

Bilkent University

Department of Computer Engineering

Senior Project

Who do you resemble?

Merve Soner

Merve Yurdakul

R. Baturalp Torun

Sedef Özlen

Supervisor: Pinar Duygulu Şahin

Jury Members: Selim Aksoy, H. Altay Güvenir

Analysis Report

October 30, 2007

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Projects course CS491.

TABLE of CONTENTS

1. INTRODUCTION	3
2. CURRENT SYSTEMS	4
2.1. MyHeritage	4
2.2. Face Double.....	5
3. PROPOSED SYSTEM	6
3.1. Overview	6
3.2. Functional Requirements	7
3.2.1. Database Construction	7
3.2.1.1. Web Crawling	8
3.2.1.2. Rectification	8
3.2.1.3. Feature Extraction	9
3.2.2. Querying and Retrieval	9
3.2.2.1. Querying and Retrieval for Finding Similar Faces	10
3.2.2.2. Querying and Retrieval for Face Comparison	10
3.3. Non-Functional Requirements	11
3.3.1. Database	11
3.3.2. Scalability	11
3.3.3. Response Time	11
3.3.4. Resource Usage	11
3.3.5. Reliability	11
3.3.6. Reusability	11
3.3.7. Platform	12
3.3.8. Technology	12
3.4. Pseudo Requirements	12
3.5. System Models	12
3.5.1. Use Cases	12
3.5.2. Scenarios	12
3.5.3. Object Models	13
3.5.4. User Interface	14
4. REFERENCES	16

1. INTRODUCTION

A facial recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source. Over the last ten years or so, detecting and recognizing human faces automatically is becoming a very important task in many applications, such as security access control systems or content-based indexing video retrieval systems. It is one of the biggest research areas for computer vision and for most successful applications of image analysis and understanding. [1]

One of the ways to do face recognition is by comparing selected facial features from the image and a facial database. It is typically used in security systems and can be compared to other biometrics such as fingerprint or eye iris recognition systems. Popular recognition algorithms include eigenface, fisherface, the Hidden Markov model, and the neuronal motivated dynamic link matching. [7] A newly emerging trend, claimed to achieve previously unseen accuracies, is three-dimensional face recognition. Another emerging trend uses the visual details of the skin, as captured in standard digital or scanned images.

The advanced technology on this topic has been trying to meet the requirements of finding desired information easily. However, when the issue is majored on searching based on images, especially faces, current technology cannot fulfill the requirements. [6] Our goal is to select the fastest, most qualified and robust algorithms and develop applications that carry human interaction with computer and visual data processing onto next level. Our primary focus is specifically faces detection and recognition field.

Our system will be functioned by query including specific face. The system performs face detection on the given input. Face recognition process will be performed together with face detection for matching the most similar faces specified by user.

For our project, we will design high level system for using multi model data for recognizing face having detected the faces with the specific face given in a query. To accomplish the aim, we will use and develop the existing algorithms of face recognition. We will use the existing systems working on face detection and recognition but the main aim is to find the best system to be developed to fulfill the requirements.

2. CURRENT SYSTEMS

Similar projects about face recognition are examined and analyzed with their pros and cons.

2.1. MyHeritage

My heritage is constructed as a web application and could be accessible by most common web browsers at www.myheritage.com. It is developed by a group of Jewish people and has a headquarters in Israel, near Tel-Aviv. My Heritage project is mainly concentrated on genealogy search engine, which let people to find their ancestors. It provides tools for genealogy and family history research. Furthermore, project has a community with almost 20 million members all around the world to meet, communicate and share families online.

On the other hand, My Heritage has some tools about face recognition, which are quite similar to our project. Users need to be signed in to be able to use its tools that are *Celebrity Collage*, *Celebrity Morph* and *Look-a Like Meter*. *Celebrity Collage* finds a set of similar celebrities based on photo which user submits. *Celebrity Morph*¹ is a new tool, which morphs a user's photo into a celebrity's face. Finally, *Look-a Like Meter* is also another new tool, which determines the resemblance ratio of children to their parents.

My Heritage claims that they have used the advanced face recognition technologies while implementing these tools. Since it is not an open source project, it is quite difficult to be sure about implementation details. Therefore, tools are examined in technical aspects of features. Face recognition tool accepts one image as query input and finds similar faces with resemblance ratio in a huge celebrity database.

First of all, the entire project has support for many languages including Turkish. Face recognition tool has support for detection of multi faces in one photo, which is quite useful. It can detect almost all faces even if they are too small and it rarely detects faces from movies scenes. Everything is done automatically including recognition of gender however it does not let users to select faces manually. Therefore, users do not have chance to find resemblance ratio when submitted photo has been rejected. It successfully finds the same person when a user submits a photo of celebrity. However it might not find the same

¹ Celebrity Morph's sample gallery can be seen at <http://www.myheritage.com/FP/Company/morph-gallery.php>

celebrity if photo has been taken from different angles rather than front view. This situation especially occurs in movie scenes.

Furthermore, photo submission can be done from another URL or by user upload. In addition to these options, most common photo service providers are supported such as Photobucket² and Bebo³. If a user has an account from one these providers, user can directly select photos from his/her albums that he/she has created already. Unfortunately, My Heritage does not have any mobile support for any kind of mobile devices.

In conclusion, My Heritage is best among competitors with its high ratio of finding similar faces.

2.2. Face Double

Face Double is also constructed as a web application such as My Heritage and could be accessible by most common web browsers at www.facedouble.com. TeamSOA developed Face Double in 2007, which is a quite new project. It directly concentrates on face recognition and focuses on entertainment purposes. Main purpose of Face Double project is to let people finding their celebrity look-alike. Since Face Double is not an open source project, it is not easy to be sure about implementation details such as which face detection algorithms and technologies are used. Therefore, project has been examined based on built-in features and usability issues.

Face Double accepts one image file as query input and eventually returns set of similar faces without their resemblance ratio. Face Double has an easy-to-use process and it is not required a membership for basic features. However user interface can be improved for a better usability. By signing up Face Double, members get access for extended features and unlimited usage of project both for mobile and web. Users can upload an image from their personal computers or give URL of an existing image file on any remote server. However it does not support any other type of image submission. Although Face Double has an easy process of face recognition; unfortunately, it does not support multi face detection on the same photo. Also, users are obligatory to select gender of the given person otherwise it does not return a logical data set. It cannot detect faces from different angles and movie scenes but it allows users to select a face manually if it cannot find any. However it might

² Homepage of Photobucket: <http://photobucket.com>

³ Homepage of Bebo: <http://www.bebo.com>

not recognize the face, even though user selects the face manually in image. It usually cannot find the same person when user submits a photo of celebrity. On the other hand, it has mobile support for image submission. Therefore, sending emails through any mobile device such as mobile phones is enough to use service properly.

Consequently, Face Double is a good alternative for My Heritage's celebrity look-alike tool even though it has many missing points on technology and technical features. Since it is a project mainly focused on entertainment purposes, lack of technology is not a big deal.

3. PROPOSED SYSTEM

3.1. Overview

The goal of our project is to find the similarities between faces by using their certain features. Our system is composed of two main systems. The first system deals with creating the database to store the faces and their features. For this phase, the face photos are first converted into Gray Scale and then they are normalized for accurate comparison. From these normalized photos, the features are extracted and stored in the database.

The second system mainly involves querying and retrieval phase. The face in the given photo should be identified, normalized and be represented in Gray Scale in order to extract its features. After the features of the photo are extracted, they are compared with all of the feature vectors that are stored in the database. The faces which have the most similar feature vectors with the given face are displayed.

We tried to make the system as flexible as possible. There is room to make further improvements and changes on this system. The overview of the system is also shown in Figure.1.

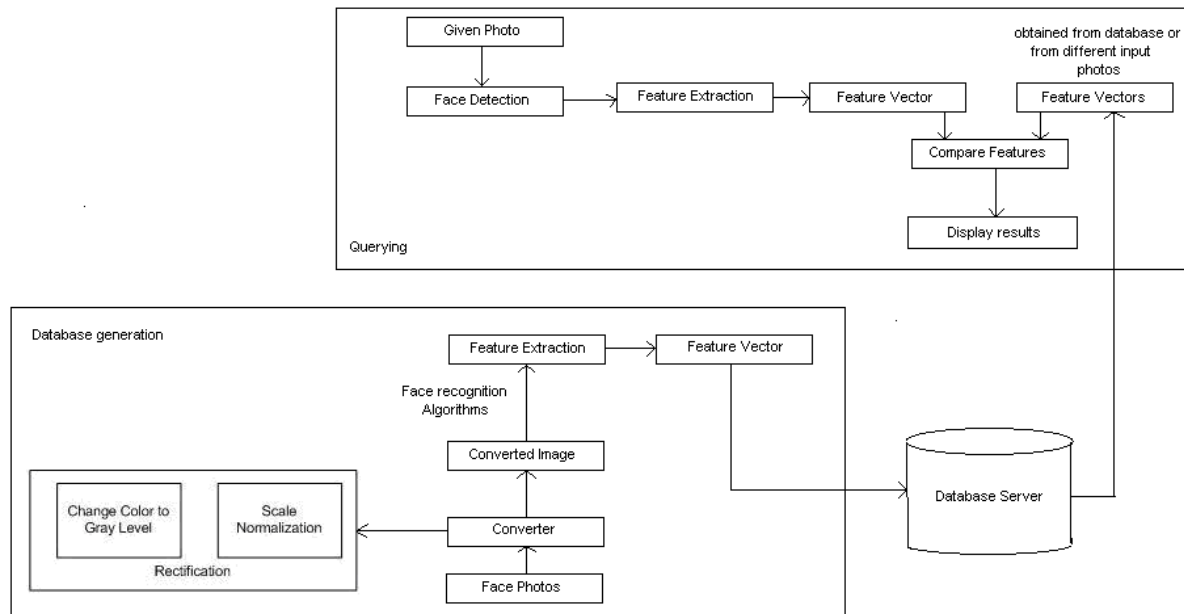


Figure 1: System overview

3.2. Functional Requirements

3.2.1. Database Construction

Before starting the project, a database needs to be constructed well to be able to get logical subsets for comparing faces. There are a few steps during database construction for collecting meaningful data that will be used for face recognition. As it can be seen from Figure.2, original face images are gathered from various sources over Internet. After that a converter performs different processes in order to standardize face images. All of these processes are called rectification. Converted images are transformed into features vector that are recorded in the database.

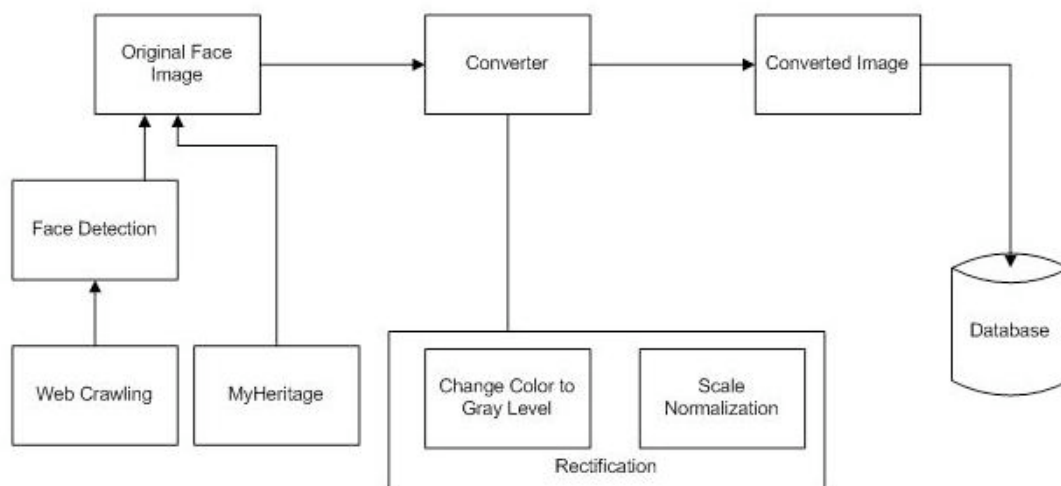


Figure 2: Database construction overview

Raw and processed image files are stored physically in hard disk and these files are referenced in database. Corresponding references are stored in database as features vector which are usually set of floats and integers. Moreover, images are also tagged and indexed with person's name if textual search is required for any purpose.

The appearance of a face is affected by a large number of factors including identity, face pose, illumination, facial expression, age, occlusion, and facial hair. The development of algorithms robust to these variations requires databases of sufficient size that include carefully controlled variations of these factors. (Gross 301) Therefore constructing database and collecting data are quite important for the reliability of the project.

3.2.1.1. Web Crawling

Real world data sets should be used in order to meet reliability requirements of project. Therefore, a web crawler has been designed which is focused on My Heritage's raw images of celebrities. A few photos of each celebrity are needed for a better and robust result set. The web crawler is designed for getting images from My Heritage's website at this stage; however, it will be a general-purpose crawler for future usage. It is planned to crawl also news with images and will get title, summary and image of the news from various news sources over Internet. Therefore, textual descriptions could be used while relating different faces among pictures. Web crawler is going to work during project development and database will be extended; therefore it gives a chance for maintaining an up-to-date database. During web crawling process, there will not be any face detection or elimination for raw images. So it will work the same way as general-purpose image crawlers.

3.2.1.2. Rectification

Rectification must be performed in order to increase efficiency of data sets and to eliminate unnecessary images. Rectification process needs to be performed in different manners. During rectification, entire database is traversed. First of all, images, which have inappropriate sizes, are eliminated. Afterwards, faces are detected and normalized if required. The aim is detecting several faces from a scene and does scaling process in order to make all images standardized. Therefore, every notable face will be cropped and standardized from raw images.

Another important point at this stage is converting RGB images into Grayscale, since colors are not important for face recognition at this project. Usually, skin color is not used as a feature. Therefore, color ranging between 0 and 255 is enough to extract many features from data sets. Eventually database consists of only faces, which are formerly normalized and converted into Grayscale. Since crawling more images might extend database, rectification process should be done continuously.

3.2.1.3. Feature Extraction

For image processing, feature extraction needs to be performed since face recognition algorithms need less data but more information rather than a raw image. Therefore images are processed in order to get more information from data set. As it can be seen from Figure.3, input data is transformed into set of features (also known as features vector) and this process is called feature extraction.

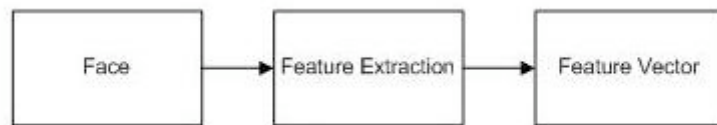


Figure 3: Feature Extraction Process

During feature extraction, PCA and LDA could be used in order to reduce multiple layered data sets to lower layered data sets. PCA stands for Principal Components Analysis that is technique used to reduce multi-dimensional data sets to lower dimensions for analysis. (Wikipedia: Principal Components Analysis) LDA stands for Linear Discriminant Analysis and it is closely related with PCA. LDA is primarily used in face recognition for reducing the number of features to a more reasonable number before classification. (Wikipedia: Linear Discriminant Analysis) One of these methods will be used for feature vectors in feature extraction.

3.2.2. Querying and Retrieval

The second functional requirement of the system is the querying and the retrieval of the data. In order to query and retrieve the data, the system should process the given data. The process step consists of face detection and face recognition. Face recognition is a complex step and the details are given in the Feature Extraction part of the previous section. For face detection, we are planning to use the algorithm in OpenCV

which has been proposed by Viola and Jones. The algorithm is distinguished by three key contributions which are the introduction of a new image representation, a learning algorithm and a method for combining increasingly more complex classifiers in a cascade (Viola& Jones, 2001).

The general process of the querying and retrieval is shown in Figure.4

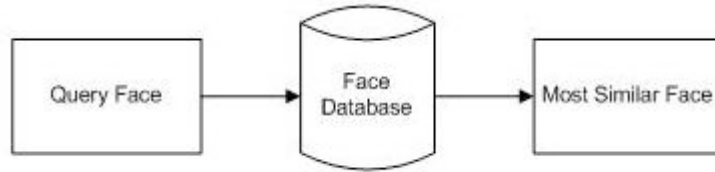


Figure.4 – Process of querying and retrieval

We divide querying and retrieval step into 2 groups in order to meet the different tasks in the application.

3.2.2.1. Querying and Retrieval for Finding Similar Faces

One of the requirements that the application should satisfy is to find similar faces in the database to a given face. In order to use the system, the users either upload an image or type a name to find an image. As indicated in the database construction section, the names of the people in the images are hold in the database. If the user types a name, the matching images with that name, if there is any, are brought to the user and the user selects the image to be processed among them. After upload or selection of the image, the system starts to detect the faces in the image. Then, the user is asked to select the faces which will be processed among the detected faces. The system performs a feature extraction on the image in order to retrieve corresponding images in the database. When the system gets the most similar faces to the given ones, they are presented to the user in the order of similarity percentage.

3.2.2.2. Querying and Retrieval for Face Comparison

The system also enables users to compare two faces in order to find the similarity percentage between them. In this case, the system takes two inputs to be compared from the user. As in the face similarity task, the inputs can be either an image or a text. The user can upload two images, upload one image and type a name or type two names in order to compare two faces. If any name is typed, the database is searched to find the corresponding images of the given names. If any image exists for the given names, the system compares the features of the images and selects the best percentage of similarity among the images. On the other hand, if the user

uploads two images, the system applies feature extraction on the images and compares the images in terms of their features. At the end of the comparison, the percentage of similarity between two faces is presented to the user.

3.3. Non-Functional Requirements

3.3.1. Database

The data for the system is going to be stored as tabular data in the database. Raw images are going to be stored in hard disk and the vectors corresponding to those raw images are going to be stored in the database. In order to implement the database Oracle, MSSQL or MySQL can be used. We prefer to use MySQL as it is an available open source in the University's server.

3.3.2. Scalability

We will use small data sets in order to start checking algorithms. However, the system should work with the same performance for the large data sets.

3.3.3. Response Time

The database store the vector sets that corresponds to the raw images stored in the hard disk. Even though it seems like we are dealing with images actually the algorithms that we will use work on the vectors. Therefore we expect the response time of the system to be much faster. Besides the system throughput also changes according how many features of the data are specified in the query. With more features, the number of comparisons will increase and throughput will decrease.

3.3.4. Resource Usage

This is an important requirement for our project. As the database is vector based, it does not occupy much space in the memory. Nevertheless raw images and processed images will be stored in the hard disk and as the processes number increases the usage of the memory increases as well.

3.3.5. Reliability

The collection of the useful database and the algorithms that are already proven together will form a reliable system.

3.3.6. Reusability

Reusability is a necessity not for only this project but also all kinds of software in order to be developed further by upcoming researchers. The system is extendable, for instance, new modules and features can be easily integrated to the current system. In

addition, it is a generic system, therefore for future needs new data types, classifiers, query types or any kind of software pieces can be added to the system.

3.3.7. Platform

The system is platform independent in that it can be run on both Windows and Linux platforms. It is designed as a portable system but for the under development we are planning to use C++ platform at Linux. For some of the trials we are thinking to use Matlab as it is much easier to do the calculations of the vector sets with the Matlab library. We have not decided the platform of the final release of the system.

3.3.8. Technology

C++ is going to be used to implement the system for the under development phase. The final release of the application can be implemented by PHP.

3.4. Pseudo Requirements

MySQL is designed as the remote usage. Database of the system is not portable. Therefore different users should reach the database such as via internet.

3.5. System Models

3.5.1. Use Cases

Our system has two use cases and one actor which is the client:

- Find similar faces:

The user can provide the system a photo or a name, and then see to which famous people the given person resembles the most.

- Compare faces:

The user may choose to compare a face with another face. He sees how much the faces resemble each other. He may provide photos or he may choose to enter the names of the people he wants to compare. If any image belonging to the given names is found in the database, then the comparison will be performed.

3.5.2. Scenarios

- Ali logs in to his account using his password and id. Then, he presses the “Who do you resemble?” option from the menu. He presses the browse button next to the given text field to upload his photo. After that he presses the “GO!” button to see who he resembles. Then his face appears on the screen and status information is given constantly to him. (For example: detecting the face, analyzing the face etc.).

The photos of the celebrities that he resembles the most are displayed on the screen sorted according to their similarity percentage. He finds out that he looks %64 like George Clooney, %50 like Kevin Federline, %35 like Kelly Osborne.

- Ali wants to see how much he resembles to Robert de Niro. He logs in to his account. He chooses the “Compare faces” option from the menu. He uploads his picture using the “Browse” button. Afterwards, he types “Robert de Niro” to the text field. He presses the “Go!” button. He can see status information while the system analyses the photos. The system compares the given picture with photos of Robert De Niro located in the database. The system displays the photo of Robert de Niro that Ali looks most alike. When the comparing process is done, the results are displayed below Ali’s photo. Next to Robert de Niro’s photo (which is found from the database) %23 which is the percentage of similarity is shown.

3.5.3. Object Models

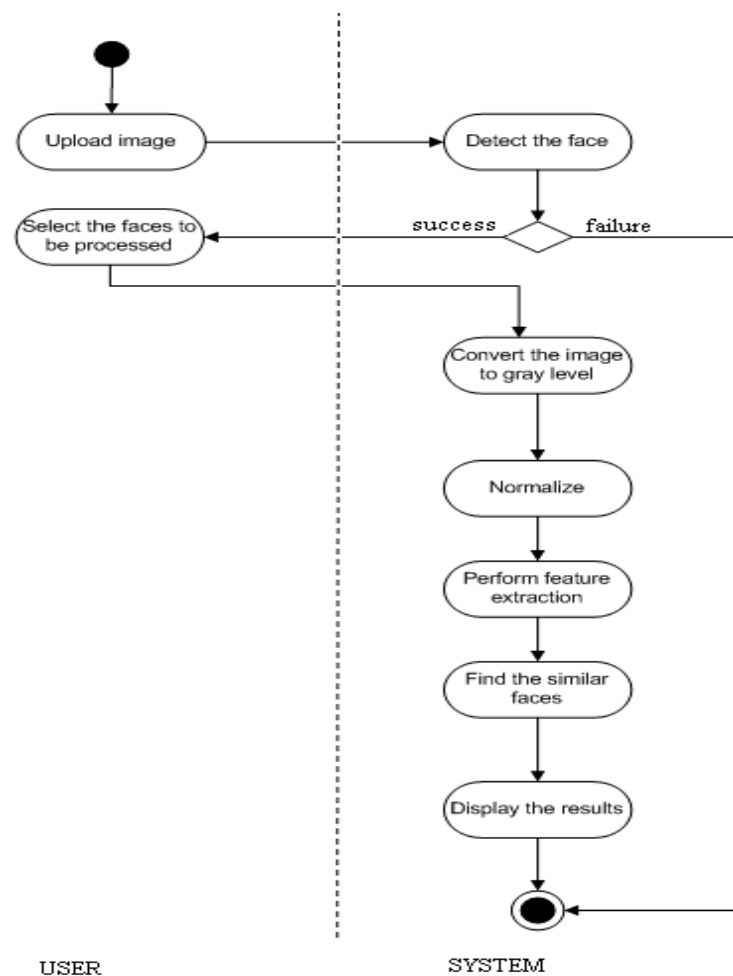
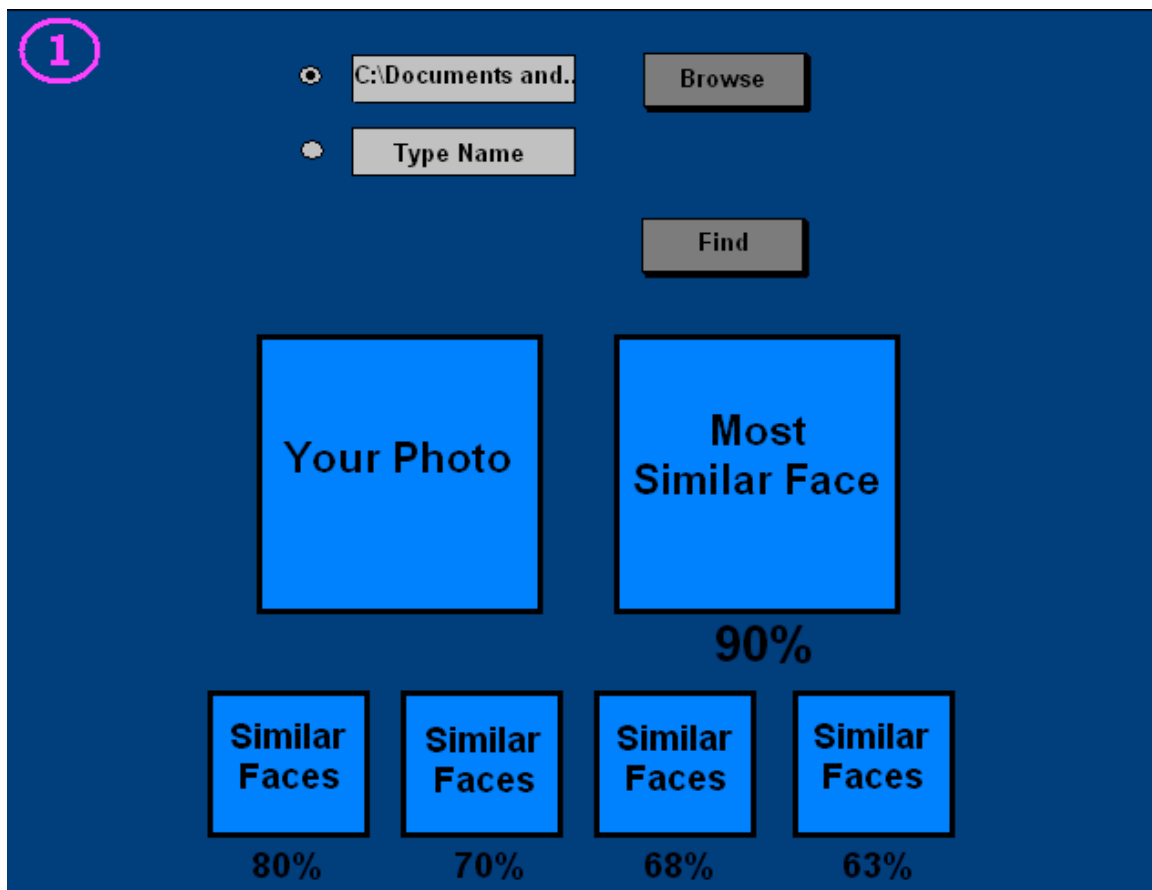
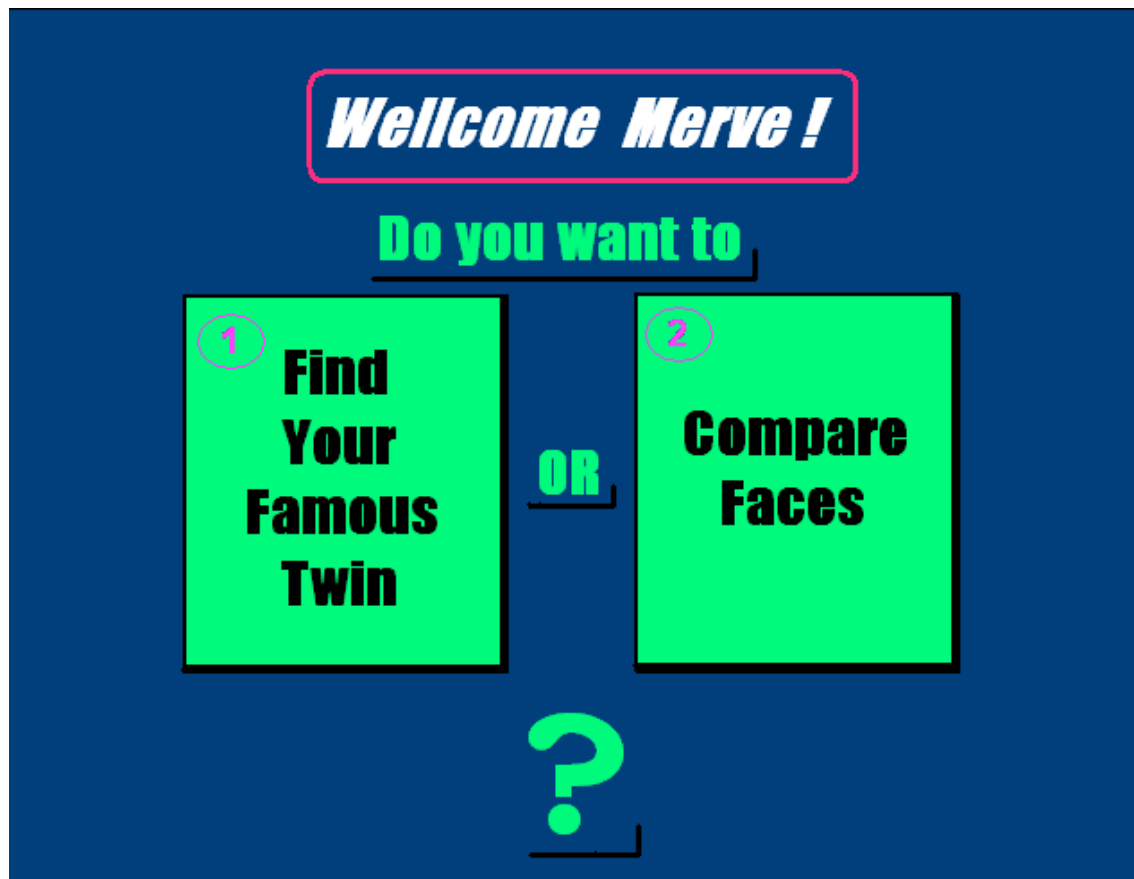


Figure.5 – Activity Diagram for the first use case

3.5.4. User Interface



2

Face 1:

☒ C:\Documents and...

Browse

☐ Type Name

Face 2:

☐

Browse

☒ Robert De Niro

Compare

Your Photo

Most Similar
Photo of
Robert De
Niro

You resemble to Robert De Niro 47% !

REFERENCES

- [1] Grgic, Mislav, and Kresimir Delac. "General Info." Face Recognition Homepage. 2005. 21 Oct. 2007 <<http://www.face-rec.org/general-info/>>.
- [2] Gross, Ralph. "Chapter 13. Face Databases" Handbook of Face Recognition Ed. Stan Z. Li, Ed. Anil K. Jain
New York: Springer, 2005. 301-328
- [3] Viola, P.; Jones, M., "Rapid Object Detection Using a Boosted Cascade of Simple Features", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, ISSN: 1063-6919, Vol. 1, pp. 511-518, December 2001
- [4] "Face Detection and Recognition." Betaface. 17 July 2007. 20 Oct. 2007 <<http://www.betaface.com/>>.
- [5] "Face Recognition Demo." My Heritage. 2006. 20 Oct. 2007 <<http://www.myheritage.com/FP/Company/tryFaceRecognition.php?lang=TR>>.
- [6] "Face Recognition." Electronic Privacy Information Center. 5 Sept. 2007. EPIC. 20 Oct. 2007 <<http://www.epic.org/privacy/facerecognition/>>.
- [7] "Facial Recognition System." Wikipedia. 18 Oct. 2007. Wikimedia Foundation. 19 Oct. 2007 <http://en.wikipedia.org/wiki/Facial_recognition_system>.
- [8] Wikipedia: "Principal Components Analysis"
<< http://en.wikipedia.org/wiki/Principal_components_analysis>>
- [9] Wikipedia: "Linear Discriminant Analysis"
<< http://en.wikipedia.org/wiki/Linear_discriminant_analysis>>

- [10] "How Facial Recognition Systems Work." How Stuff Works. 2007. 19 Oct. 2007
<<http://computer.howstuffworks.com/facial-recognition.htm>>.