriented environment. The structural and functional multi-layer model can be interpreted at the structural level in two ways, one at the functional or operational level, as shown in figure 2.1. In addition, the different stages of face recognition for distributed implementations, such as the acquisition stage for capturing face biometric information, the feature extraction stage and the classification stage, are distributed spatially and functionally. The facial recognition systems deployed in such distributed environments need an adaptive and versatile configuration of the system, for which an approach based on an advanced multi-agent model can be very promising.

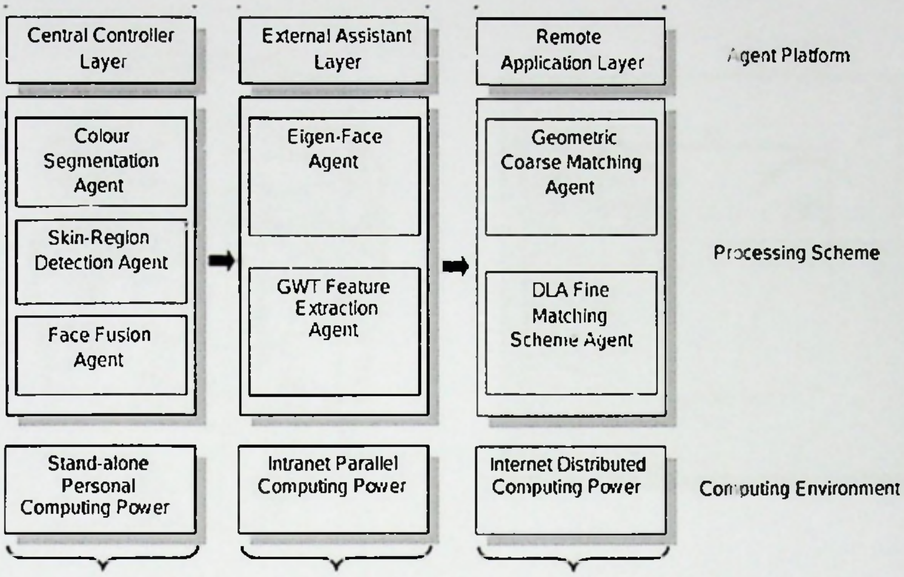

Central Controller Layer
External Assistant Layer
Remote Application Layer
Agent Platform
Colour Segmentation Agent
Skin-Region Detection Agent
Face Fusion Agent
Eigen-Face Agent
GWT Feature Extraction Agent
Geometric Coarse Matching Agent
DLA Fine Matching Scheme Agent
Processing Scheme
Stand-alone Personal Computing Power
Intranet Parallel Computing Power
Internet Distributed Computing Power
Computing Environment

Figure 2.1 : Multi agent-based face recognition architecture

Source: https://www.researchgate.net/publication/263656998_Multi-Agent_ Approach_ towards_ Face_ Recognition_ System

Facial recognition technology with a multi-agent approach offers practically feasible strategies to address efficiency and consumer acceptance barriers to widespread biometric systems adoption. A great deal of work is being conducted around the world to improve the accuracy and capabilities of this biometric domain and its use will be extended in the near future.

Hicham Hatimi and co-workers have presented face recognition using a fuzzy approach and a multi-agent system from video sequences [5]. This article proposes a multi-agent modeling method for identification and face recognition in video sequences. This method involves multiple measures to identify the identified faces in the video. The multi-agent approach adopted minimizes the complexity of the processing and achieves results with minimal time. The tasks of facial identification and classification are carried out in two stages. Faces are identified in the first step using the color of texture and the geometric form. In the second step, the multi-agent method and the fuzzy approach are used to define the degrees of membership in the recognition process. The results obtained using this approach illustrate robustness, lighting and speed variations. The parameters and characteristics of the face shown in figure 2.2.

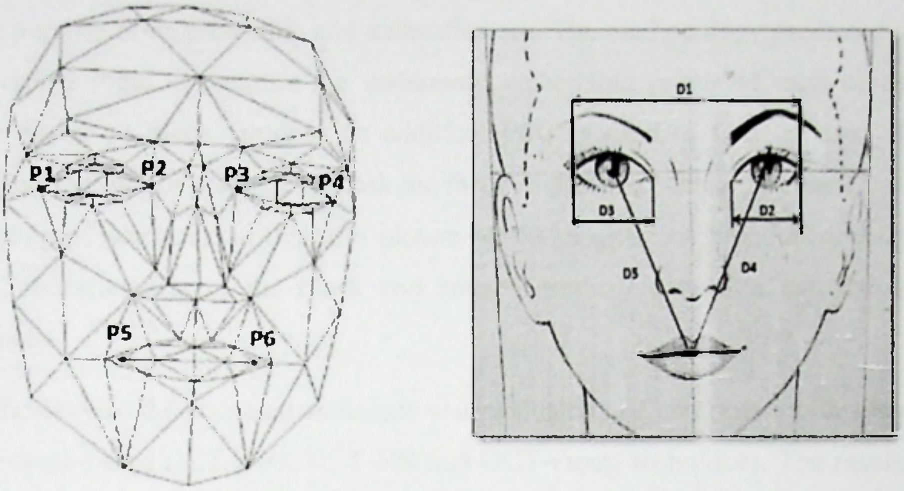


Figure 2.2 : The parameters characteristic of the face

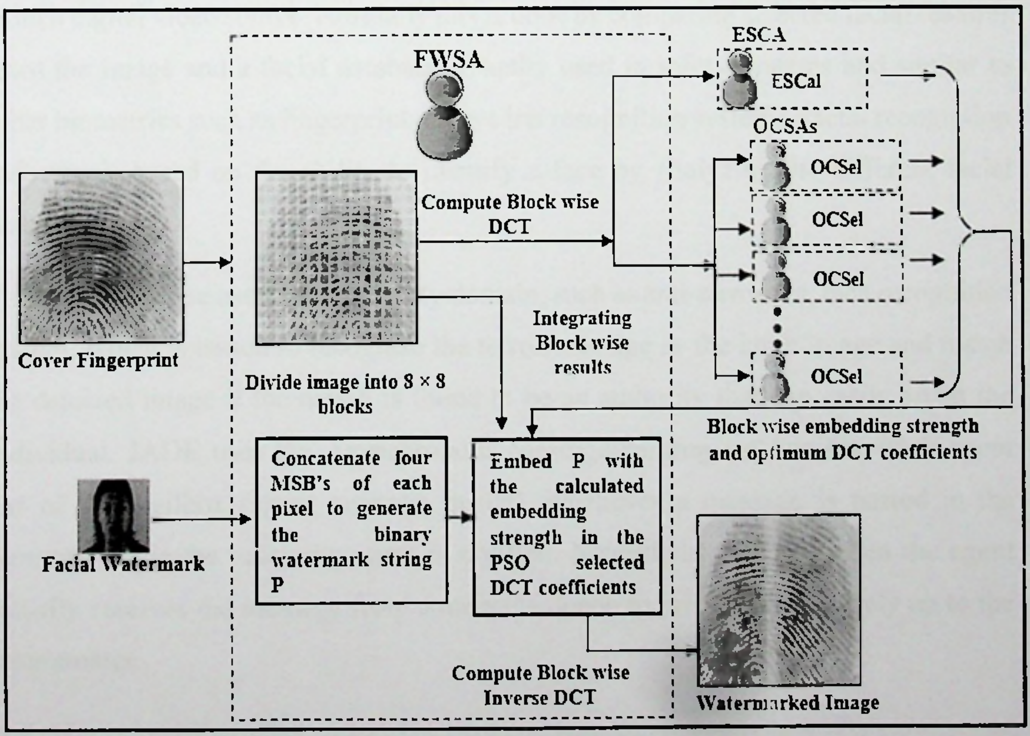
Source: https://ieeexplore.ieee.org/abstract/document/7467753

2.2.2 Fingerprint recognition using multi-agent technology

Biometric identification is becoming a topic of growing importance. Fingerprint recognition gets the most attention because of its easy use. Although there are already many fingerprint recognition systems available, more work on the subject is still needed to improve the reliability and performance of the systems. Multi-agents provide a wide array of strategies to address issues that are difficult to analytically solve. Fingerprints can be used to identify individuals because they are unique to each individual and do not alter during a person's lifetime.

Roli Bansal and co-workers have presented a multi-Agent System for Intelligent Watermarking of Fingerprint Images [6]. This paper presents an architecture of multi-agent systems for intelligent watermarking to protect fingerprints. The proposed watermarking method uses a hybrid approach based on fuzzy-PSO to protect the fingerprint image of a person by watermarking it with its corresponding face image. Since fingerprint databases are., it's difficult to process huge image data in real-time. The proposed work uses a multi-agent system as a distributed system for watermarking fingerprint images, where the distributed system performs various subtasks in parallel. This paper introduces a multi-agent structure for an effective watermarking technique using a hybrid-based fuzzy-PSO approach to cover a face image in a fingerprint picture for the purpose of its protection and authentication. The methodology presented uses fuzzy type-2 logic to measure the watermark embedding power of each block of images based on block features. In addition, PSO is used to find the best DCT coefficients in the cover image to mask the facial image pixel data. This results in the embedding of watermarks in various picture blocks of different strengths depending on the characteristics of the block and creates minimal distortion and optimum robustness.

The efficiency of the proposed technique was evaluated and contrasted with some of the literature-based DCT-PSO, DCT-NN and DCT-Fuzzy techniques. The results of the proposed technique can be seen to be stronger because of the optimum embedding power determined by type-2 fuzzy logic and the optimum coefficients chosen using PSO. Figure 2.3 shows the proposed multi-agent architecture for fingerprint.



FWSA
ESCA
ESCal
OCSAs
OCSel
OCSel
OCSel
OCSel
Cover Fingerprint
Compute Block wise DCT
Divide image into 8 × 8 blocks
Integrating Block wise results
Block wise embedding strength and optimum DCT coefficients
Concatenate four MSB's of each pixel to generate the binary watermark string P
Embed P with the calculated embedding strength in the PSO selected DCT coefficients
Facial Watermark
Compute Block wise Inverse DCT
Watermarked Image

Figure 2.3 : Proposed Multi-Agent Framework for fingerprint

Source: https://ieeexplore.ieee.org/abstract/document/6622435

2.2.3 Face and fingerprint recognition using multi-agent technology

Mostafa Akhavansaffar has presented a multi-agent biometric identification system by combining the features of fingerprint and face [7]. User authentication systems, which use a biometric process, usually face noisy data and unlimited error rates. Combinatory or hybrid biometric systems are used to enhance the execution of special adaptations and matches in such situations. We will refer, in this paper, to suggested approaches in this field. Many approaches have been suggested for using the ANN in biometrics. After evaluating specific methods in feature detection, concentrate on and compare the ANN-based method and choose the best efficiency method.

Wasim Shaikh and colleagues have presented face recognition using a multi-agent system [8]. Using Multiagent Program, Face Tracking and Face Recognition will help us identify and recognize the human face as an image provided to it. Face Recognition System is a computer program used to automatically identify or verify an individual from a digital video source. Normally this is done by comparing selected facial features from the image and a facial database. Usually used in safety systems and similar to other biometrics such as fingerprints or eye iris recognition systems, facial recognition software is based on the ability to identify a face by analyzing the different facial features.

This system can be used in the security domain, such as anti-terrorism, face recognition system, which is issued to recognize the terrorist image as the input image and match the detected image if the match is found to be an authority that can easily arrest the individual. JADE uses the Asynchronous Message-passing mechanism. Each agent has of the mailbox (agent message queue), whenever a message is posted in the message queue the receiving-agent is notified. Nevertheless, if and when the agent actually receives the message from a message queue to process it is entirely up to the programmer.

2.3. Functional Issues

The functional issues have been discussed under the limited data. rely on paper-based identity and complexity section. In biometric domains, past data were limited as well as still rely on paper-based identities in many authorization processes and it has been more complex due to the different outcome of different recognition algorithms.

2.3.1 Limited data Issue

The nonlinear problem is solved in artificial neural networks. A non-convergent chaotic neural network is proposed in order to recognize human faces [9]. This paper provides a radial basis for neural network functions combined with a non-negative matrix factorization to identify faces. In addition, use an impulse back propagation neural network for face and voice verifications. In the same research, non-negative sparse coding methods are used to learn facial characteristics using different distance metrics and standardized cross-correlation for face recognition. A post-union decision-based artificial neural network approach is proposed as it has elements of both neural networks and statistical approaches and replenishes methods for identifying partly distorted and occlusive face images. Sadly, like other statistical-based approaches, this approach is unreliable to model classes provided only one or a small number of samples of learning.

Machine learning facial recognition methods usually assume that multiple samples are available per person for discriminative extraction during the training phase. In many practical applications of face recognition, such as rule enhancement, e-passport, and classification of ID cards, Nonetheless, this presumption cannot be maintained because there is only one test per person enrolled or registered in these systems. In this scenario, most common face recognition methods fail because there are not enough samples for discriminating learning.

2.3.2 Rely on paper-based identity

David White and his co-workers have conducted research on passport officers' errors in face matching [10]. They concentrate on the practical issue because protection is often dependent on specific facial decisions. For airport security, for instance, it is important to check that a passenger is a person they claim to be by verifying that they suit the person pictured on the passport. Passport officers were asked to decide whether the "cardholders" posing matched their displayed passport photo. In this mission, on 14 percent of occasions, passport officers wrongly admitted non-matching "fraudulent" cards. To order to address the scale of the problem, note that more than 100,000 people travel through Sydney Airport every day on average and more than 12 million foreign passengers traveled through this airport last year. At this level, an error rate of 14% (approximately one in seven) will represent a significant risk of travelers with fake passports being admitted. They also showed that, despite their experience, passport issuers do not perform better than the rest of the population. This surprised us and suggests that experience alone doesn't necessarily overcome the difficulty of the task. Since this research, several studies have shown high error rates in these activities, even in optimal conditions where photographs are taken on the same day, in neutral pose and under very similar lighting. This paper concludes that "Passport Staff Miss One in Seven Fake IDs". Figure 1 shows the error rate in face identification with passport officers and students.

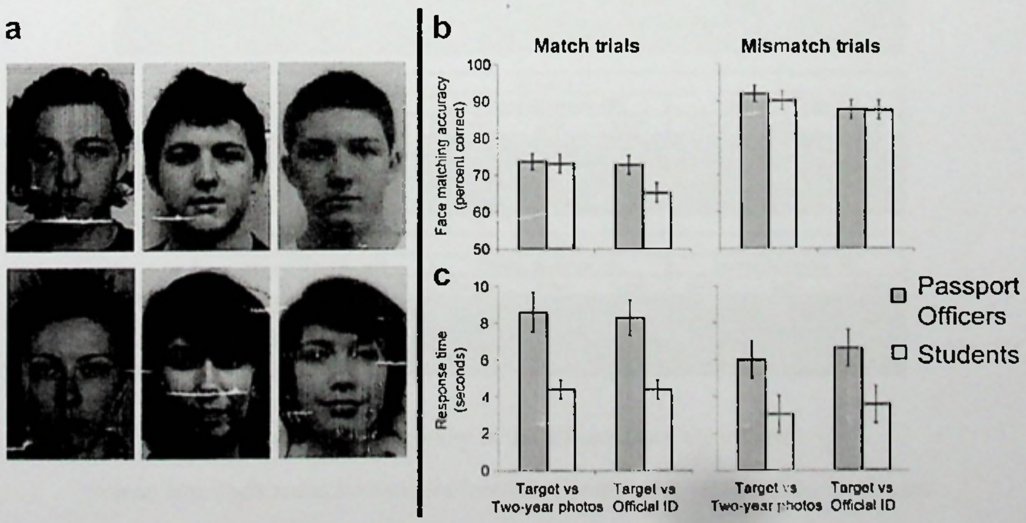
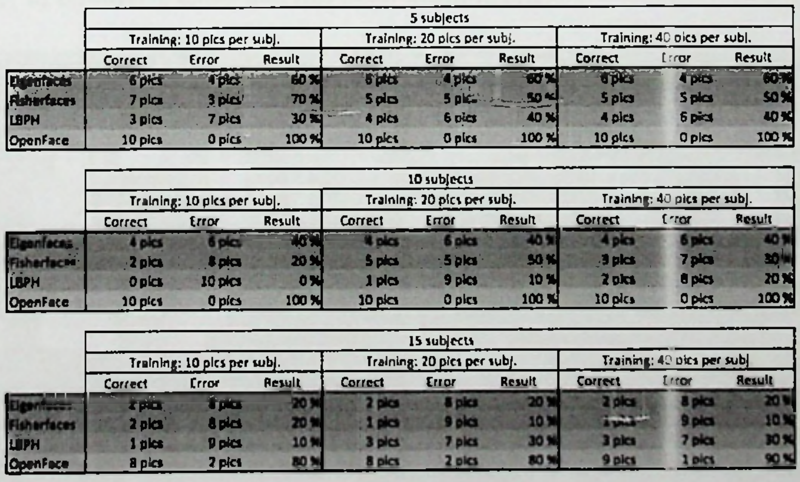


Figure 2.4 : Passport officer's error rate in face identification

Source: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0103510

2.3.3 Complexity

Nicolas Delbiaggio has done a comparison of facial recognition algorithms [11]. The researcher presented the complexity of different conditions with different face recognition algorithms. Eigenfaces during the first process were not very sensitive to a change in the number of subjects, but an increase in the size of the training set allowed the algorithm to correct its wrong prediction. In the first cycle, Fisher faces had better results with more data. His attitude in the second phase, however, was different. Sometimes with 20 pictures, the results were better, but with 40 pictures the results were the same or worse. For fewer samples of learning and time-critical applications, ANN is not suitable. Figure 2 shows the complexity of the result of different face recognition algorithms. This implies that complex algorithms make trouble in the time-critical situations for facial recognition hence giving different results at each literary. Hence, this complexity of the system gives different accuracy results, not a big advantage for border control situations. The complexity of different face recognition algorithms as shown in figure 2.5.



	5 subjects								
	Training: 10 pics per subj.			Training: 20 pics per subj.			Training: 40 pics per subj.		
	Correct	Error	Result	Correct	Error	Result	Correct	Error	Result
Eigenfaces	6 pics	4 pics	60 %	6 pics	4 pics	60 %	6 pics	4 pics	60 %
Fisherfaces	7 pics	3 pics	70 %	5 pics	5 pics	50 %	5 pics	5 pics	50 %
LBPH	3 pics	7 pics	30 %	4 pics	6 pics	40 %	4 pics	6 pics	40 %
OpenFace	10 pics	0 pics	100 %	10 pics	0 pics	100 %	10 pics	0 pics	100 %

	10 subjects								
	Training: 10 pics per subj.			Training: 20 pics per subj.			Training: 40 pics per subj.		
	Correct	Error	Result	Correct	Error	Result	Correct	Error	Result
Eigenfaces	4 pics	6 pics	40 %	4 pics	6 pics	40 %	4 pics	6 pics	40 %
Fisherfaces	2 pics	8 pics	20 %	5 pics	5 pics	50 %	3 pics	7 pics	30 %
LBPH	0 pics	10 pics	0 %	1 pics	9 pics	10 %	2 pics	8 pics	20 %
OpenFace	10 pics	0 pics	100 %	10 pics	0 pics	100 %	10 pics	0 pics	100 %

	15 subjects								
	Training: 10 pics per subj.			Training: 20 pics per subj.			Training: 40 pics per subj		
	Correct	Error	Result	Correct	Error	Result	Correct	Error	Result
Eigenfaces	2 pics	8 pics	20 %	2 pics	8 pics	20 %	2 pics	8 pics	20 %
Fisherfaces	2 pics	8 pics	20 %	1 pics	9 pics	10 %	1 pics	9 pics	10 %
LBPH	1 pics	9 pics	10 %	3 pics	7 pics	30 %	3 pics	7 pics	30 %
OpenFace	8 pics	2 pics	80 %	8 pics	2 pics	80 %	9 pics	1 pics	90 %

Figure 2.5 : Complexity of the different face algorithms

Source: https://pdfs.semanticscholar.org/7b66/ef8d74b6d32b03d69d37473f855aa80817f1.pdf

Girija Chetty and co-workers have presented a new approach to the application of agent technology to the problem of face recognition [12]. Multimodal biometrics can provide more comprehensive solutions to many applications' safety and convenience requirements, such as video surveillance, crime investigation and healthcare scenarios, where identity recognition is needed from insufficient biometric sensor data. The face recognition systems deployed in such decentralized environments need the configuration of the system to be scalable and versatile, for which an approach based on an advanced multi-agent-based model can be very promising. Cooperation, collaboration, and negotiation between agents are the easiest and safest way to achieve biometric security. The architecture consists of the combination of multi-layered structural and functional models into a hierarchical network-oriented framework. The system uses can biometric modalities for agent-oriented implementation to test and verify identity, and a set of novel fusion techniques can be used to combine multiple biometric trait data to achieve optimal authentication efficiency. The technique used for fusion adapts to environmental and human variations in which a system is accessed, and involves multiple features such as relevance, privacy, and data value, capturing the environment and identifying individual biometric trait success stories for the user. Figure 2.6 shows how to face recognition complexity in the complex background area. However, in biometric case, this has been minimized.

Figure 2.6 : Face detection in a complex visual background

Source: https://pdfs.semanticscholar.org/7b66/cf8d74b6d32b03d69d37473f855aa808f7f1.pdf

Using a smart agent solution makes it possible to effectively control the complexity of remote access use multi-biometrics. Smart autonomous agents and multi-agent systems create a vibrant area of research that is growing rapidly. Agents can be described as sub-systems that communicate and act independently with a particular environment. Therefore, in their interactions with other agents, they are versatile in reacting to their situation, pro-active in leveraging incentives and pursuing goals and "personal." In addition, they may also have other useful properties. such as adaptability and flexibility.

2.4. Technical Issues and Opportunities

Marco Ganassi and co-workers have described a Multi-modal multi-paradigm agent-based approach for developing scalable biometric distributed systems to classify [13], recognition and protection of confidentiality at a high level. The paper's goal is to address the specific problem by focusing on optimization, adaptation, and evolution by using dynamic architecture. The paper also explored how MAS operates in a native manner on adaptive and changing methods provided by message transmission as well as the sensitivity of errors. The paper showed an example of the overall process and concluded on one important aspect, which is that biometric measurements can generate non-reversible security and privacy code. Figure 2.7 shows multi-model biometric describes in Marco Ganassi research and its complexity behavior.

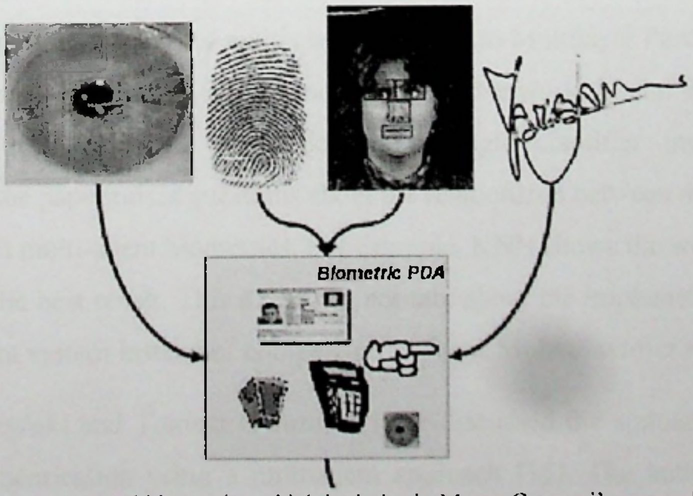


Figure 2.7 : Multi-model biometrics which includes in Marco Gamassi's research

However, authors are not involving implementation and evaluate the system with a considerable amount. The research will continue with further optimization, confidence, reliability, performances, security, and cost with real-world applications.

M.C. Da Costa-Abreu and his co-workers have discussed analyzing the benefits of a novel multiagent approach in a multimodal biometrics identification task [14]. This paper has an impact on the versatile biometric-based system that can allow more than one modality to be deployed. According to the paper, if the accuracy of identification is the main concern, a common alternative approach is to use a multi-classification approach to build a more accurate system. Multiclassifier systems are designed to incorporate more than one classification algorithm to determine the correct label for the assignment of a test sample and to combine different techniques for cooperatively solving a problem. Multi-classifier has been widely used in a range of pattern recognition problems. Nonetheless, the best choice of a combination approach that is best suited for a particular application is a difficult process that often needs rigorous testing in order to select the best implementation. Experience in creating more powerful and flexible classifiers for biometrics-based processing tasks has shown that performance in personal identification tasks can generally be significantly improved first by adopting a multi-classifier strategy and secondly by implementing a new agent-based classification in such architectures. In addition, the paper provided some quantitative data on classifier quality and an initial study to develop strategies to help more complex configurations of multi classifiers such as Multilayer Perceptron, Fuzzy Multilayer Perceptron, Radial Basis Function Neural Network, Radial Basis Function Neural Network, etc rather than adopting a single classifier implementation. Nonetheless, the paper raises questions about the relationship between multi-classifier biometrics and multi-agent biometrics. For example, KNN shows the worst result and SVM shows the best result. This paper did not talk about the implementation part of the Multi-agent system instead of comparison with the Multi classifier model.

Adrian Kapczyński and Tomasz Owczarek have discussed the simulation model of biometric authentication using a multiagent approach [15]. The authors present a model based on an agent that monitors the actions of authorized users in the journal. The agents are therefore responsible for recording all authorized user activities and for transmitting all recorded data to the central storage and decision-making manager. In this approach, 4 models were created in the Net Logo platform with agents playing different roles such as authenticator, authenticate and other with remain few future challenges.

Marco Tranquillin and others discuss the usage of mobile agents for secure biometric authentication [16]. Furthermore, this paper discusses smartphone security authentication using the password with biometric matching. The author describes the importance of hybrid solution (biometric and non-biometric) than fully AI transformation. The main objective of this paper is to explain how to combine biometric matching methods with the normal password / PIN. Nevertheless, this paper more highlights with fingerprint [15], which has less security biometric levels. The author did not discuss face recognition to combine the MAS which second important characteristic of biometric. Furthermore, the author got stuck with an experimental phase due to not enough computational power of the smart-phones.

2.5. Face recognition techniques and opportunities

Many researchers have used popular techniques like principal component analysis (PCA), linear discriminant analysis (LDA) and backpropagation neural network which can be considered as accurate in the face recognition domain. They use those technologies in different applications. However, most researches are conducted with still images and do not apply in time-critical applications. In the following section, the authors review the usage of the above-mentioned techniques and how their contribution and limitation affect the final solution.

Nawaf Hazim Barnouti et al. have introduced face detection and recognition using Viola-Jones with PCA-LDA and squared Euclidean distance [17]. The proposed system focuses on appearance-based features than local facial features. Face detection is the first part of the process and the Viola-Jones face detection method is identified as highly accurate in detection. Feature extraction will be applied after the detection phase. The PCA approach is commonly used in pattern recognition and the LDA system used to address PCA weaknesses and the new method tested in three databases

(MUCT, Face94, and Grimace). Eight individuals' one to three images are used from the database for testing the overall performance using Matlab software. The analysis shows that increasing the number of images from 1 to 8 would provide more accurate results and only a few seconds of recognition time. However, they do not consider real-time face detection and recognition. Figure 2.8 shows the latest biometric system at Narita, Japan airport.

Figure 2.8 : Latest biometric machine at Narita, Japan airport

Source: https://www.tsa.gov/biometrics-technology

Riddhi A. Vyas and his co-workers have discussed the correlation with the precision enhancement of PCA and LDA techniques for face recognition feature-based extraction [18]. The human face is a dynamic multidimensional template and two-step recognition; face detection and extraction of features. The paper has been focusing on comparative analysis of two features extraction algorithms; PCA and LDA with different criteria; facial expression, illusion, and glasses. The system has been implemented in C# - windows-universal platform and used input from the Yale database. LDA shows the better result in the Yale database with different criteria and it has accurately identified 123 images out of 165. Even though the paper mentioned that the hybrid method of LDA and PCA would give better results it has not been implemented. Utilizing still images in the implementation of their proposed system is the weakness of the system.

Mayank Chauhan and his co-workers show in their study and analysis the different face detection techniques [19]. The main goal of the paper was to come up with an approach that was good in face detection. There were four types of face detection approaches, ie. feature-based, geometric based, high-level language-based and Haar-like feature-based. The experiment carried out by the theoretical comparison of each approaches in terms of key parameters. Haar-like features are comparatively good in face detection. However, they have not implemented or carried out the experimental comparison in their study.

Kandla Arora has carried out a real-time application of the face recognition concept [20]. Face recognition can be considered as a holistic approach and use various potential applications such as biometrics, security, smart cards, and access control. There are two techniques in the traditional face recognition system, photometric and geometric approach face recognition using different light conditions and distance of face image points respectively. In proposed techniques author use PCA and eigenface which calculate the difference between a given face image and mean face image stored in a computer. If the deference is below the threshold value, it is considered a known image. The similarity score is calculated between input face image and training face image. The experiment conducted in different conditions and head orientations. The result shows higher accuracy in normal conditions. However, the system has been limited in still images and not compared with other face recognition algorithms. Also, the system does not talk about noise reduction and usage of the quality webcam is considered as the main disadvantage.

Ashutosh Chandra Bhensle and his co-workers have proposed an effective face recognition using distance classification PCA and Euclidean [21]. Instead of acquiring a high dimensional it is better performance face vector in low dimensionality for security measures and access control. The improved PCA algorithm uses facial features and classification by minimum distance which is comparatively a simple face recognition system using webcam or video. The experiment shows a better result with frontal view face images. However, the system is using only the PCA algorithm without comparing other algorithms, identified as a weakness of the total system.

A face detection algorithm based on deep learning has been proposed by Ming Li et al [22]. This paper has proposed a solution for partial occlusion and multi-pose using face detection algorithm based on deep learning. Experiment results show higher accuracy in case of occlusion and multi-pose with limitation in poor light condition. However, the author focuses on a deep learning approach that can be identified as a time consuming and computer power consuming system. Therefore, deep learning is not suitable for the time-critical environment as we expect time and accuracy both.

Sankar Suhas and his co-workers have implemented a face recognition using PCA and LDA on the holistic approach in facial images database [23]. This paper compares holistic approaches to PCA and LDA. PCA has been identified as a more accurate method in standard face 95 databases. The experiment has been conducted with various poses, angles and light condition face images. However, they were unable to find a proper solution for real-time face recognition.

Fares Jalled has presented a face recognition machine vision system using eigenfaces [24]. The eigenface approach is quite simple and efficient in a controlled environment such as one with less noise. The system has been compared between PCA and N-PCA analysis and the experiment has been carried out with an Indian face database. The accuracy of the system has been measured by Euclidean distance between test faces and train faces. N-PCA has shown positive results than PCA over the ORL face database. However, the main problem of the system is not combining or comparing it with other face recognition algorithms. Face recognition in the real-time environment too has not been discussed within the study and they are identified as a weakness of the system.

Liton Chandra Paul and his co-workers have introduced a system to face recognition using PCA [25]. The authors have implemented a face recognition system with PCA and eigenface approach. The system recognizes the faces and different angles of faces at a higher rate. The eigenfaces approach has been tested with image database ETE 07 series, RUET and implemented using MATLAB. Eigenfaces approach is practically fast and simple compared to other methods of the face recognition system. PCA is a statistical algorithm that can reduce the number of variables in the face recognition phase. However, the system has not implemented real-time face recognition. The main drawback of the system is not comparing with other face recognition algorithms such as LDA. Jun Huang and his co-workers have implemented an MPCA/LDA based dimensionality reduction algorithm for face recognition. The proposed system was based on both multiple principal component analysis (MPCA) and LDA with the K nearest neighbour (KNN) classification. Three databases ORL, FERET, and YALE were used for testing the overall performance of the system. As opposed to traditional method MPCA + LDA plays in a good relationship and it is a system capable of recognition of different light conditions with various facial expressions. However, the system utilized only a few images is the main drawback of the system.

2.6. Overview of Biometrics

Biometrics applications are broad and support not only private organizations, but also government agencies and service providers. For identification or surveillance purposes, these groups turn to biometrics. For example, for border control and welfare distribution systems, fingerprints are used by Hong Kong and India. Figure 2.9 shows all biometric levels in which we can take the computational account.

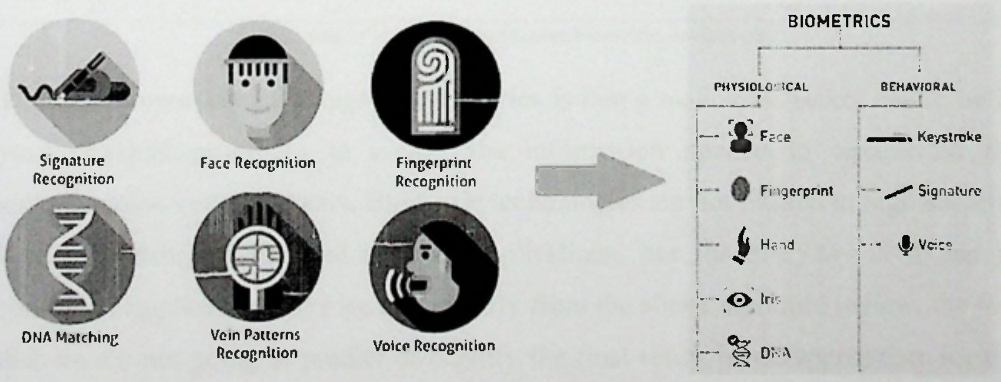


Figure 2.9 : All biometric levels

Source: https://www.tsa.gov/biometrics-technology

Walt Disney World Florida theme park takes full advantage of biometrics, using mobile technology and fingerprinting to create a magical environment for its visitors. Disney guests gain access to the theme park after acquiring a customized wristband and putting their index finger on a biometric scanner and enjoy personalized service during their entire stay. It is much easier to use a fingerprint or iris scan than a password, particularly a long one. It takes only a second (if so) to recognize a fingerprint for the most modern smartphones and allow a user to access the phone. Ultrasound scanners will soon become popular as they can be placed directly behind the screen by manufacturers without requiring any extra property on a mobile. Table 2.1 shows the summary of the biometric comparison.

Table 2.1 - Biometric Comparison

Biometric Level	Accuracy	Cost	Device Required	Social Acceptability
DNA	High	High	Test Equipment	Low
Iris Pattern	High	High	Camera	Medium - Low
Retina Scan	High	High	Camera	Low
Facial Image	Medium	Medium	Camera	High
Voice	Medium	Medium	Microphone	High
Hand Geometry	High	Medium	Scanner	High
Fingerprint	Low	Medium	Scanner	Medium

Source: https://www.bayometric.com/biometric-devices-cost/

The most important advantage of biometrics is that a malicious hacker has to be in your physical proximity to collect the information needed to circumvent the authentication in recent years, Biometric technologies are not limited to high-security border control and national security applications, but for everyday civil and e-commerce applications. As we can identify from the above literature review, the fact that we are not going to predict or classify the final result is not appropriate for the machine learning algorithm to deal with the biometric authentication process. On the other hand, to get the expected result, we have fewer data and no mush time. On the other hand, using MAS alone, it does not display the successful and timely result.

2.7. Summary

After a comprehensive study, the authors concluded that the best approaches for face recognition in time-critical applications are PCA and LDA in the MAS environment. According to many past researchers, there is a need for a system to recognize the face in a time-critical environment in an effective way. Therefore, their proposed solution used past researchers' contributions to get an idea about the most suitable algorithms in each step. In the next chapter, the Technology that is used in this system will be discussed in greater detail.

Table 2.2 - Summarization of the Literature review

Research	**Summary**	**Reference**
Face Recognition: Issues, Methods, and Alternative Applications	Limited data is an issue with face recognition applications	Waldemar Wójcik, Konrad Gromaszek andMuhtar Junisbekov
Passport Officers' Errors in Face Matching	Passport officers have done one mistake for every 140.	David White, Richard I. Kemp, Rob Jenkins, Michael Matheson, A. Mike Burton
A comparison of facial recognition's algorithms	A wide range of facial recognition algorithms gives different results.	Nicolas Delbiaggio
Distributed Face Recognition: A Multiagent Approach	Thanks to the integration with the agent-based model, the robustness of the complex face recognition system are increased	Girija Chetty, Dharmendra Sharma

A multi-modal multi-paradigm agent-based approach to design scalable distributed biometric systems	The original feature is a modular, standardized system-level design approach based on the use of distributed systems, multi-agent architectures.	M. Gamassi, V. Piuri, D. Sana , F. Scotti, O. Scotti
Analyzing the Benefits of a Novel Multiagent Approach in a Multimodal Biometrics Identification Task	Illustrate the benefits of applying multi-agent computational architecture as a means of achieving high levels of performance when the main criterion is the accuracy of recognition.	Márjory Cristiany Da Costa Abreu and Michael C. Fairhurst
Simulation Model of Biometric Authentication Using Multiagent Approach	Present the concept of multi-agent application in modeling systems for biometric authentication	Adrian Kapczyński and Tomasz Owczarek
Using mobile agents for secure biometric authentication	Biometric matching, over a Multi-Agent distributed infrastructure.	MarcoTranquillin, CarloFerrari, MicheleMoro
Face Detection and Recognition Using Viola-Jones with PCA-LDA and Square Euclidean Distance	The face recognition system is proposed based on appearance-based technologies that focus on the whole face image and not on local facial features.	Nawaf Hazim Barnouti, Sinan Sameer Mahmood Al-Dabbagh, Wael Esam Matti, Mustafa Abdul Sahib Naser

Comparison of PCA and LDA Techniques for Face Recognition Feature-Based Extraction with Accuracy Enhancement	Comparative analysis of two interface PCA and LDA facial recognition techniques on different criteria	Riddhi A. Vyas, Dr.S.M. Shah
Study and Analysis of Different Face Detection Techniques	several existing face detection approaches are analyzed and discussed	Mayank Chauhan, Mukesh Sakle
Real-Time Application of Face Recognition Concept	Description of the Face Recognition concept's real-time implementation by constructing a MatLab software using the image acquisition toolbox.	Kandla Arora
An Efficient Face Recognition using PCA and Euclidean Distance Classification	The improved PCA algorithm takes out facial characteristics and classification is carried out by classification of the minimum distance.	Ashutosh ChandraBhensle1, Rohit Raja
A Face Detection Algorithm Based on Deep Learning	The problem of partial occlusion and multi-pose in the face detection	Ming Li, Chengyang Yu,Fuzhong Nianand Xiaoxu Li
Face Recognition Using Principal Component Analysis and Linear Discriminant Analysis on Holistic	Comparative study of principal component analysis and linear discriminant analysis	Satonkar Suhas S.1, Kurhe Ajay B.2, Dr.Prakash Khanale B.

Approach in Facial Images Database		
Face Recognition Machine Vision System Using Eigenfaces	Face recognition test quality correlation using Principal Component Analysis (PCA) and Standardized Principal Component Analysis (N-PCA)	Fares Jalled
Face Recognition Using Principal Component Analysis Method	Addresses the facial recognition process through the use of Principal Component Analysis (PCA).	Liton Chandra Paul1 , Abdulla Al Sumam
An MPCA/LDA Based Dimensionality Reduction Algorithm for Face Recognition	Multilinear Master Component Analysis (MPCA) and Linear Discriminant Analysis (LDA) face recognition algorithm.	Jun Huang, Kehua Su, Jamal El-Den, Tao Hu, and Junlong Li

Chapter 3

Technology Adapted

3.1. Introduction

Multi-Agent System (MAS) principles to enable authentication and authorization processes to operate more effectively and efficiently for multi-applications and multi-client [15]. Agents are software entities that have enough anatomy and intelligence to carry out with little or no human intervention. MAS consists of a team of several communicating agents and is well suited to circumstances where there are several viewpoints of a problem-solving situation. Types of interactions that are best suited to biometric security include collaboration, communication and interagent negotiation. The data provider's needs to develop adequate trust in the user may need to be matched with the user's biometric information privacy and ease of use of the program. For each facility, transaction or session, a balance may need to be defined and may even be dynamically changed during use. Tasks for agent systems include, for instance, managing multiple authorization rates, location of data across various repositories, and user interface and quality adjustments as needed by the client or as required by the environment [18]. Figure 3.1 shows basic communication in-betweens the multiple agents.

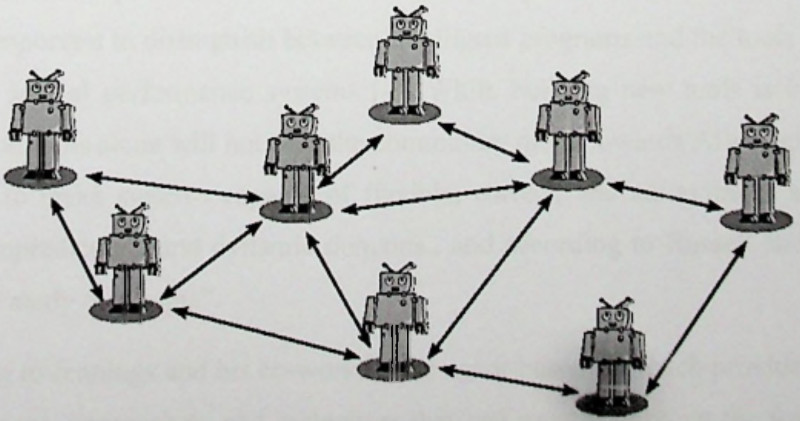


Figure 3.1: Multiple agent-based biometric authentication overview

- Proactiveness : Agents should take initiative; they don't just wait for a cue to start acting, but they can take action to achieve their objectives.
- Autonomy Reactivity Social ability compete.
- The agent is an independent entity capable of operating without direct control.
- Agents respond to signals their environment perceives.
- Agents connect, chat, collaborate and eve with each other

For this purpose, the JADE framework is the most successful approach as it has platform-independent language (Java) and open-source features. In addition to the abstraction of the operator, JADE provides a simple but effective model of task success and structure, peer-to-peer interaction based on the asynchronous paradigm [18].

An agent is a computer entity acting on behalf of a different entity (or entities). Agent systems are software programs that monitor field awareness and the ability to act independently in order to achieve specific goals. They are designed to operate in environments that change dynamically or are unstable.

3.2. History of Multi-Agent Technology

In the early 90s, pioneers in the domain of Artificial Intelligence such as McCarthy and Nilson have expressed their discontent on existing technologies for AI, indicating that it's important to distinguish between intelligent programs and the tools that they use, aka, special performance systems [4]. While building new tools is important, working on tools alone will not help the community move towards AI's original goal; which is to make systems capable of flexible, correct, and autonomous actions in various unpredictable and dynamic domains., and according to Russell and Norvig, "AI is the study of Agents".

According to Jennings and his co-workers the agent-based approach provides a range of techniques, approaches, and metaphors that can greatly improve the way people conceptualize and execute different types of technology [6]. Multi-Agent technology is used in many different types of applications from small scale email filters to large scale mission-critical applications such as air-traffic controls. Even though it might appear that this range of applications will have very little in common, on the contrary, in both these applications the Agent is used as the key abstraction. This is a principle that enables very natural and easy conceptualization of a wide range of agent-related applications, making researchers and developers in the field quite optimistic about the future potential of this technology. The interest that has developed about the Multi-Agent Technology didn't emerge out of the blue, rather, researchers and developers from many different domains have been closely studying and discussing the capacity of this technology for quite some time. The main contributors for this are:

- Artificial Intelligence
- OOP (Object Oriented Programming)
- HCI (Human-Computer Interface)

Undoubtedly, one of the main contributors to the field of Multi-Agent Systems is the domain of Artificial Intelligence. This is a domain which studies intelligent artifacts, and if such artifacts have the capability of sensing and acting in a given environment, they can be considered as Intelligent Agents [6]. Even though Agent technology is one of the main areas in AI, until the 1980s, there was only very little interest and effort within the AI community to study intelligent agents. The main reason for this can be identified as the tendency of AI researchers to study the intelligent behavior of individual components such as learning, reasoning, and problem-solving. The expectation at the time was that studying such components independently would prove more successful, and creating intelligent agents by synthesizing these individual components would be quite straightforward. This assumption seems to have been implicit within the AI researchers throughout the 1970s. However, one exception to this rule was the area of AI planning, on which there were researches connected to intelligent agents.

Early AI planning research carried out during the 1970s and early 1980s focused primarily on the various representations required for actions, their planning algorithms, and their effectiveness. Although some micro-world examples seemed to provide reasonable performance, it was quickly observed that they do not scale well with large realistic scenarios. This apparent failure in early researches on AI planning techniques to scale to cater to the real-world scenarios led many researchers during the mid-1980s to discuss and question the viability of conventional reasoning approaches for AI and to explore different methodologies to solve these issues. One of the best-known such critics can be named as Rodney Brooks, who presented various different objections towards symbolic AI modeling through a series of published papers. As a part of his research, Brooks developed an agent control architecture, known as the "Subsumption Architecture", which did not employ any sort of symbolic reasoning or representations; rather utilized a collection of intelligent agents with task accomplishing behaviors.

By the early 1990s, most researchers found that reactive architectures can only be adapted to certain domains and issues while being less suitable for others, in addition, most of the problems did not work for purely reactive or deliberative architectures. This led many researchers to investigate hybrid architectures, which tried to synthesize the best aspects of both reactive and deliberative approaches. In these hybrid architectures, the Subsumption Architecture, implemented by a collection of agents with task accomplishing behaviors, played an important role.

3.3. Multi-Agent Technology

The main technology used in the proposed system is Multi-Agent Technology; therefore, it's crucial to properly understand the technology before moving on to the subsequent chapters of this study. First, it's important to define what is meant by such terms as "agent", "agent-based system" and "multi-agent system". However, this is not simply because some primary terms in the field lack definitions that are universally accepted. Even up to this date, there's a lot of deliberation going on regarding what exactly is meant by an Agent. Even though this might be considered an obstacle to its development, the AI community still has been able to achieve leaps and bounds when it comes to agent-based technology, without such universally accepted definitions. Nevertheless, it is worth spending some time on the issue, otherwise, the terms that are used in the remainder of this study will come to lose all meaning. Therefore, an agent can, therefore, be described as a computer system in some environment that is capable of versatile autonomous action to fulfill its design objective. Therefore, our interpretation comprises three key concepts: place, independence. and versatility [6].

In this case, location means that the agent receives sensory input from his surroundings and can perform actions that in some way alter the environment. The physical world wide web can be named as an example of where Agents are situated in different locations, which is in contrast with concepts such as Expert Systems that discusses disembodied intelligence. As an example, MYCIN, which is a paradigm expert system, did not have any interactions with the physical environment. Rather it received information through a middle man via different sensors [6]. Similarly, it did not act upon the environment, rather it provided feedback to different third parties. The second key concept in Agent Technology is Autonomy which is again hard to be defined precisely, however, in layman terms, it means the ability to work on its own accord without human intervention, and the ability to control its actions and the state. In a more defined sense, Autonomy can also refer to systems that can learn and adapt on its own.

There are various examples for situated and autonomous systems at present including any process control system, which must monitor a real-world environment and perform actions in order to modify its conditions automatically; these systems can vary from simple thermostats to quite complex and sophisticated nuclear control systems. Another similar example would be software daemons, that has the ability to monitor software environments and modify its conditions by performing various actions; the UNIX xbiff program can be named as an example for such a system. which monitors an incoming email by displaying an icon when a new incoming email is detected.

Although the above are definitely examples of situated, autonomous systems. they cannot be regarded as agents as they are unable to function flexibly to achieve their development goals. By being flexible, it implies that the system is:

- **Responsive**: Agents should interpret their environment and respond to changes occurring within it in a timely manner.

- **Pro-active:** Agents should not necessarily act in response to their situation, they should be able to demonstrate opportunistic, purpose-driven actions and take the initiative where appropriate.

- **Social:** When needed, agents should be able to interact with other artificial agents and individuals to complete their own problem solving and help others with their activities.

Of course, some agents will have additional features, and some characteristics will be more important for certain types of applications than others. Nonetheless, the common belief is that ail attributes in a single software entity provide the power of the model of the agent and these attributes differentiate agent systems from software paradigms such as OOP-based systems, expert systems, and distributed computing systems.

With the basic buiiding block notion of an agent in place, more related terminology can be defined. By an agent-based system, the key abstraction used is that of an agent. In theory, an agent system can be designed by conceptualizing different types of agents and implemented without any software structure that represents them. This can be viewed in contrast with object-oriented programming, where an object-oriented program can be built but implemented without the use of an object-oriented software environment. However, this can be quite counter-productive and unusual. There is a similar situation with agent software, so an agent-based system is supposed to be built and applied in terms of agents.

As previously described, an agent-based system can contain one or more agents, although one agent may also be sufficient. A good example is the class of programs known as expert assistants in which an agent functions as an expert assistant to a client who attempts to use a device to perform certain tasks. However, when a system is designed in a pure Multi-Agent structure using different types of agents incorporating different features, it can be quite interesting and sophisticated in terms of a Software Engineering perspective. Multi-Agent systems are best used in order to solve issues that have different methods of solving them, different standpoints and/or even different types of entities with the ability to solve that problem. Typically, such systems have the advantage of more sophisticated interaction patterns. These interaction patterns include cooperation (working towards a common goal together); coordination (proper organization of an activity that solves a problem while minimizing damaging interactions and exploiting advantageous interactions); and negotiation (all the involved parties coming to an agreement that is acceptable). The nature and sophistication of these interactions distinguish multi-agent systems from other forms of computers and provides the fundamental force of the model.

3.4. Popular Frameworks for Multi-Agent System Development

Nowadays, numerous tools are available for the development of Multi Agent-based systems. Such tools have facilities for naming, executing, managing the execution, accessing system resources, maintaining integrity and protection of agents and the platform, as well as for supporting the migration of location and communication services. While there are hundreds of tools available for multi-agent systems development [22], in this research, the JADE Multi-Agent development environment has been selected.

Among others, JADE is the most generic and the most commonly used Multi-Agent development framework in this domain. JADE comprises a middleware system that has been built in compliance with the FIPA specifications, which simplifies the implementation of the Multi-Agent System. In addition, it also provides the developers with a graphical toolset to support easy troubleshooting and deployment [23].

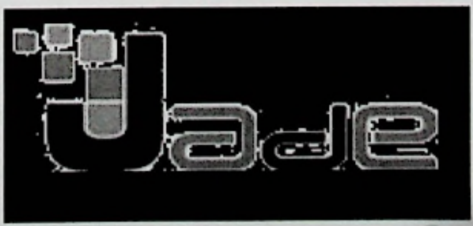


Figure 3.2: JADE Multi Agent Development Framework Logo.

Source: https://jade.tilab.com/

A JADE-based system can be spread over multiple machines and a remote GUI can manage the configuration. Even at runtime, the configuration can be modified by moving agents from one machine to another, as needed. Apart from the agent abstraction, JADE also comprises of a powerful task composition and execution model, an agent communication model with peer-to-peer support based on the asynchronous communication paradigm, a yellow-pages service that allows for publishing and subscribing mechanisms and various other highly advanced features. Due to these capabilities and its reputation, JADE has been selected as the base system for this research. In addition, MadKit was also considered a strong candidate for the development of the system in this research. It's another lightweight, Java-based Multi-Agent development framework that also can support other object-oriented programming languages such as C++ through plugins.

3.5. Other Technologies Used in BMAgent System

Other technologies have been discussed under the programming language, user interface, database technology and facial landmarks by OpenCV sections.

3.5.1. Programming Language

Apart from the JADE Multi-Agent development framework, there are various other tools and technologies that were used in the BMAgent system. The first and foremost important decision was to decide on a proper programming language to be used for the core system implementation. As such, the Java Programming language was selected as the main programming language due to the following reasons [24]:

- **Simplicity:** Java language is a common, easy and efficient language to develop applications with, yet it also offers a lot of useful features to the programmers such as automatic memory allocation and garbage collection, etc.

- **Platform-independent:** Since Java offers the ability to run a written program on any hardware or a software platform that supports JVM, it allows easy compatibility from different computer systems.

- **Rich API and Library Support:** This is one of the main reasons for selecting Java as the programming language for the implementation of the BMAgent system. Since JADE, which is the framework that is selected for Multi-Agent development, is mainly supporting Java and due to the larger number of additional APIs and libraries that can be used when developing, Java was the prominent choice of language.

- **Light-weight and Powerful:** Another important deciding factor when selecting Java, is its ability to make the application light-weight and run quite efficiently, which is important for applications such as BMAgent that relies heavily not only on Multi-Agent based communication but also on the Business Logics that are relevant for the proper functionality of the system.

By considering the above factors, Java has been selected as the main programming language when developing the BMAgent system's Business Logic Layer and the Multi-Agent module.

3.5.2. User Interface

According to the architecture of the BMAgent system, the User Interface was decided to be developed on a separate layer that will communicate with the Business Logic layer of the system. As such, it was decided to develop a Web Application as the UI for the system with the use of *JavaScript* and *React*, which is a popular JavaScript library that can be used to develop rich and sophisticated User Interfaces. React is a library developed and maintained by Facebook Inc. that provides a large number of features to create Component-Based UIs, where each component has the ability to manage its own state [25].

3.5.3. Database Technology

Another important module for any Enterprise application is its database technology. Since the BMAgent system is aimed at manufacturing organizations that manage their day to day production schedule, it is vital to provide them the ability to manage their data with an appropriate Database Management System. At present, there are numerous DBMSs available both based-on SQL such as MS SQL, Oracle, MySQL and No-SQL such as MongoDB, Hadoop, Cassandra. However, since enterprise-level organizations need to keep track and interact with their databases more frequently, MySQL relational database has been selected as the main DBMS for the BMAgent system. Additionally, Amazon Web Services were used to host the MySQL database in the cloud, so that the system can access those data from anywhere.

Figure 3.3: MySQL logo

Source: https://www.mysql.com/

3.5.4. Facial landmarks OpenCV and dlib

Facial landmarks are defined as detecting and locating certain key points on the face that affect subsequent face-focused tasks such as animation, face recognition, gaze detection, face tracking, recognition of expression, understanding of gestures, etc. OpenCV as well as dlib. Many face recognition algorithms identify facial characteristics by extracting landmarks or characteristics from a picture of the head of the subject. An algorithm, for example, can determine the relative position, width and/or shape of the ears, nose, cheekbones, and jaw. Then these features are used to find other images with matching features. Many algorithms normalize a facial image gallery and then compress the face data, only retaining the image data useful for face recognition. The face data is then compared with a probe image. One of the earliest popular systems was based on the techniques used to match models for a range of prominent facial features.

Algorithms of recognition can be divided into two main methods: geometric, which examines distinguishing characteristics, or photometric, a statistical approach that distills an object into values and compares values with models to minimize variances. Some categorize these algorithms into two broad categories: holistic models based on functionality. The former attempts to identify the face as a whole while the feature-based components such as eyebrows are separated and each and its spatial position are evaluated in relation to other features. Popular recognition algorithms include the key element analysis using the Fisherface algorithm, linear discriminate analysis, elastic bunch graph matching, the hidden Markov model, Multilinear underground learning using tensor representation and matching neuronal driven dynamic ties.

3.6. Phase 1 - Biometric Capture

In 2001, Paul Viola and Michael Jones have introduced a technique to object detection known as "Viola-jones object detection technique" [11]. This is known as the first object detection technique and possible to execute in real-time. This technique requires a front image of the face and it will highlight the square of face area in the given image because it can match human face features in the given image. Viola-Jones face detection technique needs four main ingredients HAAR feature selection, integral image, AdaBoost and cascading to detect the face.

A. HAAR features

There are four HAAR features to detect the face in the image [9]. It takes a 24*24-pixel squares window and applies each HAAR features into that window. E.g. nose is brighter than the eyes. Finally, this can calculate almost 160000+ features per image. Figure 3.4 shows basic HAAR cascade features which are commonly use. Especially this used to detect the face using a given image using the following edges.

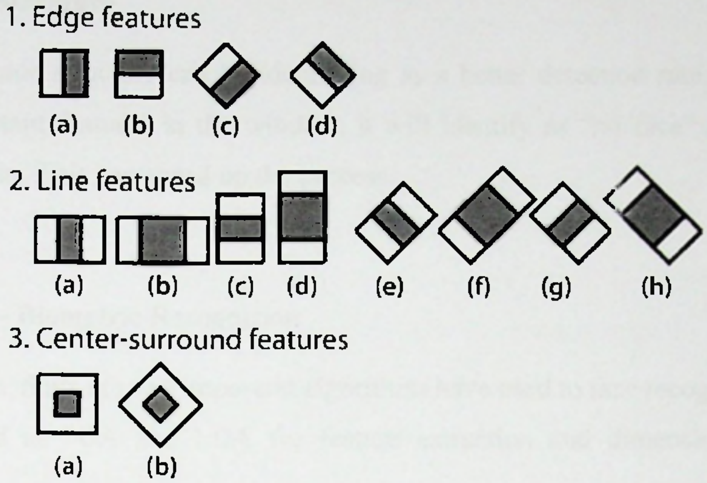


Figure 3.4: HAAR features

B. Generating an Integral Image

This phase covers the image of the input face in an integral picture [9]. As shown in the formula below (3,1), the integral picture can be determined:

$$I(x,y) = \sum_{x'<x\ ,y'<y} O(x',y') \qquad (3,1)$$

Where I is the integral image and O is the original image.

Features will be calculated in a constant time period. This is nothing but the cost-effective generation of summation of the pixel in the rectangle area of an image [14].

C. AdaBoost Training

AdaBoost Training is a machine learning algorithm that can find the most accurate features by combining all weak features [13] among them. This step can be identifying in many practical boosting methods. This system can remove irrelevant features and continue with relevant features.

D. Cascade Structure

Using a cascade structure can be identifying as a better detection rate. If there are absent important features in the window, it will identify as "no face" and it won't calculate again. This can speed up the process.

3.7. Phase 2 - Biometric Recognition

In this section, there are two important algorithms have used to face recognition; those are identified as PCA and LDA for feature extraction and dimension reduction respectively.

A. Principal Component Analysis (PCA)

Before initializing this process, face detection and cropping part have been done by the face detection phase. Predictive analysis and explanatory data analysis used to transform high dimensional into low dimensional. Sizes of M x M training images are converted into low dimensional face images by applying PCA. Principal components are known as correlated N variables into a set of uncorrelated k variables using a mathematical approach. This mathematical principle used to transform a set of correlated N face images into a set of uncorrelated K face images. A similar 2-D face image vector transforms into a 1-D face image vector and this can be either row or column vector. This is called eigenfaces which used to represent the current and new faces.

Steps in PCA Method:

1) A training set of M images are used to calculate the average image.

$$Average = \frac{1}{M}\sum_{N=1}^{M} TraningImage(n) \qquad (3,2)$$

1) The original image will subtract from the average image.

$$Sub = TraningImage - average \qquad (3.3)$$

2) Calculate the covariance matrix.

$$Coverince = \sum_{n=1}^{M} sub(n)sub^{T}(n) \qquad (3,4)$$

3) Calculate the eigenvectors of the covariance matrix.

Finally, select the best eigenvalues and then eigenvectors can be select by choosing the highest eigenvalues. These M eigenvectors consider as eigenfaces. In this method much discriminative information can be lost therefore, the system used another method to overcome the limitations. However, according to the past studies PCA is positive in the small number of images and it has less execution time [8].

3.7.1. Fingerprint Technology

A fingerprint's unique nature makes it ideal for use in automatic recognition systems. A fingerprint consists of a series of ridges and grooves. The device locates the minute points once a fingerprint is detected. Such specific points occur where the ridgelines start, end, branch off and merge with other ridgelines. Then these points are traced and between each point, a line is drawn. This produces a map of the relationship between each point and the other points. The map is then processed for future comparison with other fingerprints as a data stream called a minutia model in a server. It should be remembered that no fingerprint images are saved on the device during the entire process and that a fingerprint image cannot be recreated from the minute model. Figure 3.5 shows the flow diagram of fingerprint technology.

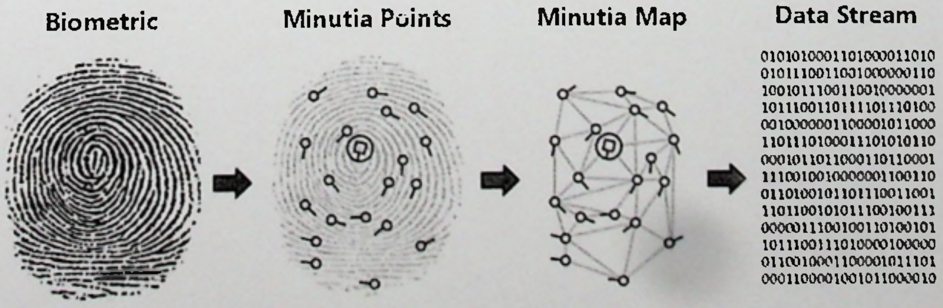


Figure 3.5: Fingerprint Technology

3.8. Summary

This chapter mainly deliberated on the Multi-Agent technology which is the main technology used in the BMAgent system. In addition, other technologies used in the development of the BMAgent system were also discussed such as Java as the main programming language for the development of the core, JavaScript-based ReactJS as the main development library for the UI and the use of GitHub as the version controller for the system. In the next chapter, it will discuss the approach of the BMAgent system.

Chapter 4

A Multi-Agent-Based Approach to Biometric Authentication

4.1. Introduction

In chapter3 we discuss multi-agent technology to solve the research problem biometric authentication technology this chapter presents our approach and biometric authentication. This approach is the BMAgent system, an acronym for Biometric Multi-Agent, Here the approach is presented in terms of hypothesis, input, output, process, features, and users of the system.

4.2. Hypothesis

Implementing a Multi-Agent based automated system, BMAgent, would be a solution for issues such as inefficiency, lack of proper identification system at the border control and it will also allow addressing issues with traditional approaches such as requiring a large amount of data, power and time.

4.3. Inputs

BMAgent system accepts multiple inputs from a different aspect of verifications. Here face images of the person are used as the main inputs. The image of the face is captured real-time by when the person passes through the security points.

4.4. Outputs

The output of the BMAgent system with the message saying whether the authentication was successful or unsuccessful.

4.5. Process

As a result of the literature review, Face recognition biometrics is the best way to approach the cost-effectiveness and accuracy of the solution. A facial recognition technique is a software program to automatically identify or validate a person from a video source's digital image or video frame. It is the most common way to identify biometrically.

As many researchers have done their system with PCA face recognition algorithms, this section describes the agent-based system using the PCA algorithms. In this section, the authors focus on the agent-based approach, design, and implementation of the proposed system. Face images and personal details are the main input of the system and authentication result is the output. As mentioned above in the introduction, the proposed system consists of two phases,

- Biometrics Detection Phase
- Biometric Recognition Phase

Both two phases are responsible for a set of agents and agent behaviors that generate the optimal values by message passing. Figure 4.1 shows an overview of the proposed system in both phases.

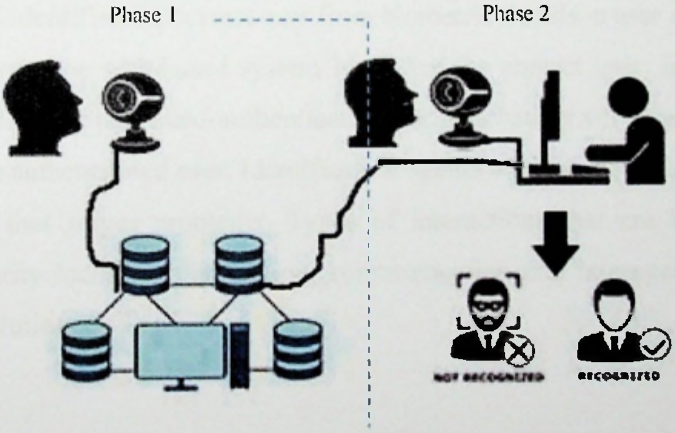


Figure 4.1: Overview of the proposed system

4.5.1. Agent Types

There are two types of agents such as biometric detection and recognition agents. Both agents are responsible for different tasks and following listed are the main agent types available in the BMAgent system.

Face detect and crop agent - This agent initializes the whole process by taking three images of the face and cropped them for future reference for recognition agents. There are three attempts to execute and collect the images while both phases. The output of the agent is a message with ready the other agents in the system to recognize the face at any time.

Colour separation agent - This agent locates the facial region based on the statistical distribution and thresholding of the skin color. Knowledge of facial trends (distribution of non-skin sub-regions) is then used within these regions to determine facial instances. The agent divides various convex objects using morphological operators, removes too small regions and recovers the sizes of regions while maintaining the same topological structure.

Skin area detection agent - This agent locates the facial region by comparing the skin regions with a model and eliminates areas of the skin that do not fit facial regions such as hands. MAS identifies the correct user from biometric details, a user enters the user id and passwords, the web-based system identifies the correct user, both biometric authenticated user and password-authenticated user matched by verification agent and finally map the authenticated user. Identification agents work as multiple perspectives of a situation that solves problems. Types of interactions that are best suited to biometric security include collaboration, communication and interagent negotiation and the best solution.

Feature extraction agent – This agent acquires the three images from detectandcorp agent and applies the PCA algorithm and compares it with input images at the 2nd phase by calculating the distance function as discussed in chapter 3.

Fingerprint agent - This agent captures the figure print and store it in the cloud server and compare with given input at the 2nd phase. The final output is the message with distance value.

Authorize agent - This agent tallies the distance values of facial recognition agents and fingerprint agent using the final message at each itinerary. This agent was responsible for the final output of the system with a confidence value. After initiating 3 attempts of this agent other agent's processes are killed by the agent manager and conclude the session.

4.6. Features

The proposed system has been designed for on-time access for time-critical applications such as border control. The BMAgent solution gives the most accurate results than conventional or multi-mode systems. The system can execute in normal computers and less power consumption, cost-effectiveness and user-friendliness are an adequate advantage.

Listed below are the main features of the BMAgent system:

- Automated system for biometric authentication
- Real-time updating system and smooth adaptability of the traditional system
- Can introduce new biometric levels in future
- Providing a user-friendly interface to interact with the system and provide proper feedback to the user regarding the decisions taken by the system.

4.7. Users

The proposed BMAgent system can be utilized by different types of users such as:

- Immigrants and emigrants
- VISA offices
- Immigration offices
- Airport authorities

4.8. Summary

In this section, the most important part of this thesis was discussed consisting of the hypothesis, basic inputs, main outputs, identified users such as the planners, supervisors and labors, main processes and features of the proposed BMAgent system.

etc. In the upcoming chapters, these details will be elaborated and discussed in-depth. Accordingly, in the next chapter, design details of the BMAgent system are discussed including the high-level architecture of the system, class and database diagrams as well as the novel concepts introduced in the BMAgent system.

Chapter 5

Design of the Multi-Agent-Based BMAgent System

5.1. Introduction

In this chapter, the design of the proposed BMAgent system will be discussed in detail. Initially, the higher-level architecture of the system will be discussed and afterward, the main design details considered for the evaluation of biometrics when authorization will be elaborated. Later, the Class Diagram of the system will be illustrated and finally, the novel concepts introduced in the BMAgent system will be detailed with regards to conventional and existing Multi-Agent based approaches for biometric authorization.

5.2. The architecture of the System

The following figure 5.1 displayed is a high-level architecture of the proposed system. Two phases of the system depicted as one architecture:

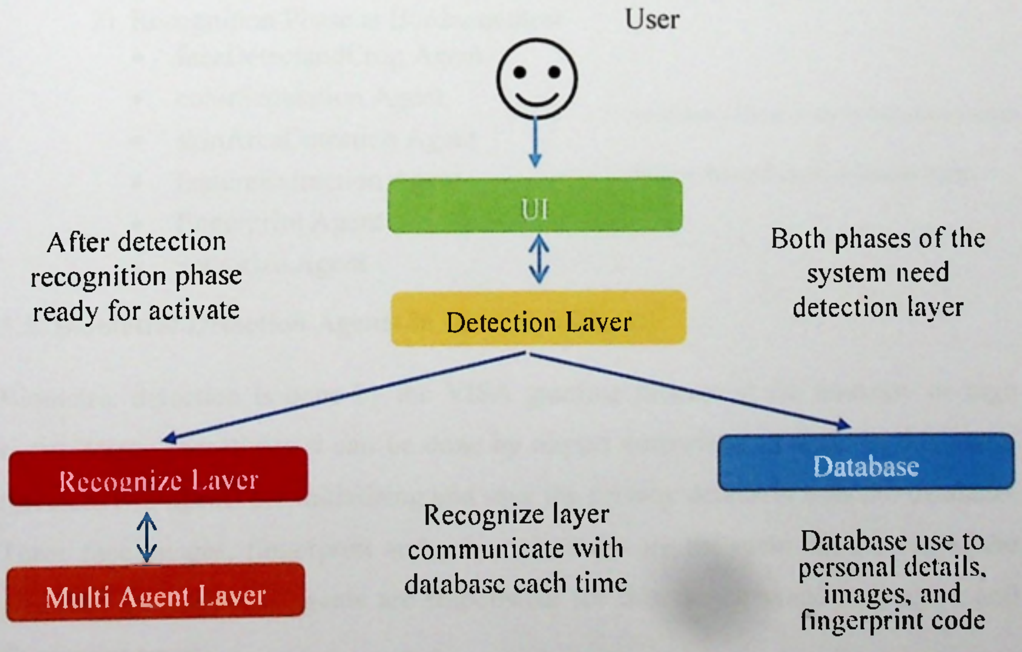


Figure 5.1: High-level architecture of the System

Figure 13 illustrates how the users can interact with the system through the User Interface and perform various functions such as inserting face biometrics, inserting fingerprint, inserting personal details, view the recognition result and the instructions to the next step, etc. The UI is developed thin and loosely coupled from the detection layer.

The Detection layer as the name implies, acts as a middle layer and handles both phases and communicates with the database. In the detection phase this layer store the images in the database and in the recognition phase this layer act as initializing the phase. The database is currently managed by a relational database management system using MySQL, where the relevant models are maintained. Finally, the BMAgent system works as the backbone of the main system, which comprises of a Multi-Agent component. It contains agents who access a common ontology of knowledge, rules, and practices of constraint-based procedures. The agents consist of types such as:

1) Detection Phase at VISA granting process
   - faceDetectandCrop Agent
   - fingerprint Agent
   - dataCollection Agent

2) Recognition Phase at Border control
   - faceDetectandCrop Agent
   - colorSeperation Agent
   - skinAreaDetection Agent
   - featureExtraction Agent
   - fingerprint Agent
   - authorize Agent

Appearance Based Face Recognition Agents

Feature Based Face Recognition Agents

5.3. Biometric Detection Agents in Detection Phase

Biometric detection is done by the VISA granting process at the embassy or high commission. Sometimes it can be done by airport authorities as well. In this phase there are two agents are initializing and save the privacy details in a secure database. Three face images, fingerprint and personal details are the main ingredients of the proposed system. Two agents are responsible for this faceDtectandCrop agent and fingerprint agent.

5.3.1. faceDetectandCrop Agent

This agent starts both phases and acts as detect the face from the given image. After the detect the face from the given image it will crop and save in the database. It used a famous face detection algorithm. In 2001, Paul Viola and Michael Jones have introduced a technique to object detection known as "Viola-Jones object detection technique". This is known as the first object detection technique and possible to execute in real-time. There are two object classifiers face object and eye object. Figure 5.2 shows the high-level architecture of face detection and crop agent.

Properties

- image size (300px * 300px)
- passport id
- index id
- sequence number

Functions

- initialize both phase and detect the face from the given image
- crop and save in database size (300px * 300px)
- take 3 images at the detection phase
- identify the main face and remove others

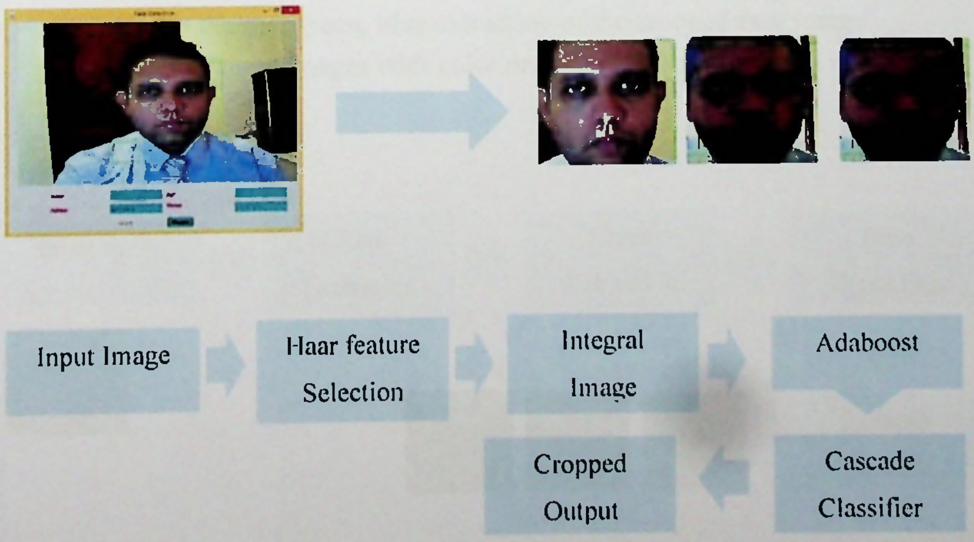


Figure 5.10: High-level architecture of the facedetectandcrop agent

5.4. Biometric recognition agents in Recognition phase

Captured biometric recognition is done by this agent. This agent is activated at the border control process in the proposed automated system. The agent acquires a common database to match the projection of face image and fingerprint. There are few agents are responsible for this such as colorSperation Agent. skinAreaDetection Agent, featureExtraction Agent, and fingerprint Agent.

5.4.1. colorSeperation Agent

This agent uses the cropped image and extract the red, green, blue colors from the face and store the three images. It uses a color extraction java code to detect the colors and store them. Figure 5.3 shows the high-level architecture of a color separation agents.

Properties

- Color-code
- Passport id
- Index id
- Sequence number

Functions

- Identify the red, green, blue extraction of the cropped face image
- Save it as three images with color prefix e.g red_xxxxx

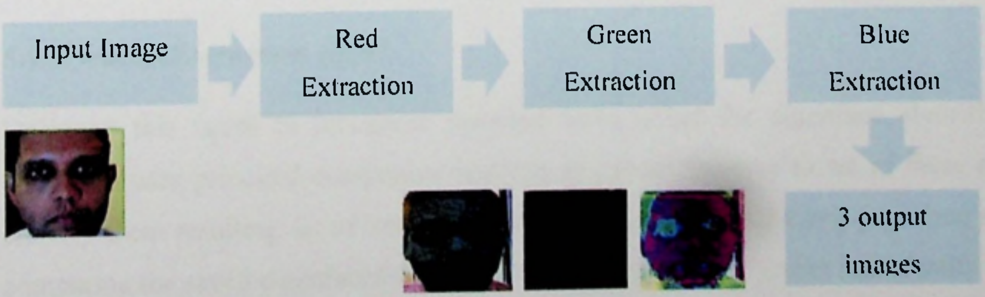


Figure 5.3: High-level architecture of the colorSperation agent

5.4.2. skinAreaDetection Agent

This agent used to separate skin area and non-skin area from the face image, this process repeating for optimal result. Skin detect using java algorithm. Figure 5.4 shows the high-level architecture of the skin region detection agent.

Properties

- Skin texture color code
- Passport id
- Index id
- Sequence number

Functions

- Identify the skin and non-skin area of the cropped face image
- Save it as three images

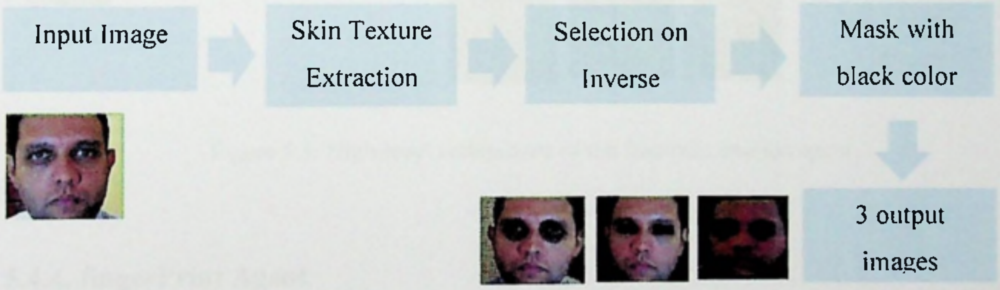


Figure 5.4: High-level architecture of the skinRegionDetection agent

5.4.3. featureExtraction Agent

Designed this agent to recognize detected faces using the eigenface algorithm. eigenface uses principal component analysis to extract features to set of faces and classify them resulting set of vectors called eigenfaces. By projecting your face and comparing the saved eigenfaces the algorithm can find the similarities and identity the face other than face detection 15 components and threshold 4000. Figure 5.5 shows the high-level architecture of the feature extraction agent and workflow.

Properties

- Threshold value
- Source image path
- Destination path
- Sequence number

Functions

- Convert the cropped image into black and white
- Generate eigenfaces and check the similarity with a projected face
- Result identity with distance value.

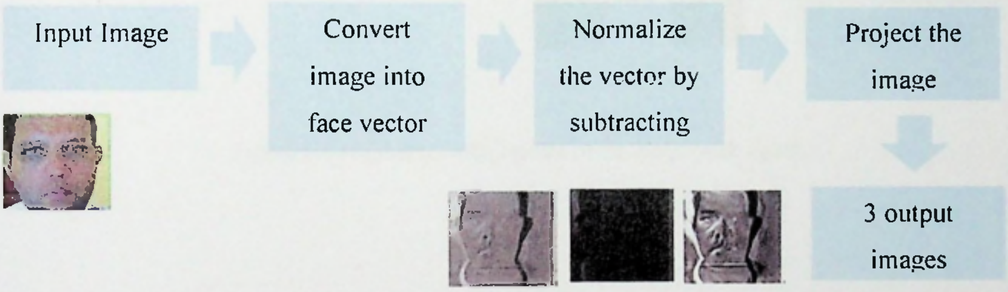

Input Image
Convert image into face vector
Normalize the vector by subtracting
Project the image
3 output images

Figure 5.5: High-level architecture of the featureExtraction agent

5.4.4. fingerPrint Agent

This agent recognizes the fingerprint pattern and makes classification on it. Finally, it possible to match the projected fingerprint. Figure 5.6 shows the high-level architecture of the fingerprint agents.

Properties

- Threshold value
- Source image path
- Destination path
- Sequence number

Functions

- Convert the cropped image into black and white
- check the similarity with projected fingerprint
- Result identity with distance value.

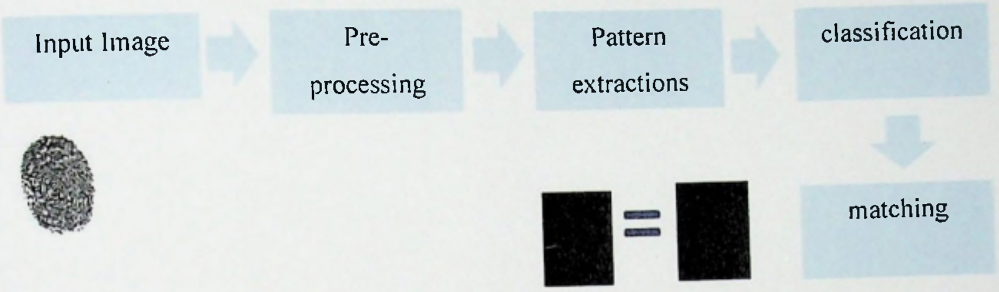


Figure 5.6 High-level architecture of the fingerprint agent

5.1. Class Diagram

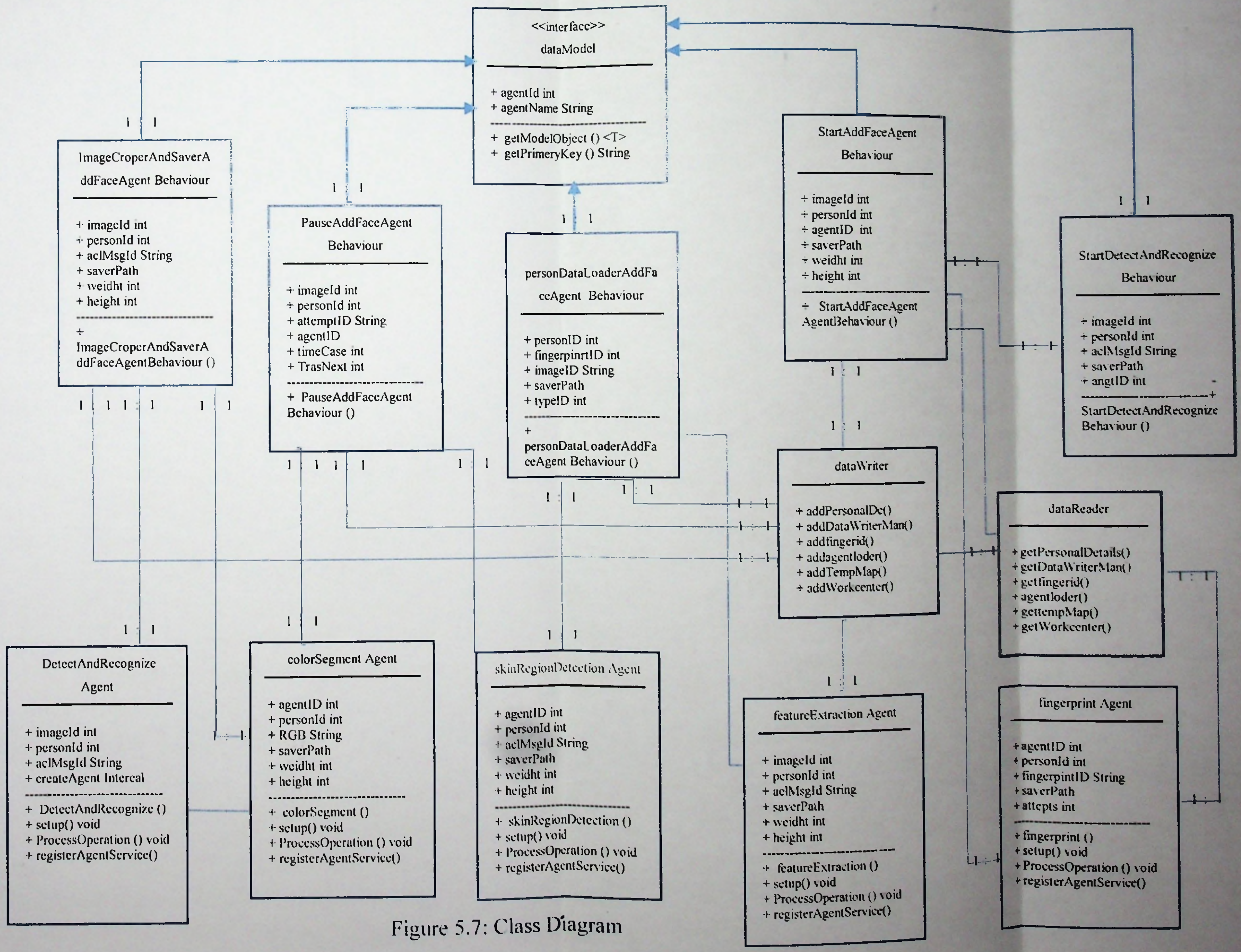


Figure 5.7: Class Diagram

5.4. Novel Concepts Introduced in the BMAgent System

The main concept can be identified as there is no such multi-agent systems develop with multi-model biometric levels in the past. There are two main novel features of the BMAgent system when compared to both conventional biometric authentication systems as well as available Multi-Agent based systems:

- Multiple processes of three face recognition algorithms such as color segmentation, skin region detection, and feature extraction together. The system can introduce a new algorithm as an agent in the future.
- Realtime authentication after the execution of phase one which already captured the face, fingerprint and personal details of the candidate.
- Biometrics can be updated or maintained by the proposed automated system with any border points in a single or multiple tour itinerary.
- A semi-automated system that can mutually handshake the VISA granting process and recognition process of border points.
- The system calculates the upper limit of a particular operation start date known as the ***Latest authentication Start Date (LASD)*** and the authentication Agent greedily tries to start a given operation on or before that date by negotiating with an authorized Agent. As long as the authentication Agent can schedule operations by meeting that point of time, it will keep authentication the subsequent operations, however, as soon as one operation fails to meet its LASD, all the subsequent operations from that operation will also fail to meet their respective LASD hence the authentication Agent will stop authentication from that operation onwards and try to unscheduled lower priority biometrics and continue the authentication process yet again.

- Each authentication is given an "importance" score according to its agent priority and the revenues that are generated by them, using a weighted average method, which allows the airport organization to prioritize them. These scores are then used for Prioritized Adaptive authentication of the generated schedule when a disruptive event occurs; these scores are evaluated by the Authorize

Agent and only the orders that have received a lower priority score than the current authentication and is closest to the current authentication will be used in the chain event, and using a cyclic approach, the system will try to come to an equilibrium as soon as possible with the least amount of biometric levels affected when dynamically authentication.

- **Advanced negotiation mechanism**: This negotiation mechanism is used by the main Agents associated with the BMAgent system, which provide a highly dynamic and scalable approach to authentication. There, as mentioned previously, the authentication Agent uses the system calculated LASD in order to keep the authentication on track when engaging in the advanced negotiation mechanism with different authorize Agents.

5.5. Summary

In this chapter the design of the system was discussed by emphasizing the Architecture of it in terms of both a higher-level as well as a lower-level. indicating how the different agents are designed and their functionalities, etc. In addition, the novel concepts introduced in the BMAgent system compared to existing dynamic authorization systems as well as conventional authorization systems are also discussed. Furthermore, the design of the system has also been illustrated using different diagrams such as the Class Diagram and the Database Diagram

Chapter 6

Implementation of the BMAgent System

6.1. Introduction

In this chapter, the implementation particulars of the system are discussed elaborately. Initially, the different types of agents available in the system are discussed such as the detection agents and recognition agents. Afterward, the implementation details of the system will be discussed exhaustively using flow charts, sequence diagrams, etc. for different functions of the system such as the initial authorization process and the Prioritized-Adaptive authorization process when parts are unavailable, or work centers are interrupted.

6.2. Implementation of Biometric Detection agents

In this phase, there are few agent behaviors are responsible for capturing the face image and fingerprint. All cropped face images are saved with 300 * 300 px weight and height respectively and this phase executed at the VISA granting process.

Step 1 - faceDetectandCrop Agent - this agent starts the form AddData_FaceDetection and make it visible to end user. This agent has 4 main behaviours added on it

- StartAddFaceAgentBehaviour,
- PauseAddFaceAgentBehaviour
- PersonDataLoaderAddFaceAgentBehaviour,
- imageCroperAndSaverAddFaceAgentBehaviour

Step 2 - startAddFaceAgentSender this agent sends an ACL message with content "StartAddFace" to AddDataFaceDetection Agent Behavior, based on this message AddDataFaceDetection Agent Behavior will run the cyclic behavior StartAddFace Agent Behavior which will open the camera and make the end-user able to capture his photo.

Step 3 - Then user enters the personal data related with face image and press "Pause" button, this is will fire the agent called PauseAddFaceAgentSender behavior which will send ACL message to AddDataFaceDetection Agent and based on that run

PauseAddFaceAgent behavior, this behavior will start a new agent called PersonDataLoaderAddFaceAgentSender which will send ACL message to AddDataFaceDetectionAgent with content "LoadPersonData", based on this message the PersonDataLoaderAddFaceAgent behavior will be invoked and read the form data. build person data object. In the end, PersonDataLoaderAddFace Agent behavior starts a new agent called ImageCroperAndSaverAddFace Agent Sender which sends ACL message to AddDataFace Detection Agent "CropAndSaveFaceImage" based on this message, the AddDataFace Detection Agent invokes behavior ImageCroperAndSaver AddFace Agent behavior which crops the face image and saves it with the associated person data object.

6.2.1. faceDetectandCrop Agent

Displayed in Figure 6.1 is a flowchart for the faceDetectandCrop Agent. This agent stat with webcam input and capture frame. It will go through a HAAR cascade feature and create two object classifiers one for face and another for eyes. It will act as an optimal way to select the face from the given image.

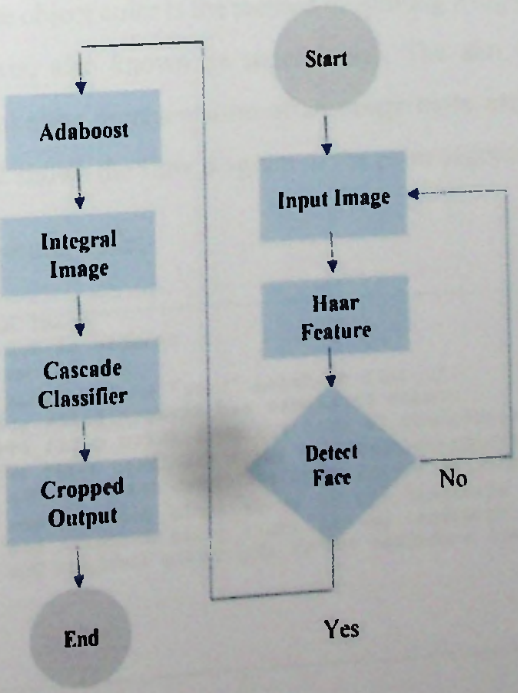


Figure 6.1: faceDetectandCrop Agent Providing the optimal area of the face from the given image

6.3 Implementation of Biometric Recognition agents

Step 1- This agent DetectAndRecognize Agent starts the form Detect and Recognize and make it visible to the end-user.

Step 2 - Then start the agent "StartDetect and Recognize Agent" this agent will fire the action of the start "Detect and Recognize" button and start real-life detection.

Step 3 - Next, start ClearFieldsDetect and Recognize Agent which related to cyclic behavior, the role of this agent keeps text fields empty when the system is not able to recognize the face image.

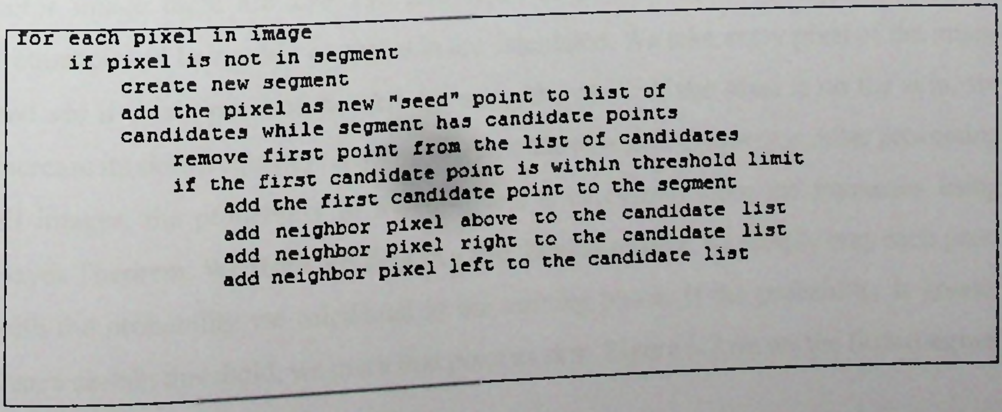
Step 4 - Then start ReteriveFieldsDetect and Recognize Agent. this agent retrieves data related to the detected face from all_data map.

Step 5 - This agent DetectAndRecognizeTrainData Agent is responsible for building face data, bind the personal information with face images.

6.3.1. colorSegment Agent

Segmentation of the object color is the method of splitting a digital image into multiple segments (pixel sets, also known as superpixels). The aim of segmentation is to simplify and/or make the representation of an image more meaningful and easier to analyze. Figure 6.2 shows the flow diagram of the color segmentation agent.

The algorithm in pseudo-code:



```
for each pixel in image
    if pixel is not in segment
        create new segment
        add the pixel as new "seed" point to list of
        candidates while segment has candidate points
            remove first point from the list of candidates
            if the first candidate point is within threshold limit
                add the first candidate point to the segment
                add neighbor pixel above to the candidate list
                add neighbor pixel right to the candidate list
                add neighbor pixel left to the candidate list
```

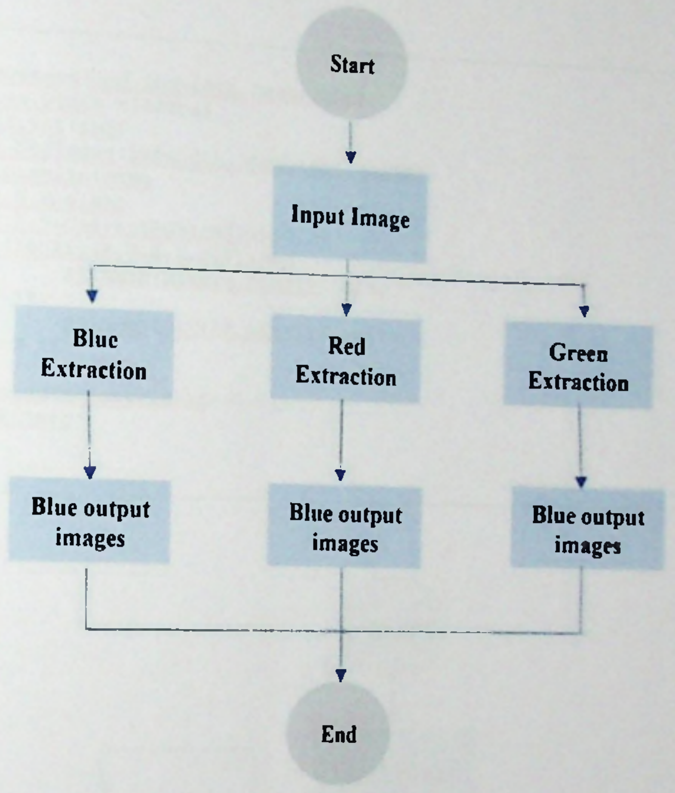
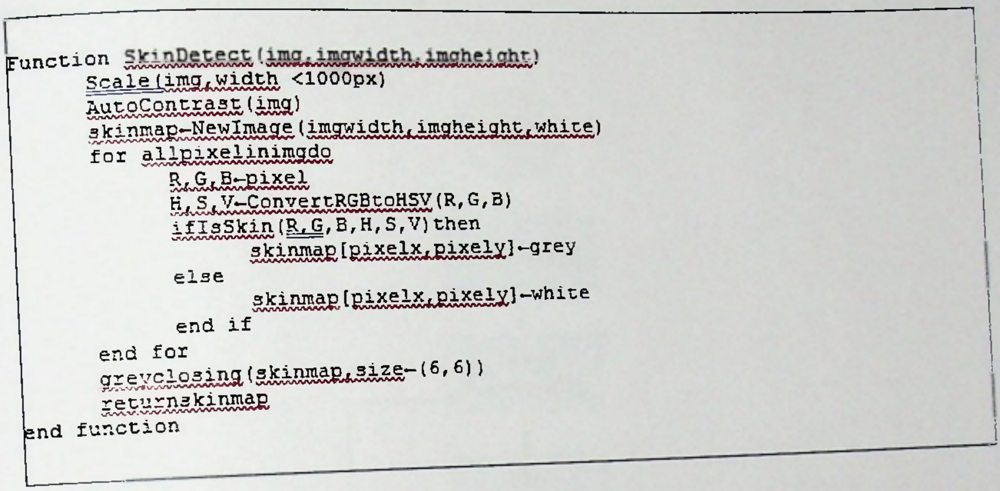


Figure 6:2: colorSegment Agent flowchart

6.3.2. skinRegionDetection Agent

We have used naive Bayes here for classification (skin or non-skin pixel). As it is a color image there are 256*256*256 types of pixels. In the training phase, pixel frequencies of being skin or non-skin are calculated. We take every pixel of the image and see if it is a pixel of the skin by using the mask. If the pixel is on the skin, we increase its skin-frequency. Else we increase the non-skin-frequency. After processing all images, the probability of a skin-pixels is calculated from the frequency using Bayes Theorem. We store this data in a file. During testing, we simply map each pixel with the probability we calculated in the training phase. If the probability is greater than a certain threshold, we mark that pixel as skin. Figure 6.3 shows the flow diagram of the skin region detection agent.

The algorithm in pseudo-code:



```
Function SkinDetect(img,imgwidth,imgheight)
      Scale(img,width <1000px)
      AutoContrast(img)
      skinmap←NewImage(imgwidth,imgheight,white)
      for allpixelinimgdo
            R,G,B←pixel
            H,S,V←ConvertRGBtoHSV(R,G,B)
            ifIsSkin(R,G,B,H,S,V)then
                  skinmap[pixelx,pixely]←grey
            else
                  skinmap[pixelx,pixely]←white
            end if
      end for
      greyclosing(skinmap,size←(6,6))
      returnskinmap
end function
```

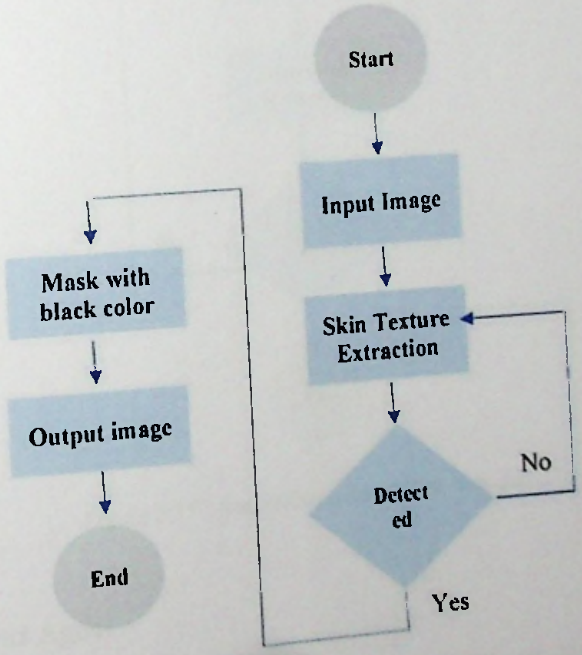


Figure 6:3: skinRegionDetection Agent flowchart

6.3.3. featureExtraction Agent

Using the Principal Component Analysis (PCA) model for face recognition. PCA is a computational technique used to reduce face recognition variables. Each image in the training set is represented in PCA as a linear combination of weighted own vectors called individual faces. Figure 6.4 shows the flow diagram of the feature extraction agent.

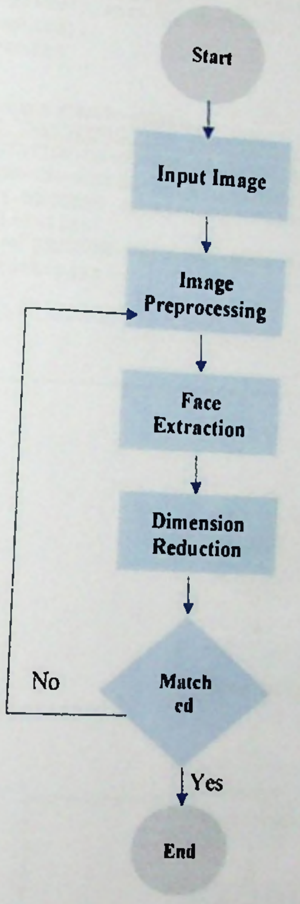
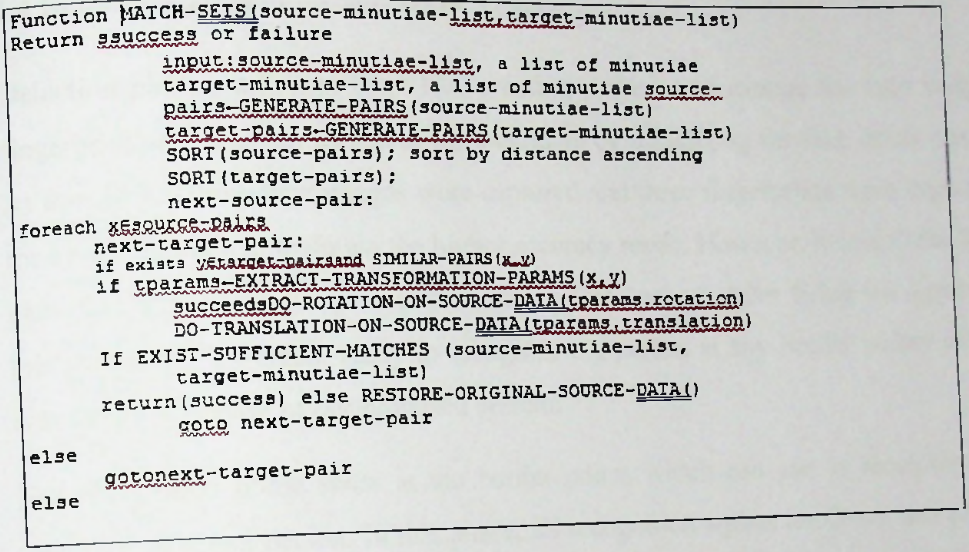


Figure 6:11 : featureExtraction Agent flowchart

6.3.4. fingerprint Agent

SourceAFIS for Java is SourceAFIS ' pure Java port, an algorithm for human fingerprint recognition. It can compare two 1:1 fingerprint or check for large 1: N fingerprint matching. It takes input fingerprint images and produces an output similarity value. The similarity score is then compared to a game threshold that can be adjusted. SourceAFIS is a fingerprint recognition system that takes a couple of human fingerprints and returns the similarity value. It can do 1:1 comparison as well as efficient 1: N search. This is the Java implementation of the SourceAFIS algorithm. Figure 6.5 shows the flow diagram of the fingerprint agent.



```
Function MATCH-SETS(source-minutiae-list,target-minutiae-list)
Return ssuccess or failure
            input:source-minutiae-list, a list of minutiae
            target-minutiae-list, a list of minutiae source-
            pairs←GENERATE-PAIRS(source-minutiae-list)
            target-pairs←GENERATE-PAIRS(target-minutiae-list)
            SORT(source-pairs); sort by distance ascending
            SORT(target-pairs);
            next-source-pair:
foreach x∈source-pairs
       next-target-pair:
       if exists y∈target-pairsand SIMILAR-PAIRS(x,y)
       if tparams←EXTRACT-TRANSFORMATION-PARAMS(x,y)
             succeedsDO-ROTATION-ON-SOURCE-DATA(tparams.rotation)
             DO-TRANSLATION-ON-SOURCE-DATA(tparams.translation)
       If EXIST-SUFFICIENT-MATCHES (source-minutiae-list,
             target-minutiae-list)
       return(success) else RESTORE-ORIGINAL-SOURCE-DATA()
             goto next-target-pair
 else
       gotonext-target-pair
 else
```

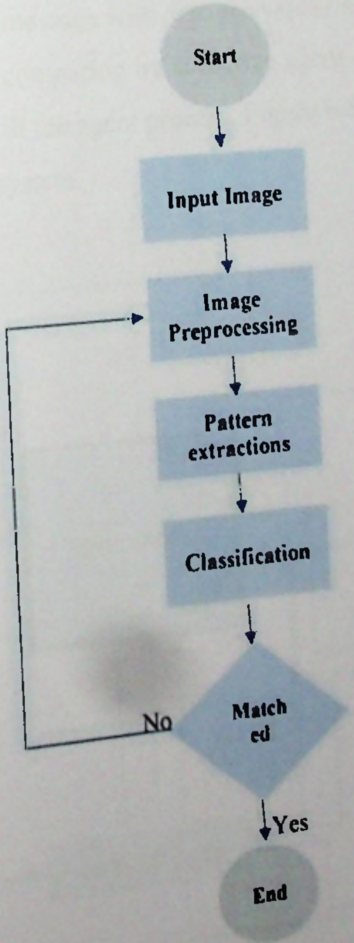


Figure 6.5: fingerPrint Agent flowchart

6.4 Implementation of the BMAgent System

Detection phase starting at the VISA granting process and capture the face image. fingerprint and personal details of the candidate by initializing the face detect agent. At this phase, three face images were captured and three fingerprints were captured for each candidate to maintain the higher accuracy result. However, it could take data gathering time when comparing to the traditional method. After firing the agents at this phase, the BMAgent ready to recognize the person at any border points which describe in 2nd phase of our proposed system.

The recognition phase starts at thc border points which can use to recognize the candidate as a real person. In this phase, all recognition agents are firing and pass a message with distance values to the agent manager. In addition to that fingerprint agent firing and matching and pass a message with distance value on order implement higher accuracy. Finally, values are comparing by authorize agent at threshold literary and take the optimize values and kill the agent process. Figure 6.6 shows the overall agent architecture of the BMAgent system.

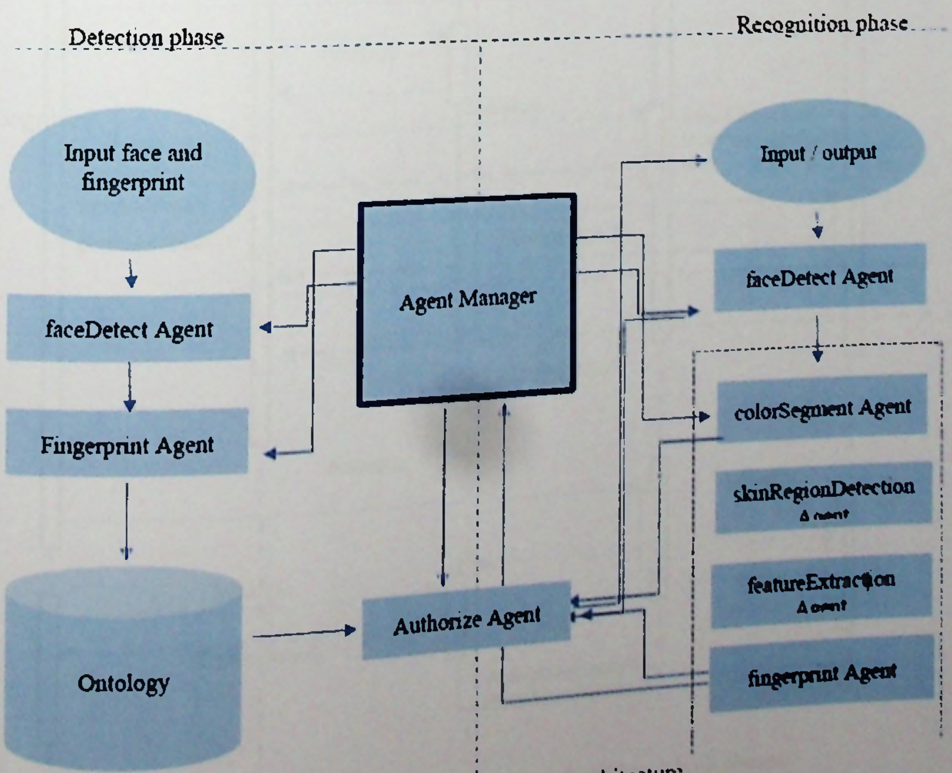


Figure 6:6: Overall Agent Architecture

Figure 6.7 shows a sequence of diagrams of the agent interaction that happens during this process:

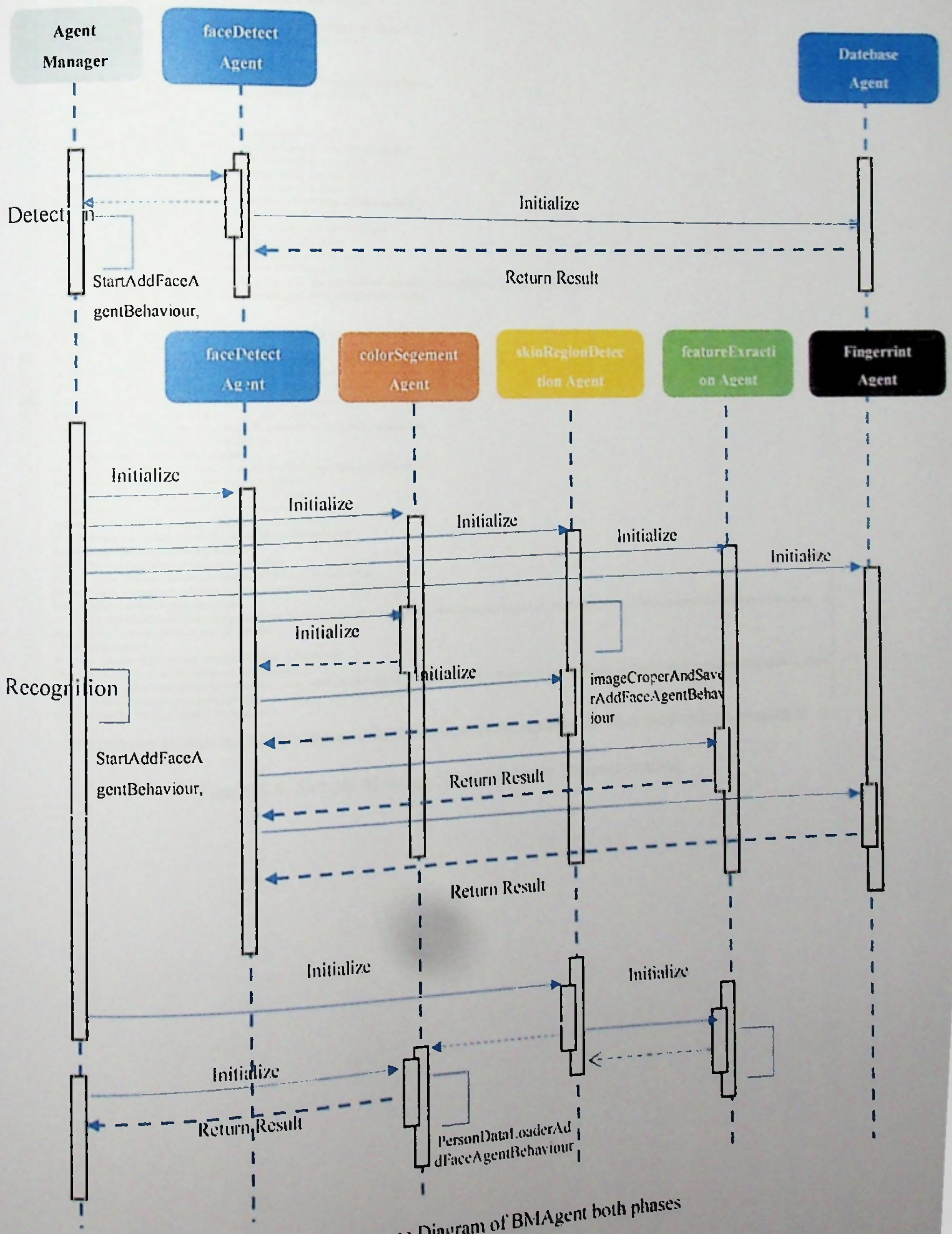
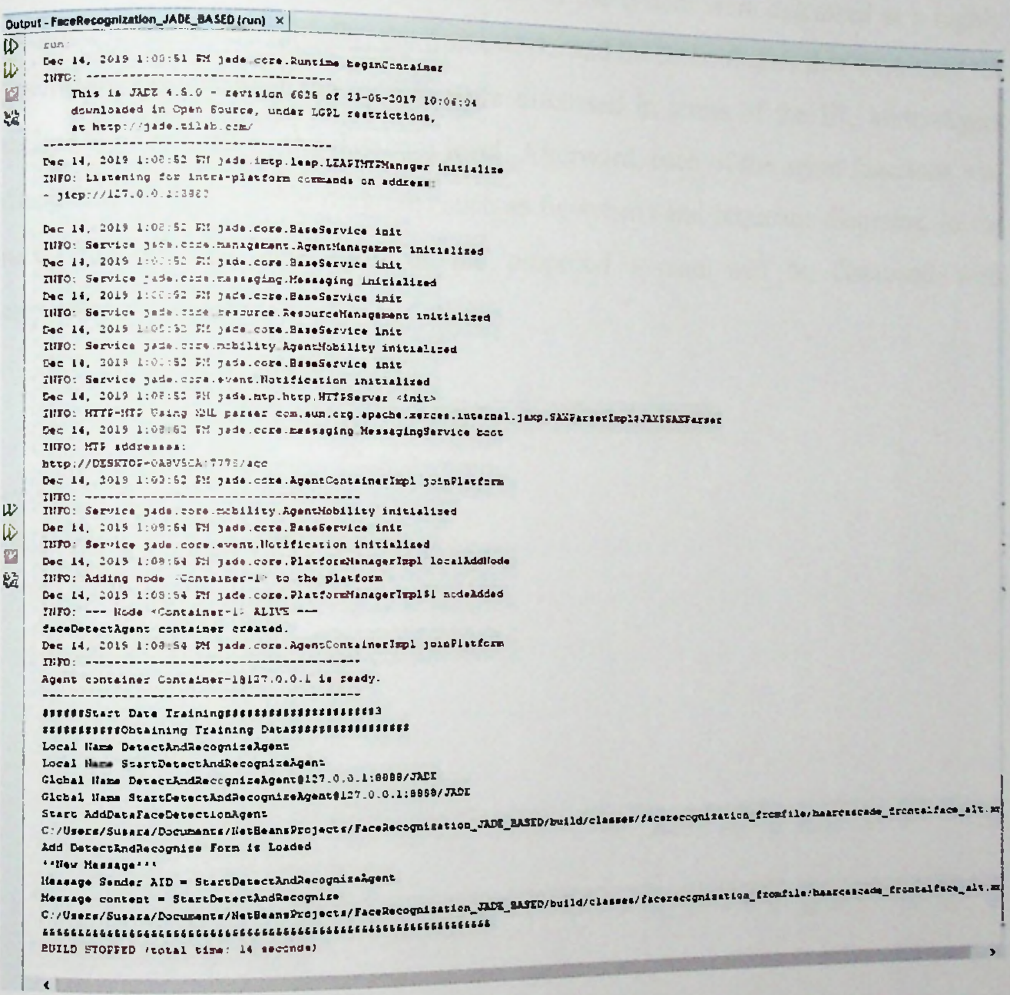


Figure 6.7: Sequence Diagram of BMAgent both phases

Figure 6.8 shows a sample message space of the communication that happens between the recognition Agent and the authorize Agents when initially in phase 2.



```
run:
Dec 14, 2019 1:08:51 PM jade.core.Runtime beginContainer
INFO: ----------------------------------
    This is JADE 4.5.0 - revision 6825 of 23-05-2017 10:06:04
    downloaded in Open Source, under LGPL restrictions,
    at http://jade.tilab.com/
----------------------------------------
Dec 14, 2019 1:08:52 PM jade.imtp.leap.LEAPIMTPManager initialize
INFO: Listening for intra-platform commands on address:
- jicp://127.0.0.1:3382

Dec 14, 2019 1:08:52 PM jade.core.BaseService init
INFO: Service jade.core.management.AgentManagement initialized
Dec 14, 2019 1:08:52 PM jade.core.BaseService init
INFO: Service jade.core.messaging.Messaging initialized
Dec 14, 2019 1:08:52 PM jade.core.BaseService init
INFO: Service jade.core.resource.ResourceManagement initialized
Dec 14, 2019 1:08:52 PM jade.core.BaseService init
INFO: Service jade.core.mobility.AgentMobility initialized
Dec 14, 2019 1:08:52 PM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
Dec 14, 2019 1:08:52 PM jade.mtp.http.HTTPServer <init>
INFO: HTTP-MTP Using XML parser com.sun.org.apache.xerces.internal.jaxp.SAXParserImpl$JAXPSAXParser
Dec 14, 2019 1:08:52 PM jade.core.messaging.MessagingService boot
INFO: MTP addresses:
http://DESKTOP-OA8VSCA:7778/acc
Dec 14, 2019 1:08:52 PM jade.core.AgentContainerImpl joinPlatform
INFO: --------------------------------------
INFO: Service jade.core.mobility.AgentMobility initialized
Dec 14, 2019 1:08:54 PM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
Dec 14, 2019 1:08:54 PM jade.core.PlatformManagerImpl localAddNode
INFO: Adding node <Container-1> to the platform
Dec 14, 2019 1:08:54 PM jade.core.PlatformManagerImpl$1 nodeAdded
INFO: --- Node <Container-1> ALIVE ---
faceDetectAgent container created.
Dec 14, 2019 1:08:54 PM jade.core.AgentContainerImpl joinPlatform
INFO: --------------------------------------
Agent container Container-1@127.0.0.1 is ready.
--------------------------------------------
######Start Date Training######################3
###########Obtaining Training Data##################
Local Name DetectAndRecognizeAgent
Local Name StartDetectAndRecognizeAgent
Global Name DetectAndRecognizeAgent@127.0.0.1:8888/JADE
Global Name StartDetectAndRecognizeAgent@127.0.0.1:8888/JADE
Start AddDataFaceDetectionAgent
C:/Users/Susara/Documents/NetBeansProjects/FaceRecognization_JADE_BASED/build/classes/facerecognization_fromfile/haarcascade_frontalface_alt.xm
Add DetectAndRecognize Form is Loaded
**New Message***
Message Sender AID = StartDetectAndRecognizeAgent
Message content = StartDetectAndRecognize
C:/Users/Susara/Documents/NetBeansProjects/FaceRecognization_JADE_BASED/build/classes/facerecognization_fromfile/haarcascade_frontalface_alt.xm
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
BUILD STOPPED (total time: 14 seconds)
```

Figure 6.8: Sample Message Space of Agent Communication

6.5 Summary

In this chapter, the implementation details of the system were discussed at a highly detailed level. There initially, the frameworks and the technologies that were used for the implementation of the system were discussed in terms of the UI, Multi-Agent technology as well as the database level. Afterward, each of the agent functions was discussed in depth using diagrams such as flowcharts and sequence diagrams. In the next chapter, the evaluation of the proposed system will be discussed with experimental results.

Chapter 7

Evaluation of the BMAgent System

7.1. Introduction

The entire system runs on the JADE framework as it can easy to develop and manage the agent behaviors in a comprehensive manner [20]. The proposed system has evaluated and compared with the traditional system and tested with 800 (2400 face images) candidates who were applying for a VISA for Schengen countries. Each candidate gives 3 images; therefore, the overall testing sample is 2400 images. All face images (300*300px) were captured by the same lighting condition and a very less noise environment. Three face images are captured from each individual for training the system. This has been done by agents in the face recognition phase with the help of OpenCV. OpenCV has been optimized to provide algorithmic efficiency especially for the processing of real-time programs. Figure 32 in Appendix A shows the user agreement before collecting the biometric and personal details from candidates.

7.2. Experimental Design

As for the experimental design, the developed BMAgent system is compared to an existing eborder system, which utilizes both a manual and dynamic authentication process, in order to verify the accuracy, effectivity and the efficiency of the BMAgent system.

7.2.1. Measurements

When evaluating the developed BMAgent system, the following listed are the different criteria that will be taken into consideration.

True Positive (TP) : True person is identified as True (correct identification)

True Negative (TN) : False person is identified as False (correct identification)

False Positive (FP) : True person is identified as False (wrong identification)

False Negative (FN) : False person is identified as True (wrong identification)

Detection Rate and Recognition Rate:

Face detection involves finding whether or not there are faces in a given picture (usually in grayscale) and returning the position and content of each face when present.

$$\% \, Detection\ Rate = \frac{True\ Positive}{Total\ No.of\ images}\ x\ 100 \qquad (7,1)$$

Precision or Positive Predictive Value (PPV):

Precision is the fraction of the relevant instances among the instances retrieved, thus retaining the fraction of the total number of instances actually retrieved. Accordingly, both specificity and recall are based on an awareness and relevance scale.

$$\% \, Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}\ x\ 100 \qquad (7,2)$$

False Detection Rate or False Recognition Rate:

The False Discovery Rate (FDR) is a way of conceptualizing the rate of type I errors when making multiple comparisons in null hypothesis testing. FDR-controlling procedures are intended to control the predicted proportion of false (incorrect rejections) "discoveries" (rejected null hypotheses).

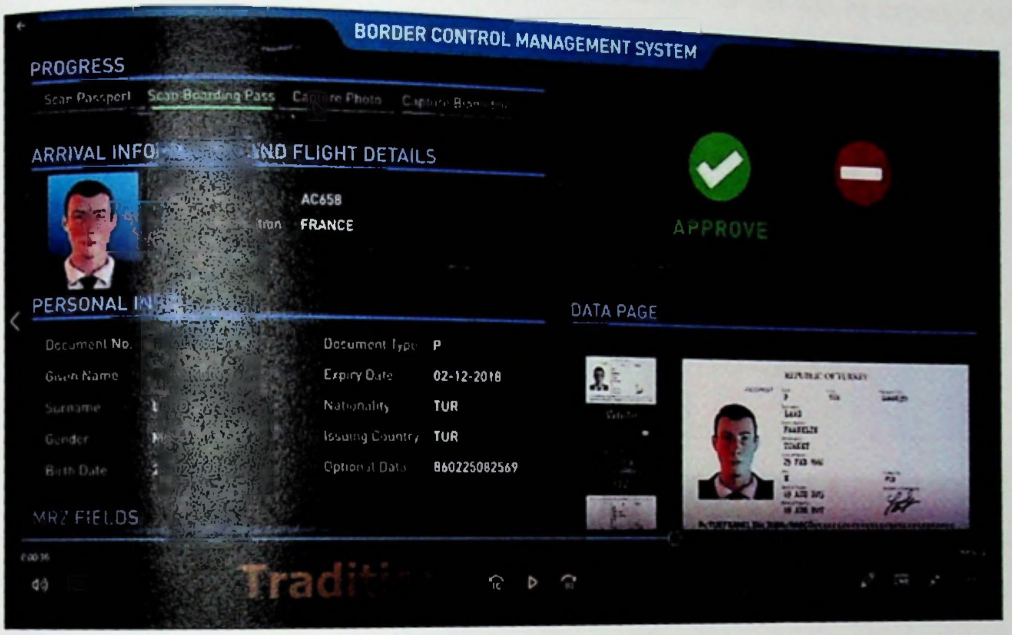
$$\% \, False\ Detection\ or\ Recog\ Rate = \frac{False\ Positive}{Total\ No.of\ images}\ x\ 100 \qquad (7,3)$$

Average Time is taken for the authentications Process:

Time taken for the authentication is another important indicator that should be considered when comparing the BMAgent system to an existing adaptive eboder system algorithm.

7.2.2 The system considered for Evaluation

The system that is used for the evaluation purpose of the BMAgent system is a conventional border control system management (eborder application) which is currently working in border control points. It consists of manual and dynamic control as shown in figure 7.1.



AC658

from FRANCE

Document No.

Given Name

Surname

Gender

Birth Date

Document Type P

Expiry Date 02-12-2018

Nationality TUR

Issuing Country TUR

Optional Data 860225082569

Figure 7.1: Traditional eborder software interface

Source: http://eborder.com

7.2.2. Data Set

In order to evaluate the BMAgent system, as mentioned, a test dataset has been taken from the high commission with permission and modified it to fit the data model of the BMAgent system. Table 7.5 lists the details of the dataset used to evaluate the BMAgent system:

Table 7.2 : Test Dataset Details

Dataset Details	
No. of Approved Candidates	800
No. of Rejected Candidates	176
No. of Images	800 x 3 = 2400
Time period	06/JAN/2019 – 01/DEC/2019
Average Biometric Capture time per person	35 min

7.3 Evaluation Strategy

For the evaluation of the BMAgent system, the modified dataset will be imported to the eborder system database and BMAgent system and will run the initial authentications process. Both the recognition phase starting from the same time and calculate the average time and true positive, false negative, true negative and false negative. Afterward, the conventional system will be compared to the BMAgent system using the previously introduced metrics.

7.4 Experimental Results

The above dataset was replicated in the BMAgent system and the initial authentication algorithm was executed. Afterward, traditional border control system management software initializes in parallel in a separate location and even the lighting environment at embassy premises. Every attempt manually recording for evaluation purposes with the necessary permission of the candidates.

7.4.2 BMAgent System Phase 1 Experimental Results

Viola-Jones method was used to detect the image and cropping in detection and crop agent. After face detection sample 12 results as follows in figure 29. This result shows 300px *300px images in greyscale mode which can use for recognition agents. During this phase, other personal details have been collected and stored in the MySql cloud database. Figure 7.2 shows the after cropped and saved images (intermediate step).

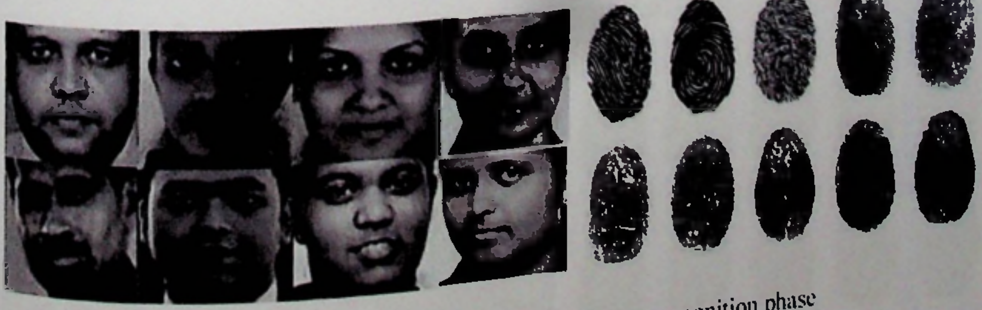


Figure 7.2: Intermediate test sample at the recognition phase

T= Traditional, M= Multi-agent, TP- True Positive, FP- False Positive, TN- True Negative, FN- False negative.

Table 7.1: Phase 1 biometric capture results

	Total	TP	FP	TN	FN	Detection Rate (%) TP/TOTAL	Precision (%) TP/TP+FP	False Detection FP/TOTAL
T	2400	1510	330	280	280	62.91	82.06	13.75
M	2400	2010	120	150	120	83.75	94.34	5

Table 7.1 shows a higher number of detection rates using MAS than a traditional system. In the detection phase, both traditional and proposed systems are used the same algorithm but in a different way. Therefore, the detection rate higher than traditional. Also, the authors are not expecting any improvements in the detection phase because it comes from the same algorithm. In practice, this has been done by the VISA grading process, which has even lighting conditions and minimizes facial expressions. Authors have identified the reason behind the false rate due to the quality of webcam used in the practical scenario.

7.4.2. BMAgent System Phase 2 experimental results

After the detected and cropped, images are ready for the recognition and this phase can be identified as a major part of the authentication process. Recognition agents are used as face recognition and fingerprint verification most accuracy agents will emerge from the system with the help of authorized agents. Results and the intermediate face figures (eigenfaces) are shown in the below figure 7.3.

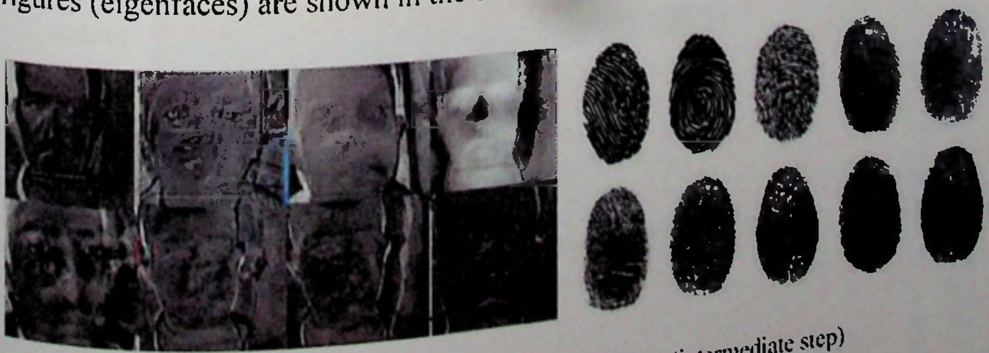


Figure 7.3: Eigenface images and fingerprints (intermediate step)

Table 7.2: Phase 2 biometric Recognition Results

	Total	TP	FP	TN	FN	Recognition Rate (%) TP/TOTAL	Precision (%) TP/TP+FP	False Recognition FP/TOTAL
T	2400	1410	330	380	280	58.75	95.91	13.75
M	2400	2110	120	150	20	87.91	94.61	5

Table 6 shows a higher number face recognition rate with the proposed system as predicted by authors 7.2. False-positive rates have identified the quality of webcam in practical.

Table 7.3: Average Time taken for face recognition

	Total images	The average time is taken for face recognition (ms)
T	2400	1230
M	2400	560

The system itself calculates the average time consumption for face recognition and Table 7.3 shows significant improvement in the proposed system. Figure 7.4 shows the fingerprint verse face biometric accuracy in traditional and multi-agent systems.

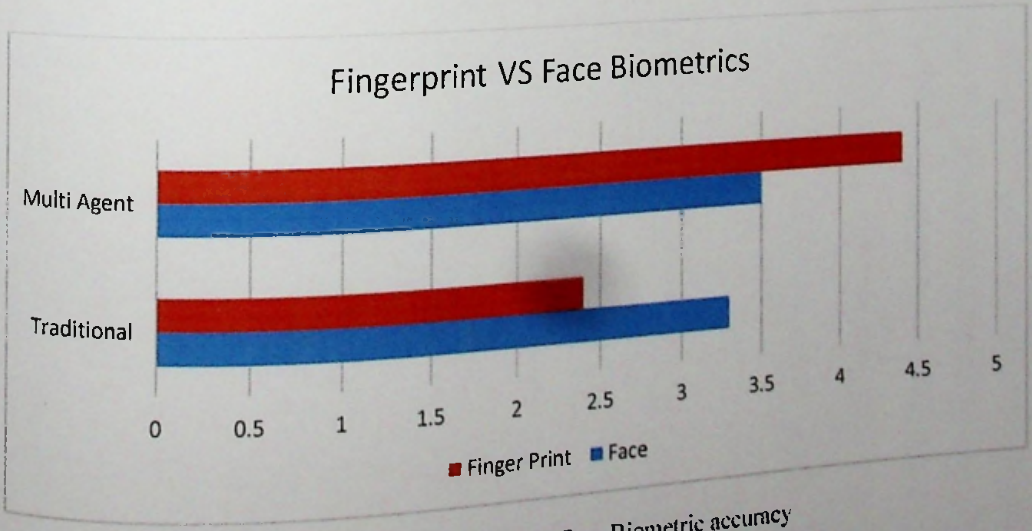


Figure 7:4: Finger Print and Face Biometric accuracy

7.5 Summary

In this chapter, it was discussed how the developed BMAgent system was evaluated thoroughly for its functionality as well as the efficiency that it provides for the potential authentication process. There, the experimental results were detailed using tables and diagrams which indicates clear performance improvements when using the BMAgent system. In the next chapter, these results will be discussed in detail and conclusions will be determined in terms of the performance of the system as well as this study in general. Finally, the limitations of the developed BMAgent system as well as the works available on the pipeline will also be discussed.

Chapter 8

Conclusion and Future Work

8.1. Introduction

In the previous chapter, the developed BMAgent system was evaluated using different measures, and this chapter will focus on the interpretation of those results and assess the performance of the BMAgent system accordingly. In addition, conclusions will be made on how the different objectives identified in the initial stages of the BMAgent system have been achieved throughout the timeline of the project. Afterward, it will deliberate on the limitations and the future works planned for the developed system.

8.2. Conclusion

This study tries to address the common problem of authentication at the border points using a novel Multi-Agent approach called BMAgent system, which is an extension to the existing authentication algorithms, alongside an advanced market-like negotiation mechanism between different kinds of Agents available in the identification context. As such, the BMAgent system has been developed and discussed in detail in the previous chapters of this thesis, also, it has been evaluated using different indicators such as the percentage of detection, percentage of recognition and average time consumption.

When comparing the results taken from the evaluation, it's apparent that the BMAgent system provides much better overall results when compared to the test dataset extracted from an existing border management system, which uses both manual and dynamic processing for authentication. The primary objective of this research was to provide an effective face biometric identification and authentication system to border protection. In the above section, the researchers have brought into the attention of the reader, how to implemented the system and the accuracy in time-critical application. Viola-Jones algorithm was used to detect the face. The color segmentation, skin region detection, feature extraction, fingerprint agents were used for the recognition phase. The overall system has been developed by MAS which can emerge solutions with coordination and co-operation. It can be identified as the novelty of the system. After the implementation, accuracy was tested with authentic biometric samples and compared with the traditional system using 800 candidates with 2400 face images. As observed in the experiment MAS gives a higher rate of accuracy in identification within a few seconds and GUI based BMAgent system illustrated in Appendix A. The objectives of the research study were achieved to a significant level. In the biometric capturing phase, face detection rates were higher than conventional system at 89% detection rate and codes illustrated in Appendix A.

In the biometric recognition phase, true positive rates are higher whilst false-negative rates were decreased which implied that 97% percent of candidates were recognized by the BMAgent system and codes illustrated in Appendix B. The BMAgent system participants are taken with their permission until evaluating the result and authors do not transparent the sensitive data to the public. The average time measure for the recognition phase at border control and it shows 560ms which less than the traditional system. This results in the system coming to equilibrium much sooner than expected with a minimum amount of operations being rescheduled. This results in a comparatively lower overall time taken for the system to complete the dynamic authentication process. However, when comparing the time taken to process operation, it is still quite high; 2 seconds compared to 1.4 seconds of the test system. This is an area where the BMAgent system can improve on in the future.

Overall, the BMAgent system provided excellent results when compared to the dataset extracted from the system which utilized a combination of manual and dynamic authentication for their day to day operations. This can mainly be attributed to the authentication conception introduced in the BMAgent system which makes the dynamic authentication process much more efficient and effective allowing the system to come into an equilibrium much quickly when a disruptive event occurs.

When it comes to the objectives identified in this study, a critical study on Multi-Agent systems and how it has been utilized in different real-world time-critical applications as well as the existing literature on how to use Multi-Agent based approach to solve the authentication problems including face biometric is available on the second chapter of this thesis. Moreover, a fully functional BMAgent system has been developed providing various functions, of which, the design and implementation details were discussed in the fifth and the sixth chapters of the thesis respectively. Afterward, a critical evaluation of the developed BMAgent system has been conducted and the details related to it are available in chapter seven.

8.3. Limitations of the System

There are a few limitations identified in the developed BMAgent system, which are listed below:

- ***Reduced per-operation performance***: as discussed previously, even though the system takes a considerably lower amount of time for the authentication process to be executed when considering the time taken per operation. it takes a higher amount of time than the system from which the test dataset was extracted. This is a limitation of the system that can be addressed in the future.

- ***Issues with parallel operations of biometric levels***: Since the authentication algorithm uses only a date from which the lower priority biometric levels' operations (which runs on the affected work center) should be unscheduled. it can result in multiple parallel operations from the authentication that runs through the mentioned date (in different border points) to be selected by the algorithm to be dynamically unscheduled, which is quite unnecessary. Even though this will not affect the overall authentication of the biometric levels, this can be a contributing factor for the performance gap discussed in the previous point.

8.4. Future Work

Following listed are the currently identified future works for the BMAgent system:

- ***Further improvement to the performance of the system***: Per-Operation performance of the system has been identified as a certain limitation of the system. As future work, this will be improved by providing proper caching mechanisms on the database, using better design patterns on the code level, etc.

- ***Better Feedback and Actions mechanism:*** Currently the system provides feedback about the authentication that was affected by the authentication algorithm to the user. However, it doesn't provide any action for users such as Undo or Cancel. As future work, improvement to the system will be made to the BMAgent system with regards to the feedbacks and actions, which provide a better overall experience to the user.

- ***Implementing the system in a Mobile system:*** mobile systems are used by airport authorities to manage their authentication information and daily operations. Integrating the BMAgent system with an existing mobile solution will allow the system to take the relevant data from the mobile system and perform the various functions implemented in the system.

8.5. Summary

The first chapter of this thesis provided a detailed introduction to this study, highlighting its aims and objectives, the background, the identified problem, and the proposed solution. Afterward, the literature around Multi-Agent systems in general, as well as the use of Multi-Agent technologies in the authentication process, were elaborated, identifying the functional as well as technical opportunities. Subsequently, the identified technologies were discussed in greater detail in the third chapter. The fourth chapter provided details about the Approach followed in the development of the BMAgent system was deliberated highlighting its hypothesis, inputs, outputs, process, features, etc. The fifth and the sixth chapters of the thesis went into intricate details about the design and the implementation of the BMAgent system respectively. There, the details of the system were discussed in terms of the types of agents, processes, and functions using diagrams such as flow charts, sequence diagrams, and class diagrams. In the subsequent chapter, it was discussed how the BMAgent system was evaluated using a dataset acquired from an existing border management system in an airport authority. Finally, in this chapter, the conclusions for the BMAgent system were discussed highlighting its pros and cons and the improvements that can be done on it in the future.

References

[1] Jammi Ashok et. al. (2010) an overview of Biometrics, (IJCSE) International Journal on Computer Science and Engineering., vol. 36, no. 4, pp. 563–577, Jul. 2010.

[2] Marcos Faundez-Zanuy Escola Universitaria Politècnica de Mataró. (2006) Biometric security technology, IEEE Aerospace and Electronic Systems Magazine Conference , Spain.

[3] Illegal Immigrants are burning issue | Ministry of Foreign Affairs (2014) https://www.mfa.gov.lk/tam/illegal-immigrants-a-burning-issue/

[4] Waldemar Wójcik, Konrad Gromaszek and Muhtar Junisbekov (2016) Face Recognition: Issues, Methods, and Alternative Applications Institute of Electronics and Information Technology, Lublin University of Technology. Lublin, Poland. vol. 21, no. 1, pp. 3–24, 2016.

[5] David White, Richard I. Kemp, Rob Jenkins, Michael Matheson, A. Mike Burton (2014), Passport Officers' Errors in Face Matching, School of Psychology, The University of New South Wales, Sydney, Australia

[6] Yanfei Zhu 1, Qiuqi Ruan (2012), Face Feature Extraction Based on Agents with Multi-camera System International Journal of Information and Computer Science, IJICS Volume 1, Issue 2, May 2012 PP. 34-38

[7] Wasim Shaikh, Hemant Shinde and Grishma Sharma (2016) Face Recognition Using Multi-Agent System, International Journal of Computer Science and Engineering.

[8] Rajeev Gupta (2011), multi-agent approach towards face recognition, M.M. Institute of Computer Technology & Business Management Maharishi Markandeshwar University. pp. 305–314.

[9] Girija Chett and Dharmendra Sharma (2006) Distributed Face Recognition: A Multi-Agent Approach, School of Information Sciences and Engineering. University of Canberra, Australia IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.8A, August 2006.

[10] Adrian Kapczyński and Tomasz Owczarek (2010), Simulation Model of Biometric Authentication Using Multiagent Approach, Journal of Telecommunication and Information Technology

[11] Deng Zhang, Shingo Mabu and Kotaro Hirasawa (2011), A Robust Intelligent Face Recognition Framework using GNP-based Multi-agent System SICE

Annual Conference 2011, September 13-18, 2011, Waseda University, Tokyo, Japan

[12] Meryem Benyoussef and Driss Aboutajdine (2011), A Distributed Approach to Color Image Segmentation [13] R. J. Rabelo and L. M. Camarinha-Matos, "Negotiation in multi-agent-based dynamic scheduling," *Robot. Comput.-Integr. Manuf.*, vol. 11, no. 4, pp. 303–309.

[14] M.Abdullah-Al-Wadud, Mohammad Shoyaib and OksamChae (2011) A Skin Detection Approach Based on Color Distance Map, The 2011 European Signal Processing Conference., pp. 105–117.

[15] Nawaf Hazim Barnouti, Sinan Sameer Mahmood Al-Dabbagh, Muhammed Hazim Jaafer Al-Bamarni (2016), Real-Time Face Detection and Recognition Using Principal Component Analysis (PCA) back Propagation Neural Network (BPNN) And Radial Basis Function (RBF). Journal of Theoretical and Applied Information Technology., vol. 23, no. 3, pp. 902–907, May 1993.

[16] M. A. Dabbah, W. L. Woo, and S. S. Dlay, "Secure Authentication for Face Recognition," In Proc. of IEEE Symposium on Computational Intelligence in Image and Signal Processing, Apr. 2007. USA, pp. 121 - 126.

[17] F.Bellifemine, G. Caire, A. Poggi, G. Rimassa, (2003) Jade A WhitePaper.., vol. 17, no. 6, pp. 667–679, Dec. 2006.

[18] Y. Demazeau, K. Hallenborg, and A. J Jensen, "Reactive agent mechanisms for scheduling manufacturing processes."

[19] X. Li, W. Li, L. Gao, C. Zhang, and X. Shao, "Multi-agent based integration of process planning and scheduling," in *2009 13th International Conference on Computer Supported Cooperative Work in Design*, Santiago, Chile, 2009, pp. 215–220.

[20] R. M. Sundaram and S.-S. Fu, "Process planning and scheduling—a method of integration for productivity improvement," *Comput. Ind. Eng.*, vol. 15, no. 1, pp. 296–301, Dec. 1988.

[21] C. E. Nugraheni and L. Abednego, "Multi-Agent Hyper-Heuristics based framework for production problem," in *2016 International Conference on Informatics and Computing (ICIC)*, 2016, pp. 309–313.

[22] K. Kravari and N. Bassiliades, "A Survey of Agent Platforms," *J. Artif. Soc. Soc. Simul.*, vol. 18, no. 1, p. 11, 2015.

[23] "Jade Site | Java Agent Development Framework.".

[24] Admin, "Java Advantages and Disadvantages," *MindsMapped*, 23-Jul-2015.

[25] K. Dyrr, "The Complete Beginner's Guide to React," p. 89.

[26] V. Industries, "The Fast Guide to OEE," *Vorne Ind.*, p. 27, 2008.

[27] A. J. De Ron and J. E. Rooda, "OEE and equipment effectiveness: an evaluation," *Int. J. Prod. Res.*, vol. 44, no. 23, pp. 4987–5003, Dec. 2006.

Bibliography

- G. Rzevski, J. Himoff, M. Hinton, and P. Skobelev, “Magenta technology multi-agent logistics i-Scheduler for road transportation,” *Proc. Fifth.*

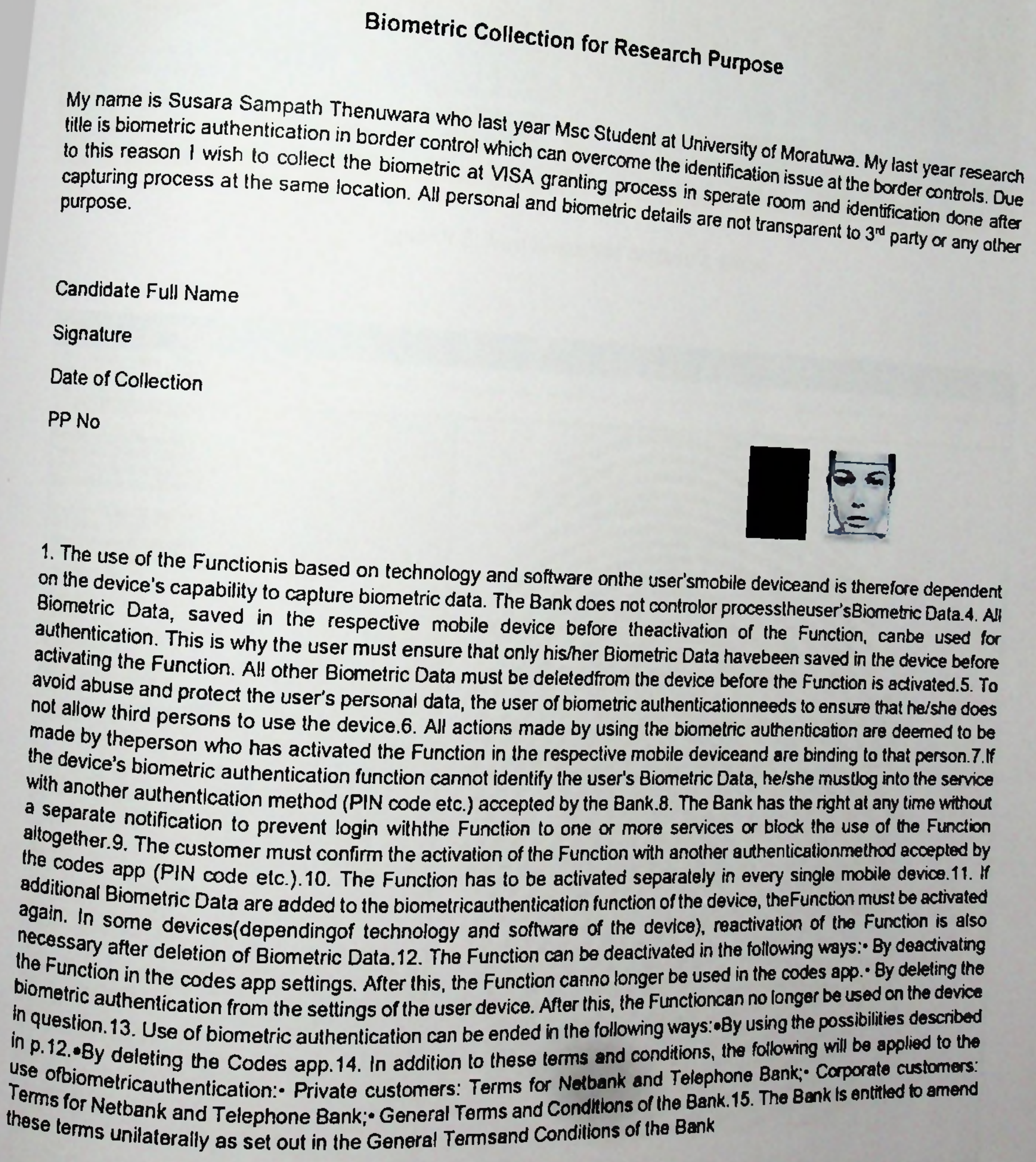
- Nawaf Hazim Barnouti and R. Rabelo, “Multi-agent-based authentication,” *Robot. Auton. Syst.*

- Girija Chett and Dharmendra Sharm: *Understanding Overall Equipment Effectiveness, Reliability, and Maintainability*, 0 ed. Productivity Press, 2017.

- Marcos Faundez-Zanuy Escola Universitaria Politècnic, “Optimal multi-agent biometrics with the multi-model approach,” in *Proceedings of the National Conference on Artificial Intelligence*, 2007, vol. 22, p. 1813.

Appendix A

Evaluating the BMAgent system using the Test Dataset



Biometric Collection for Research Purpose

My name is Susara Sampath Thenuwara who last year Msc Student at University of Moratuwa. My last year research title is biometric authentication in border control which can overcome the identification issue at the border controls. Due to this reason I wish to collect the biometric at VISA granting process in sperate room and identification done after capturing process at the same location. All personal and biometric details are not transparent to 3rd party or any other purpose.

Candidate Full Name

Signature

Date of Collection

PP No

[redacted]

1. The use of the Functionis based on technology and software onthe user'smobile deviceand is therefore dependent on the device's capability to capture biometric data. The Bank does not controlor processtheuser'sBiometric Data.4. All Biometric Data, saved in the respective mobile device before theactivation of the Function, canbe used for authentication. This is why the user must ensure that only his/her Biometric Data havebeen saved in the device before activating the Function. All other Biometric Data must be deletedfrom the device before the Function is activated.5. To avoid abuse and protect the user's personal data, the user of biometric authenticationneeds to ensure that he/she does not allow third persons to use the device.6. All actions made by using the biometric authentication are deemed to be made by theperson who has activated the Function in the respective mobile deviceand are binding to that person.7.If the device's biometric authentication function cannot identify the user's Biometric Data, he/she mustlog into the service with another authentlcation method (PIN code etc.) accepted by the Bank.8. The Bank has the right at any time without a separate notification to prevent login withthe Function to one or more services or block the use of the Function altogether.9. The customer must confirm the activation of the Function with another authenticationmethod accepted by the codes app (PIN code etc.).10. The Function has to be activated separately in every single mobile device.11. If additional Biometric Data are added to the biometricauthentication function of the device, theFunction must be activated again. In some devices(dependingof technology and software of the device), reactivation of the Function is also necessary after deletion of Biometric Data.12. The Function can be deactivated in the following ways:• By deactivating the Function in the codes app settings. After this, the Function canno longer be used in the codes app.• By deleting the biometric authentication from the settings of the user device. After this, the Functioncan no longer be used on the device in question.13. Use of biometric authentication can be ended in the following ways:•By using the possibilities described in p.12.•By deleting the Codes app.14. In addition to these terms and conditions, the following will be applied to the use ofbiometricauthentication:• Private customers: Terms for Netbank and Telephone Bank;• Corporate customers: Terms for Netbank and Telephone Bank;• General Terms and Conditions of the Bank.15. The Bank Is entitled to amend these terms unilaterally as set out in the General Termsand Conditions of the Bank

Figure 9.1: Biometric Agreement Foam

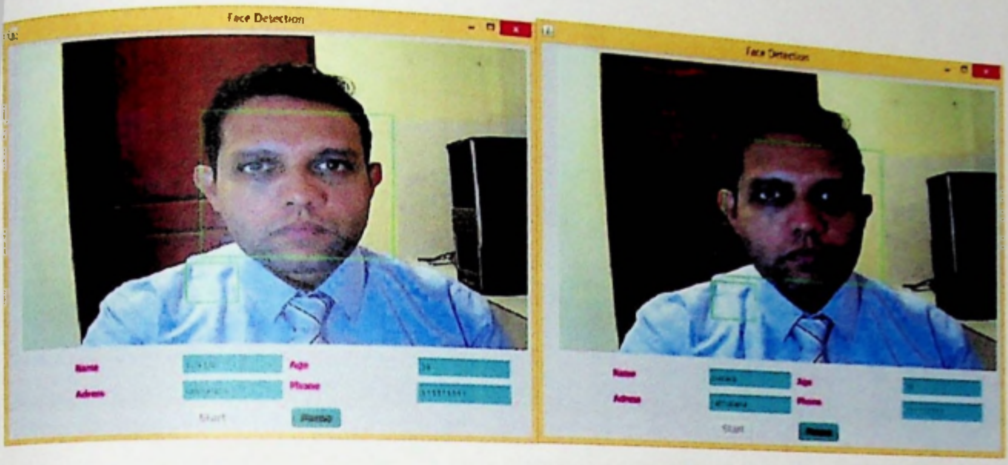
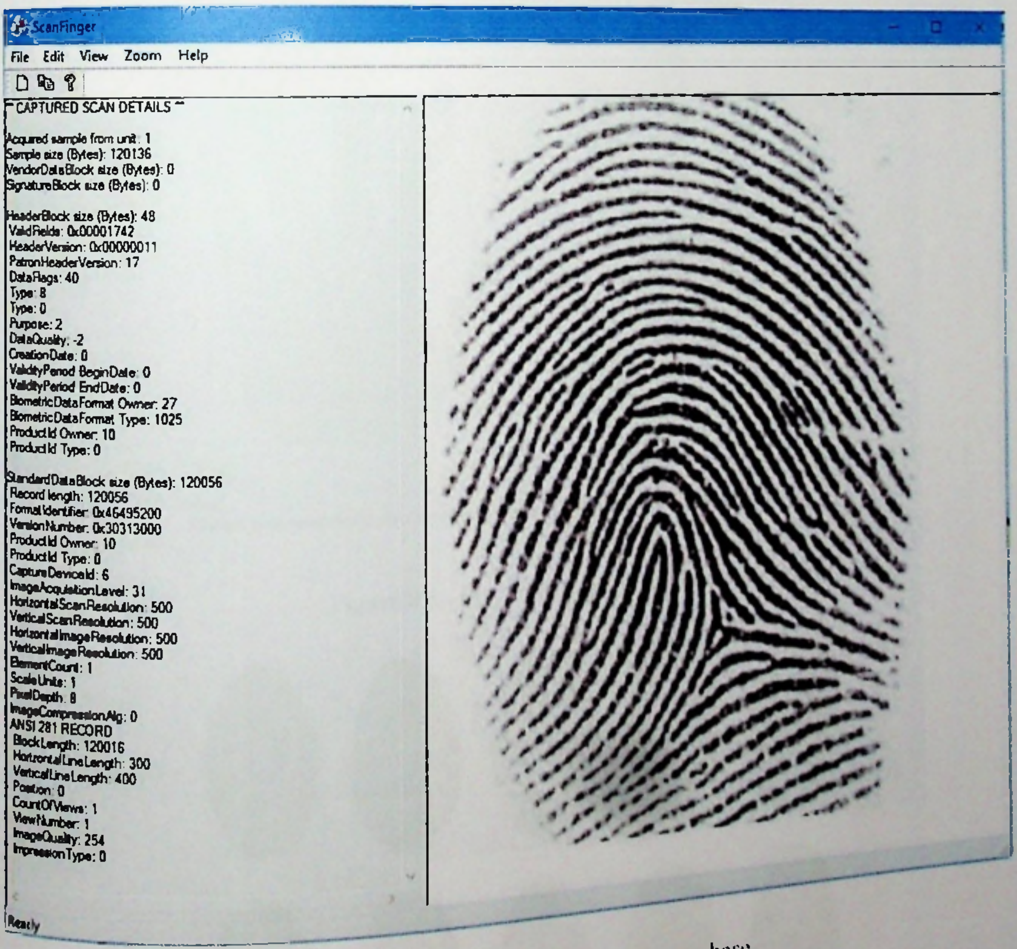


Figure 9.2: face biometric capturing phase



```
- CAPTURED SCAN DETAILS -

Acqured sample from unit: 1
Sample size (Bytes): 120136
VendorDataBlock size (Bytes): 0
SignatureBlock size (Bytes): 0

HeaderBlock size (Bytes): 48
 ValidFields: 0x00001742
 HeaderVersion: 0x00000011
 PatronHeaderVersion: 17
 DataFlags: 40
 Type: 8
 Type: 0
 Purpose: 2
 DataQuality: -2
 CreationDate: 0
 ValidityPeriod BeginDate: 0
 ValidityPeriod EndDate: 0
 BiometricDataFormat Owner: 27
 BiometricDataFormat Type: 1025
 ProductId Owner: 10
 ProductId Type: 0

StandardDataBlock size (Bytes): 120056
 Record length: 120056
 FormatIdentifier: 0x46495200
 VersionNumber: 0x30313000
 ProductId Owner: 10
 ProductId Type: 0
 CaptureDeviceId: 6
 ImageAcquisitionLevel: 31
 HorizontalScanResolution: 500
 VerticalScanResolution: 500
 HorizontalImageResolution: 500
 VerticalImageResolution: 500
 ElementCount: 1
 ScaleUnits: 1
 PixelDepth: 8
 ImageCompressionAlg: 0
 ANSI 281 RECORD
  BlockLength: 120016
  HorizontalLineLength: 300
  VerticalLineLength: 400
  Position: 0
  CountOfViews: 1
  ViewNumber: 1
  ImageQuality: 254
  ImpressionType: 0

Ready
```

Figure 9.12: Fingerprint in capturing phase

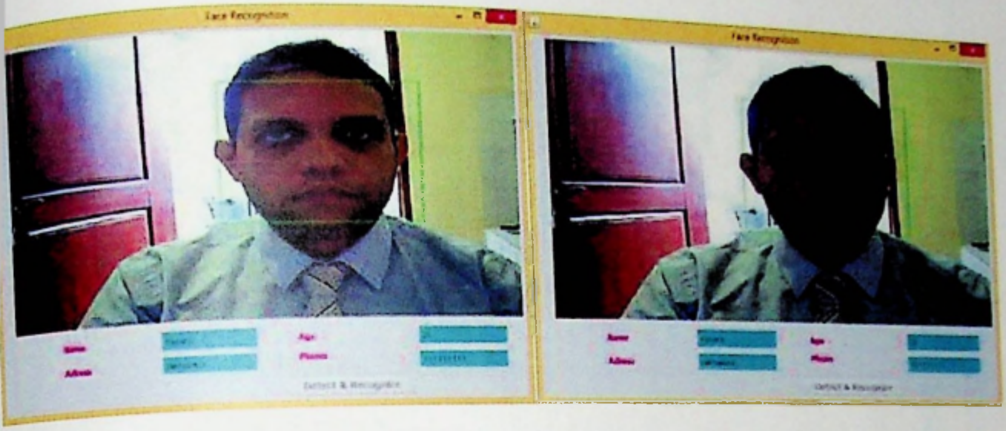
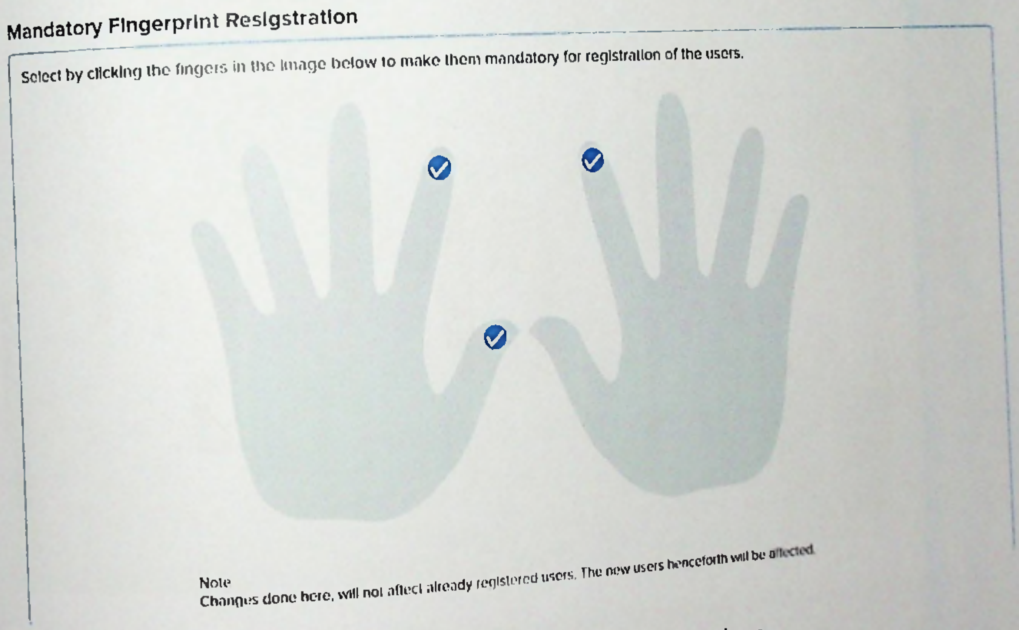


Figure 9.4: Face biometric recognition phase



Mandatory Fingerprint Resigstration

Select by clicking the fingers in the image below to make them mandatory for registration of the users.

Note
Changes done here, will not affect already registered users. The new users henceforth will be affected.

Figure 9.5: Fingerprint recognition phase

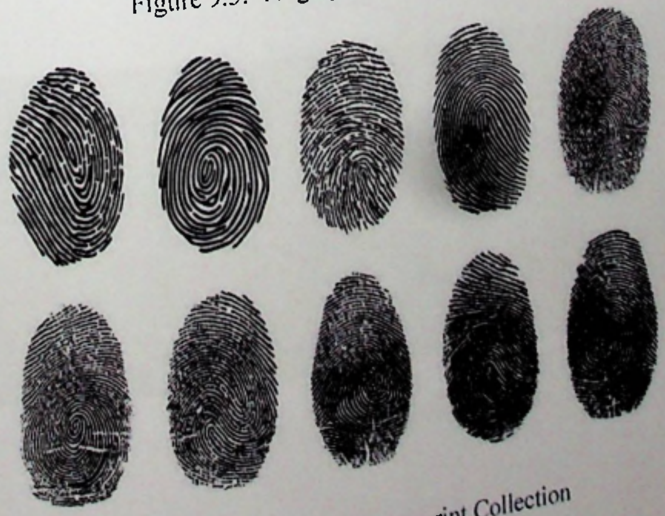


Figure 9.6: Sample fingerprint Collection

Figure 9.7: Sample images at evaluation the software

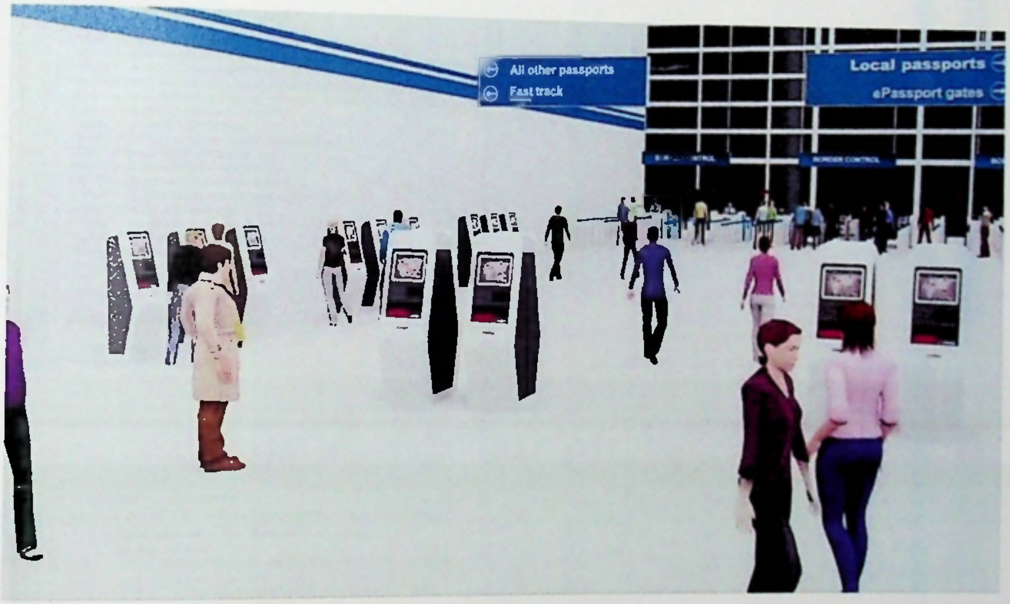


Figure 9.8: The final imaginary situation at border points after the proposed system

8

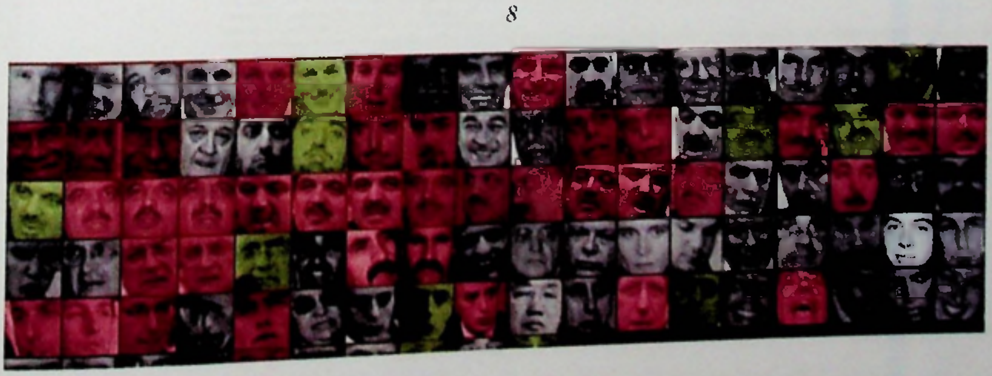
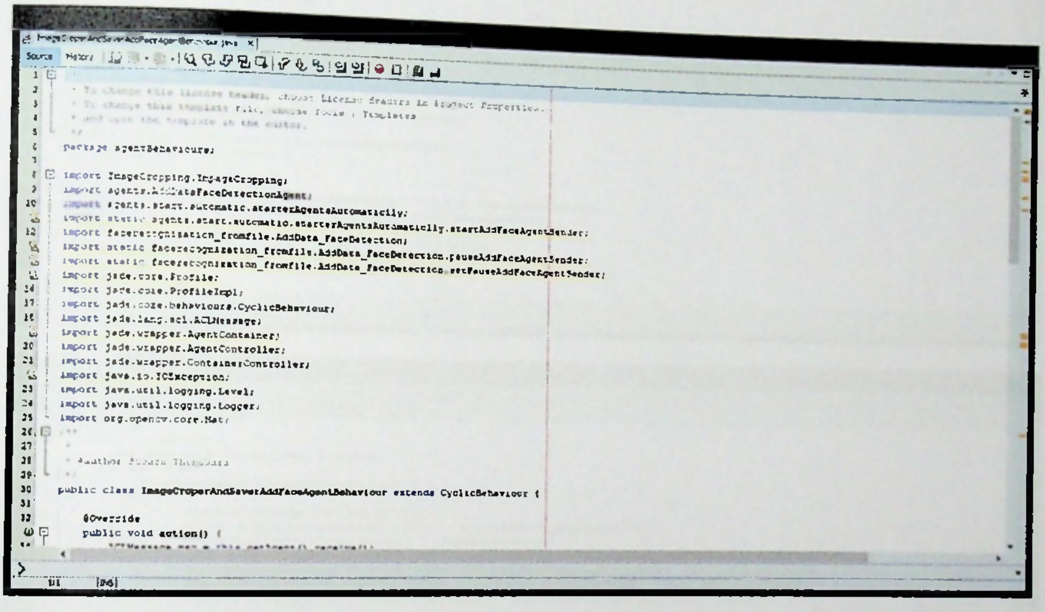


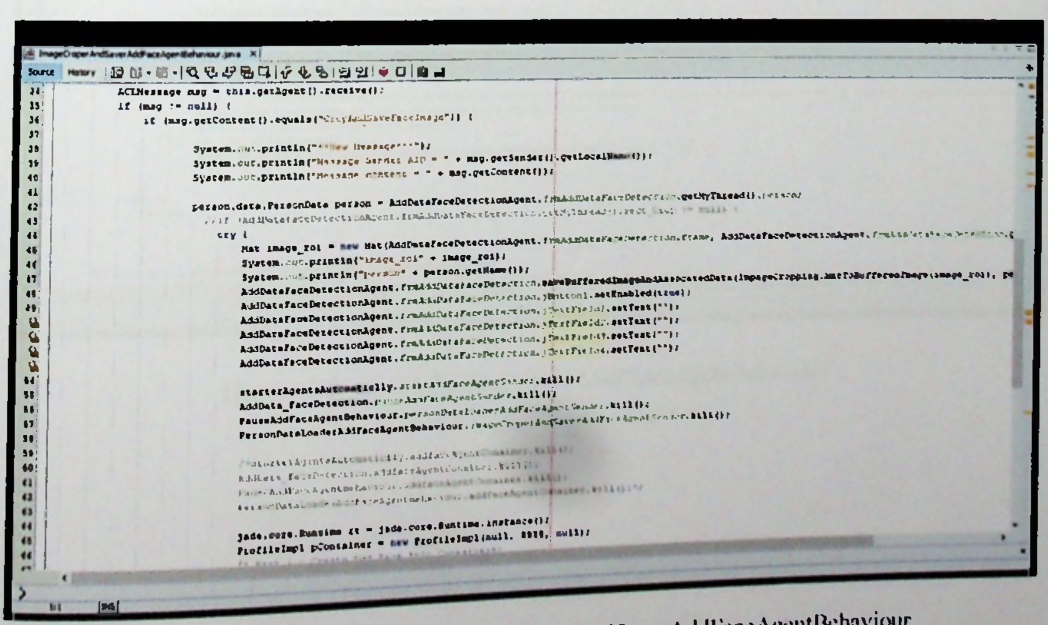
Figure 9.9: Sample eigenface Images

Appendix B

Code Sections of Different Agent Types, UIs and Message Space Screenshots

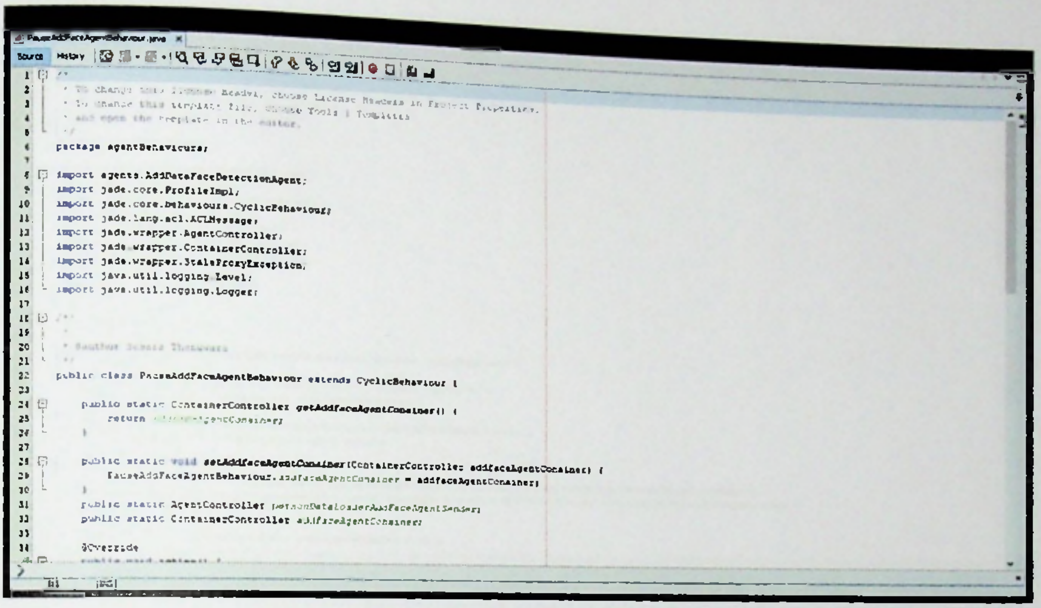


```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package agentBehaviours;

import ImageCropping.ImageCropping;
import agents.AddDataFaceDetectionAgent;
import agents.start.automatic.starterAgentsAutomaticlly;
import static agents.start.automatic.starterAgentsAutomaticlly.startAddFaceAgentSender;
import facerecognization_fromfile.AddData_FaceDetection;
import static facerecognization_fromfile.AddData_FaceDetection.pauseAddFaceAgentSender;
import static facerecognization_fromfile.AddData_FaceDetection.setPauseAddFaceAgentSender;
import jade.core.Profile;
import jade.core.ProfileImpl;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
import jade.wrapper.AgentContainer;
import jade.wrapper.AgentController;
import jade.wrapper.ContainerController;
import java.io.IOException;
import java.util.logging.Level;
import java.util.logging.Logger;
import org.opencv.core.Mat;
/**
 *
 * @author [illegible]
 */
public class ImageCroperAndSaverAddFaceAgentBehaviour extends CyclicBehaviour {

    @Override
    public void action() {
```

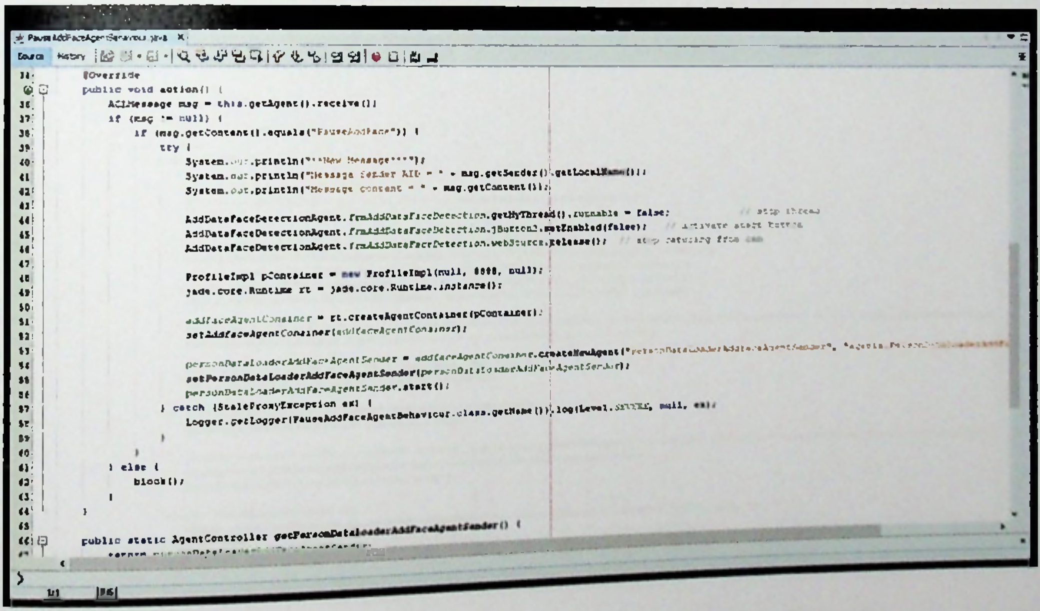


```java
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("CropAndSaveFaceImage")) {

                System.out.println("***New Message***");
                System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                System.out.println("Message content = " + msg.getContent());

                person.data.PersonData person = AddDataFaceDetectionAgent.frmAddDataFaceDetection.getMyThread().person;
              //if (AddDataFaceDetectionAgent.frmAddDataFaceDetection.getMyThread().rect_Crop != null) {
                try {
                    Mat image_roi = new Mat(AddDataFaceDetectionAgent.frmAddDataFaceDetection.frame, AddDataFaceDetectionAgent.frmAddDataFaceDetection.[illegible]
                    System.out.println("image_roi" + image_roi);
                    System.out.println("person" + person.getName());
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.saveBufferedImageAndAssociatedData(ImageCropping.matToBufferedImage(image_roi), pe
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jButton1.setEnabled(true);
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField1.setText("");
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField2.setText("");
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField3.setText("");
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField4.setText("");
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.[illegible]

                    starterAgentsAutomaticlly.startAddFaceAgentSender.kill();
                    AddData_FaceDetection.pauseAddFaceAgentSender.kill();
                    PauseAddFaceAgentBehaviour.personDataLoaderAddFaceAgentSender.kill();
                    PersonDataLoaderAddFaceAgentBehaviour.imageCroperAndSaverAddFaceAgentSender.kill();

                    /*starterAgentsAutomaticlly.addFaceAgentContainer.kill();
                    AddData_FaceDetection.addFaceAgentContainer.kill();
                    PauseAddFaceAgentBehaviour.addFaceAgentContainer.kill();
                    PersonDataLoaderAddFaceAgentBehaviour.addFaceAgentContainer.kill();*/

                    jade.core.Runtime rt = jade.core.Runtime.instance();
                    ProfileImpl pContainer = new ProfileImpl(null, 8888, null);
```

Figure 10.1: agent behaviour – ImageCroperAndSaverAddFaceAgentBehaviour

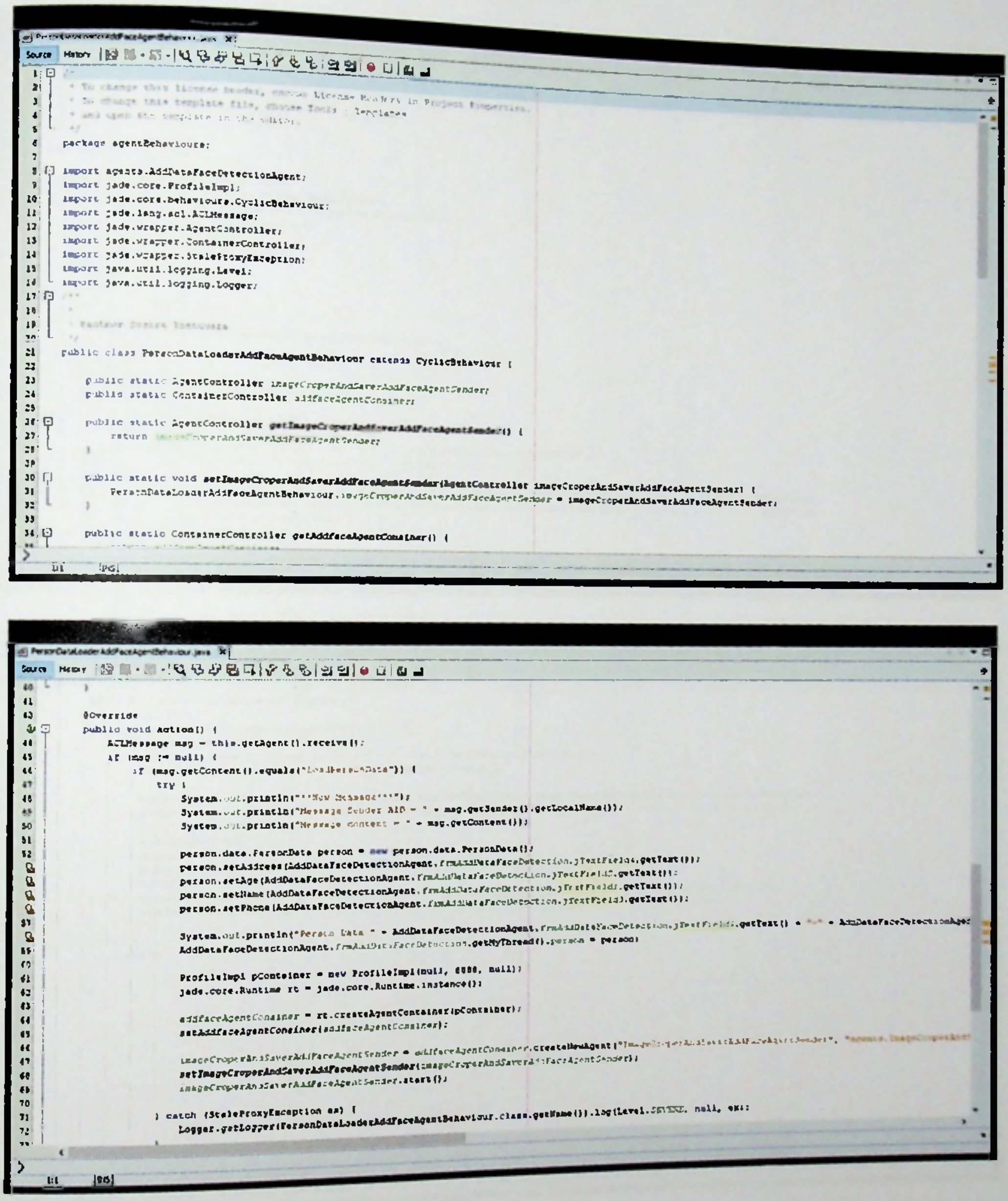


```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package agentBehaviours;

import agents.AddDataFaceDetectionAgent;
import jade.core.ProfileImpl;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
import jade.wrapper.AgentController;
import jade.wrapper.ContainerController;
import jade.wrapper.StaleProxyException;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author [illegible]
 */
public class PauseAddFaceAgentBehaviour extends CyclicBehaviour {

    public static ContainerController getAddfaceAgentConainer() {
        return addfaceAgentConainer;
    }

    public static void setAddfaceAgentConainer(ContainerController addfaceAgentConainer) {
        PauseAddFaceAgentBehaviour.addfaceAgentConainer = addfaceAgentConainer;
    }
    public static AgentController personDataLoaderAddFaceAgentSender;
    public static ContainerController addfaceAgentConainer;

    @Override
```



```
    @Override
    public void action() {
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("PauseAddFace")) {
                try {
                    System.out.println("**New Message***");
                    System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                    System.out.println("Message content = " + msg.getContent());

                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.getMyThread().runnable = false;          // stop thread
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.jButton2.setEnabled(false);    // activate start button
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.webSource.release();  // stop caturing from cam

                    ProfileImpl pContainer = new ProfileImpl(null, 8888, null);
                    jade.core.Runtime rt = jade.core.Runtime.instance();

                    addfaceAgentConainer = rt.createAgentContainer(pContainer);
                    setAddfaceAgentConainer(addfaceAgentConainer);

                    personDataLoaderAddFaceAgentSender = addfaceAgentConainer.createNewAgent("personDataLoaderAddFaceAgentSender", "agents.PersonDataLoader
                    setPersonDataLoaderAddFaceAgentSender(personDataLoaderAddFaceAgentSender);
                    personDataLoaderAddFaceAgentSender.start();
                } catch (StaleProxyException ex) {
                    Logger.getLogger(PauseAddFaceAgentBehaviour.class.getName()).log(Level.SEVERE, null, ex);
                }
            }
        } else {
            block();
        }
    }

    public static AgentController getPersonDataLoaderAddFaceAgentSender() {
```

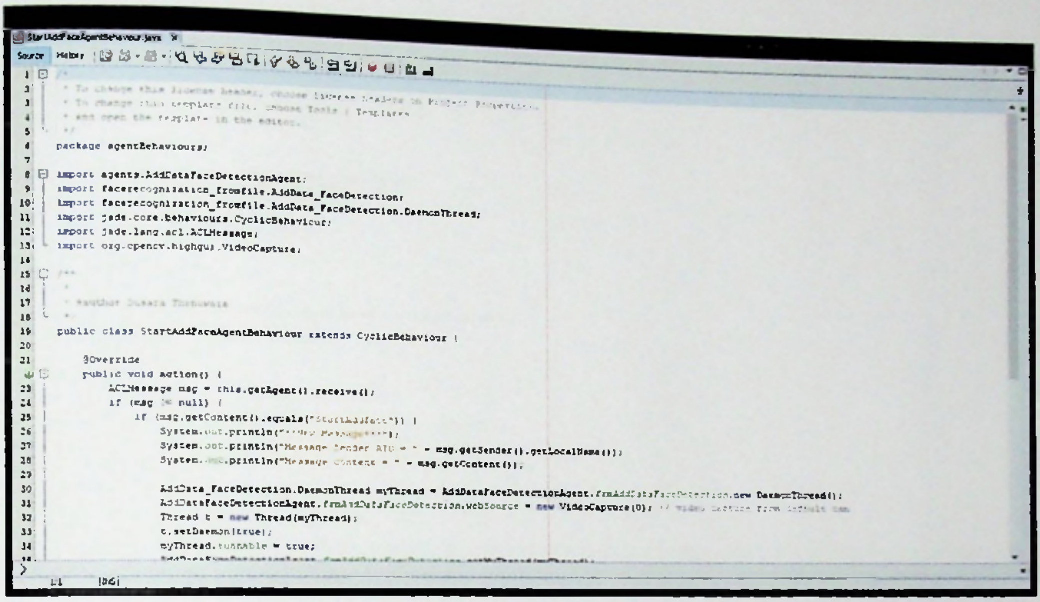
Figure 10.2: agent behaviour - PauseAddFaceAgentBehaviour



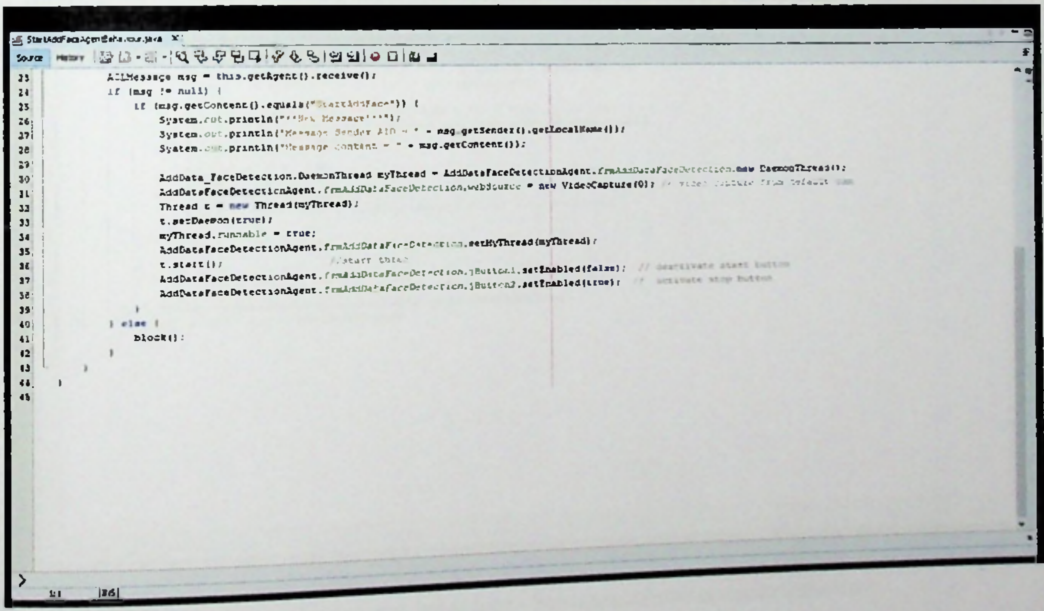
```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package agentBehaviours;

import agents.AddDataFaceDetectionAgent;
import jade.core.ProfileImpl;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
import jade.wrapper.AgentController;
import jade.wrapper.ContainerController;
import jade.wrapper.StaleProxyException;
import java.util.logging.Level;
import java.util.logging.Logger;
/**
 *
 * [illegible]
 */
public class PersonDataLoaderAddFaceAgentBehaviour extends CyclicBehaviour {

    public static AgentController imageCroperAndSaverAddFaceAgentSender;
    public static ContainerController addfaceAgentConsiner;

    public static AgentController getImageCroperAndSaverAddFaceAgentSender() {
        return imageCroperAndSaverAddFaceAgentSender;
    }

    public static void setImageCroperAndSaverAddFaceAgentSender(AgentController imageCroperAndSaverAddFaceAgentSender) {
        PersonDataLoaderAddFaceAgentBehaviour.imageCroperAndSaverAddFaceAgentSender = imageCroperAndSaverAddFaceAgentSender;
    }

    public static ContainerController getAddfaceAgentConsiner() {
```

```
    }

    @Override
    public void action() {
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("LoadPersonData")) {
                try {
                    System.out.println("**New Message***");
                    System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                    System.out.println("Message content = " + msg.getContent());

                    person.data.PersonData person = new person.data.PersonData();
                    person.setAddress(AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField4.getText());
                    person.setAge(AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField2.getText());
                    person.setName(AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField1.getText());
                    person.setPhone(AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField3.getText());

                    System.out.println("Person Data " + AddDataFaceDetectionAgent.frmAddDataFaceDetection.jTextField1.getText() + "," + AddDataFaceDetectionAge
                    AddDataFaceDetectionAgent.frmAddDataFaceDetection.getMyThread().person = person;

                    ProfileImpl pContainer = new ProfileImpl(null, 8888, null);
                    jade.core.Runtime rt = jade.core.Runtime.instance();

                    addfaceAgentConsiner = rt.createAgentContainer(pContainer);
                    setAddfaceAgentConsiner(addfaceAgentConsiner);

                    imageCroperAndSaverAddFaceAgentSender = addfaceAgentConsiner.createNewAgent("ImageCroperAndSaverAddFaceAgentSender", "agents.ImageCroperAnd
                    setImageCroperAndSaverAddFaceAgentSender(imageCroperAndSaverAddFaceAgentSender);
                    imageCroperAndSaverAddFaceAgentSender.start();

                } catch (StaleProxyException ex) {
                    Logger.getLogger(PersonDataLoaderAddFaceAgentBehaviour.class.getName()).log(Level.SEVERE, null, ex);
```

Figure 10.3: agent behaviour – PersonDataLoaderAddFaceAgentBehaviour

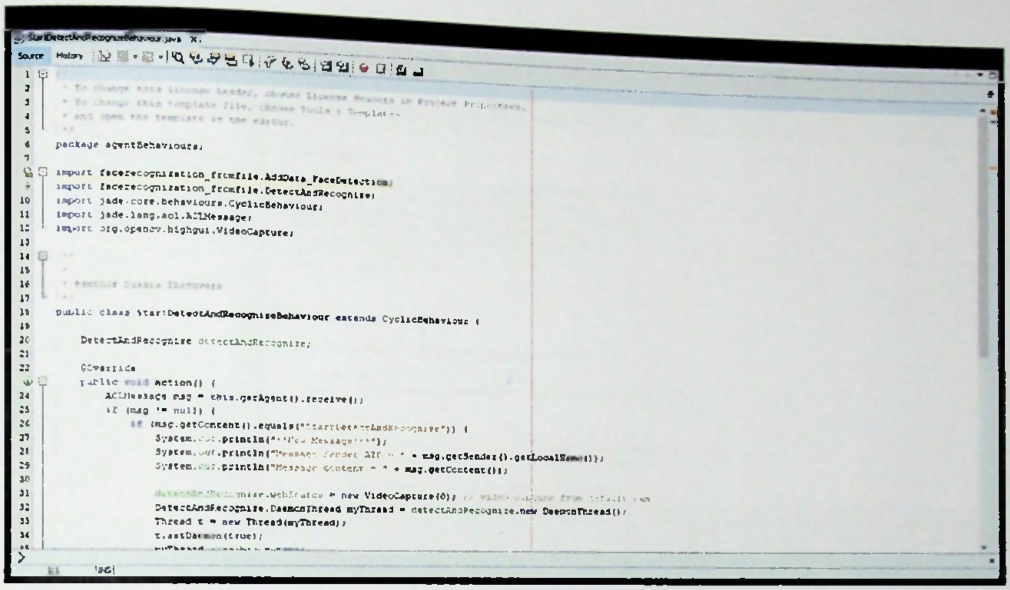


```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package agentBehaviours;

import agents.AddDataFaceDetectionAgent;
import facerecognization_fromfile.AddData_FaceDetection;
import facerecognization_fromfile.AddData_FaceDetection.DaemonThread;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
import org.opencv.highgui.VideoCapture;

/**
 *
 * @author [illegible]
 */
public class StartAddFaceAgentBehaviour extends CyclicBehaviour {

    @Override
    public void action() {
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("StartAddFace")) {
                System.out.println("***New Message***");
                System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                System.out.println("Message Content = " + msg.getContent());

                AddData_FaceDetection.DaemonThread myThread = AddDataFaceDetectionAgent.frmAddDataFaceDetection.new DaemonThread();
                AddDataFaceDetectionAgent.frmAddDataFaceDetection.webSource = new VideoCapture(0); // video capture from default cam
                Thread t = new Thread(myThread);
                t.setDaemon(true);
                myThread.runnable = true;
```

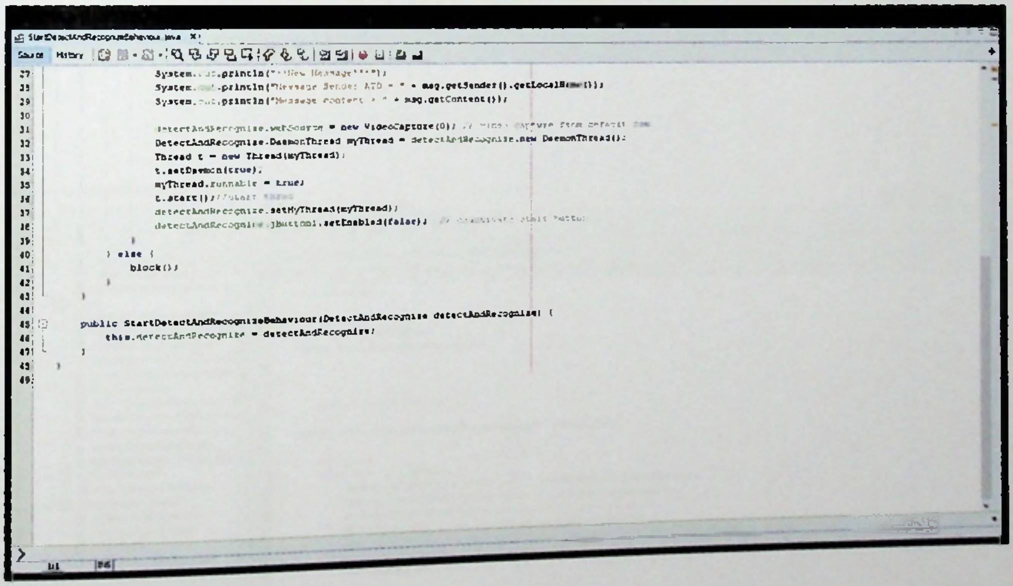


```java
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("StartAddFace")) {
                System.out.println("***New Message***");
                System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                System.out.println("Message Content = " + msg.getContent());

                AddData_FaceDetection.DaemonThread myThread = AddDataFaceDetectionAgent.frmAddDataFaceDetection.new DaemonThread();
                AddDataFaceDetectionAgent.frmAddDataFaceDetection.webSource = new VideoCapture(0); // video capture from default cam
                Thread t = new Thread(myThread);
                t.setDaemon(true);
                myThread.runnable = true;
                AddDataFaceDetectionAgent.frmAddDataFaceDetection.setMyThread(myThread);
                t.start();                  //start thread
                AddDataFaceDetectionAgent.frmAddDataFaceDetection.jButton1.setEnabled(false);   // deactivate start button
                AddDataFaceDetectionAgent.frmAddDataFaceDetection.jButton2.setEnabled(true);   // activate stop button

            }
        } else {
            block();
        }
    }
}
```

Figure 10.4: agent behaviour - StartAddFaceAgentBehaviour

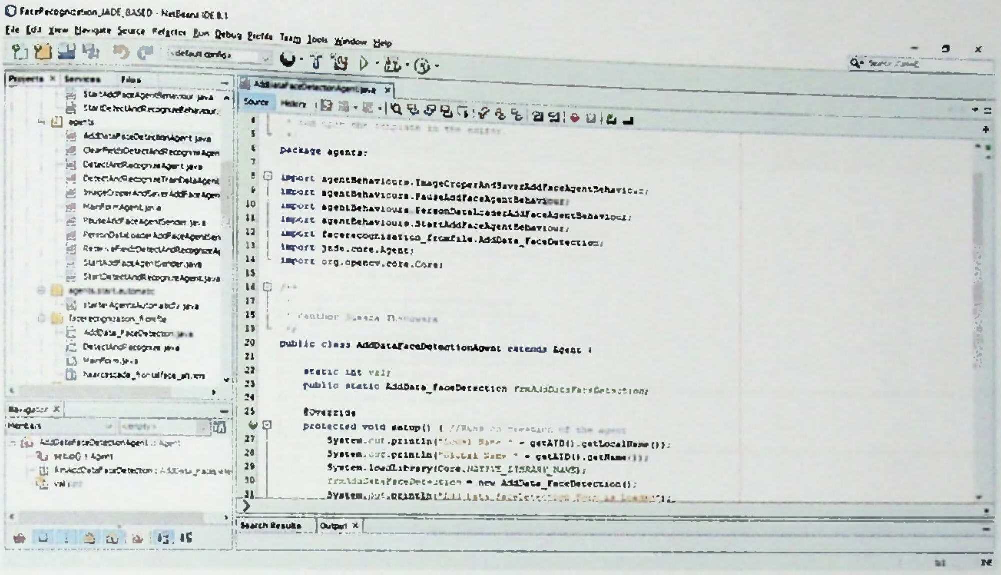


```
StartDetectAndRecognizeBehaviour.java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package agentBehaviours;

import facerecognization_fromfile.AddData_FaceDetection;
import facerecognization_fromfile.DetectAndRecognize;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;
import org.opencv.highgui.VideoCapture;

/**
 *
 * @author [illegible]
 */
public class StartDetectAndRecognizeBehaviour extends CyclicBehaviour {

    DetectAndRecognize detectAndRecognize;

    @Override
    public void action() {
        ACLMessage msg = this.getAgent().receive();
        if (msg != null) {
            if (msg.getContent().equals("StartDetectAndRecognize")) {
                System.out.println("***New Message***");
                System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                System.out.println("Message content = " + msg.getContent());

                detectAndRecognize.webSource = new VideoCapture(0); // video capture from default cam
                DetectAndRecognize.DaemonThread myThread = detectAndRecognize.new DaemonThread();
                Thread t = new Thread(myThread);
                t.setDaemon(true);
```



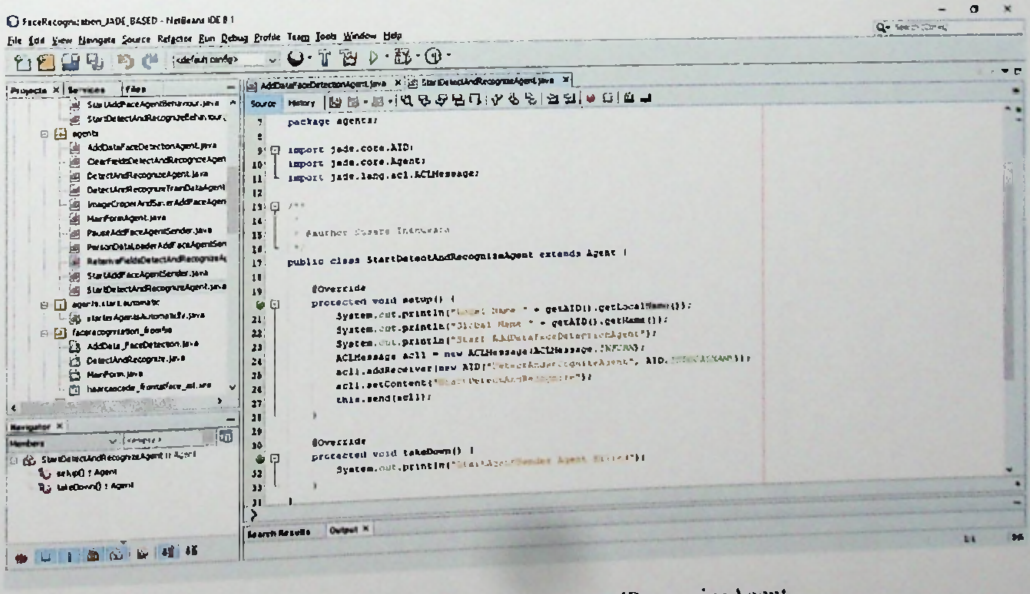
```
StartDetectAndRecognizeBehaviour.java
                System.out.println("***New Message***");
                System.out.println("Message Sender AID = " + msg.getSender().getLocalName());
                System.out.println("Message content = " + msg.getContent());

                detectAndRecognize.webSource = new VideoCapture(0); // video capture from default cam
                DetectAndRecognize.DaemonThread myThread = detectAndRecognize.new DaemonThread();
                Thread t = new Thread(myThread);
                t.setDaemon(true);
                myThread.runnable = true;
                t.start();//start thread
                detectAndRecognize.setMyThread(myThread);
                detectAndRecognize.jButton1.setEnabled(false); // deactivate start button
            }
        } else {
            block();
        }
    }

    public StartDetectAndRecognizeBehaviour(DetectAndRecognize detectAndRecognize) {
        this.detectAndRecognize = detectAndRecognize;
    }
}
```

Figure 10.4: agent behaviour - StartDetectAndRecognizeBehaviour


FaceRecognization_JADE_BASED - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects
Services
Files
StartAddFaceAgentBehaviour.java
StartDetectAndRecognizeBehaviour.java
agents
AddDataFaceDetectionAgent.java
ClearFieldsDetectAndRecognizeAgent
DetectAndRecognizeAgent.java
DetectAndRecognizeTrainDataAgent
ImageCroperAndSaverAddFaceAgent
MainFormAgent.java
PauseAddFaceAgentSender.java
PersonDataLoaderAddFaceAgentSen
RetrieveFieldsDetectAndRecognizeAg
StartAddFaceAgentSender.java
StartDetectAndRecognizeAgent.java
agents.start.automatic
starterAgentsAutomatically.java
facerecognization_fromfile
AddData_FaceDetection.java
DetectAndRecognize.java
MainForm.java
haarcascade_frontalface_alt.xml
Navigator
Members
AddDataFaceDetectionAgent :: Agent
setup() : Agent
frmAddDataFaceDetection : AddData_FaceDetection
val : int
AddDataFaceDetectionAgent.java
Source
History
Search Results
Output

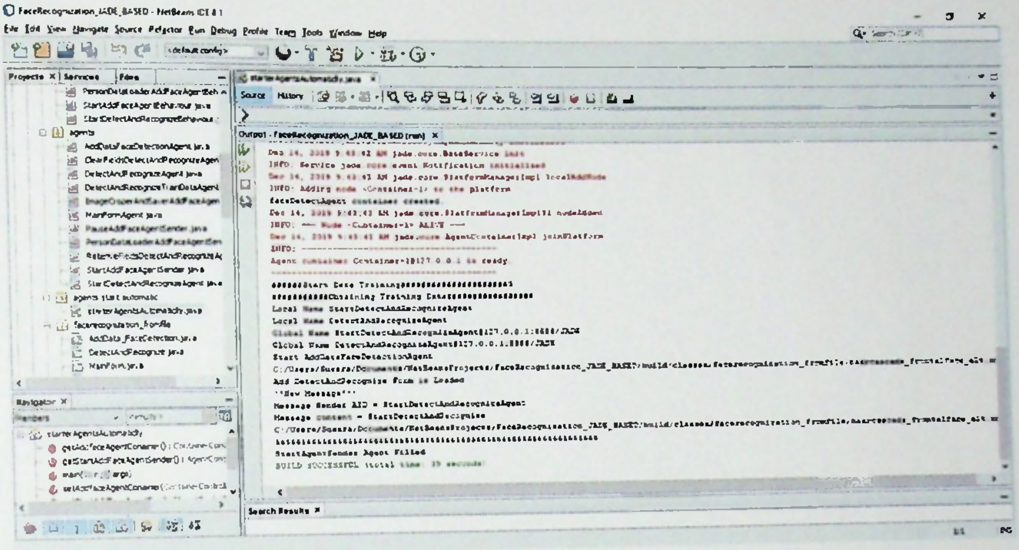
```
 * and open the template in the editor.
 */
package agents;

import agentBehaviours.ImageCroperAndSaverAddFaceAgentBehaviour;
import agentBehaviours.PauseAddFaceAgentBehaviour;
import agentBehaviours.PersonDataLoaderAddFaceAgentBehaviour;
import agentBehaviours.StartAddFaceAgentBehaviour;
import facerecognization_fromfile.AddData_FaceDetection;
import jade.core.Agent;
import org.opencv.core.Core;

/**
 *
 * @author [illegible]
 */
public class AddDataFaceDetectionAgent extends Agent {

    static int val;
    public static AddData_FaceDetection frmAddDataFaceDetection;

    @Override
    protected void setup() { //Runs on creation of the agent
        System.out.println("Local Name " + getAID().getLocalName());
        System.out.println("Global Name " + getAID().getName());
        System.loadLibrary(Core.NATIVE_LIBRARY_NAME);
        frmAddDataFaceDetection = new AddData_FaceDetection();
        System.out.println("Add data facedetection form is loaded");
```

Figure 10.13: agent - AddDataFaceDetectionAgent


FaceRecognization_JADE_BASED - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects
Services
Files
StartAddFaceAgentBehaviour.java
StartDetectAndRecognizeBehaviour.java
agents
AddDataFaceDetectionAgent.java
ClearFieldsDetectAndRecognizeAgen
DetectAndRecognizeAgent.java
DetectAndRecognizeTrainDataAgent
ImageCroperAndSaverAddFaceAgen
MainFormAgent.java
PauseAddFaceAgentSender.java
PersonDataLoaderAddFaceAgentSen
RetrieveFieldsDetectAndRecognizeAg
StartAddFaceAgentSender.java
StartDetectAndRecognizeAgent.java
agents.start.automatic
starterAgentsAutomatically.java
facerecognization_fromfile
AddData_FaceDetection.java
DetectAndRecognize.java
MainForm.java
haarcascade_frontalface_alt.xml
Navigator
Members
StartDetectAndRecognizeAgent :: Agent
setup() : Agent
takeDown() : Agent
AddDataFaceDetectionAgent.java
StartDetectAndRecognizeAgent.java
Source
History
Search Results
Output

```
package agents;

import jade.core.AID;
import jade.core.Agent;
import jade.lang.acl.ACLMessage;

/**
 *
 * @author [illegible]
 */
public class StartDetectAndRecognizeAgent extends Agent {

    @Override
    protected void setup() {
        System.out.println("Local Name " + getAID().getLocalName());
        System.out.println("Global Name " + getAID().getName());
        System.out.println("Start AddDataFaceDetectionAgent");
        ACLMessage acl1 = new ACLMessage(ACLMessage.INFORM);
        acl1.addReceiver(new AID("DetectAndRecognizeAgent", AID.ISLOCALNAME));
        acl1.setContent("StartDetectAndRecognize");
        this.send(acl1);
    }

    @Override
    protected void takeDown() {
        System.out.println("StartAddFaceSender Agent Killed");
    }
}
```

Figure 10.14: agent – StartDetectAndRecognizeAgent



FaceRecognization_JADE_BASED - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects Services Files
PersonDataLoaderAddFaceAgentBeh
StartAddFaceAgentBehaviour.java
StartDetectAndRecognizeBehaviour
agents
AddDataFaceDetectionAgent.java
ClearFieldsDetectAndRecognizeAgent
DetectAndRecognizeAgent.java
DetectAndRecognizeTrainDataAgent
ImageCropperAndSaverAddFaceAgen
MainFormAgent.java
PauseAddFaceAgentSender.java
PersonDataLoaderAddFaceAgentSen
RetrieveFieldsDetectAndRecognizeA
StartAddFaceAgentSender.java
StartDetectAndRecognizeAgent.java
agents.start.automatic
StarterAgentsAutomatically.java
facerecognization_fromfile
AddData_FaceDetection.java
DetectAndRecognize.java
MainForm.java
Navigator
Members
StarterAgentsAutomatically
getAddFaceAgentContainer() : ContainerCont
getStartAddFaceAgentSender() : AgentCont
main(String[] args)
setAddFaceAgentContainer(ContainerControll
StarterAgentsAutomatically.java
Source History
Output - FaceRecognization_JADE_BASED (run)
Search Results

```
Dec 14, 2019 9:43:42 AM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
Dec 14, 2019 9:43:43 AM jade.core.PlatformManagerImpl localAddNode
INFO: Adding node <Container-1> to the platform
faceDetectAgent container created.
Dec 14, 2019 9:43:43 AM jade.core.PlatformManagerImpl$1 nodeAdded
INFO: --- Node <Container-1> ALIVE ---
Dec 14, 2019 9:43:43 AM jade.core.AgentContainerImpl joinPlatform
INFO: ----------------------------------------
Agent container Container-1@127.0.0.1 is ready.
--------------------------------------------
######Start Face Training################################
##########Obtaining Training Data####################
Local Name StartDetectAndRecognizeAgent
Local Name DetectAndRecognizeAgent
Global Name StartDetectAndRecognizeAgent@127.0.0.1:1099/JADE
Global Name DetectAndRecognizeAgent@127.0.0.1:1099/JADE
Start AddDataFaceDetectionAgent
C:/Users/Susara/Documents/NetBeansProjects/FaceRecognization_JADE_BASED/build/classes/facerecognization_fromfile/haarcascade_frontalface_alt.xm
Add DetectAndRecognize Form is Loaded
**New Message***
Message Sender AID = StartDetectAndRecognizeAgent
Message content = StartDetectAndRecognize
C:/Users/Susara/Documents/NetBeansProjects/FaceRecognization_JADE_BASED/build/classes/facerecognization_fromfile/haarcascade_frontalface_alt.xm
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
StartAgentSender Agent Killed
BUILD SUCCESSFUL (total time: 35 seconds)
```

Figure 10.7: Message space