

Design and Implementation of an Online Social Network with Face Recognition

Ray K. C. Lai

University of Macau, Av. Padre Tomás Pereira, Taipa, Macao, China

Email: chonchondotcom@gmail.com

Jack C. K. Tang, Angus K.Y. Wong and Philip I.S. Lei

Macao. Polytechnic Institute, Rua de Luis Gonzaga Gomes, Macao, China

Email: { jactang, kywong, philiplei }@ipm.edu.mo

Abstract—This paper presents the idea of an online social network that makes use of the face recognition technology. With the technology, friend relationship can be established without the need of having the text-based information of a user, friend recommendation algorithm can be more accurate, and face tagging can be done automatically. The design and implementation issues of such system will be discussed. The prototype of our proposed social network will be demonstrated to show the feasibility of adopting the face recognition technology in online social networks.

Index Terms—face recognition, face tagging, social networks

I. INTRODUCTION

Social network is a set of people (or organizations or other social entities) connected by a set of socially-meaningful relationships [1]. Online social network has become very popular in recent years. Some popular social networking websites have hundreds of million users registered. In this kind of sites, users can update their personal profiles, notify friends about themselves, play game and share photos with their friends internationally.

Before interacting with friends, a user must add them to form a relationship. In the existing social networking web sites, it is found that, to find a friend, the web sites are commonly based on the text-based information such as email addresses, names of friends, school names of friends, etc. Though this approach is working, in this paper, we argue that face recognition can be used to improve the friend searching and other services in online social network.

Research in automatic face recognition can date back

at least until the 1960s [2]. However, most face recognition algorithms were developed in 1980s and 1990s. Two most common of them are Principal Component Analysis (PCA) and Independent Component Analysis (ICA). Kirby and Sirovich were among the first to apply PCA to face images, and showed that PCA is an optimal compression scheme that minimizes the mean squared error between the original images and their reconstructions for any given level of compression [3][4]. Turk and Pentland popularized the use of PCA for face recognition [5]. PCA were matched the images in the database by projecting them onto the basis vectors and finding the nearest compressed image in the subspace (called eigenspace). ICA can also be used to create feature vectors that uniformly distribute data samples in subspace [6][7]. This conceptually very different use of ICA produces feature vectors that are not spatially localized. Instead, it produces feature vectors that draw fine distinctions between similar images in order to spread the samples in subspace.

Though the face recognition technology has been well established, there are no online social networks using it. The technology can provide a number of advantages to online social networks, including:

1. Friends on photos can be searched even without text-based information. It is particular useful when we only took photo with a new friend but forgot to exchange the contact.
2. The friend recommendation algorithm can be more accurate because it can make use both the text-based and face information.
3. Face tagging can be automatically done with the face recognition technology.

In the rest of this paper, we will show the design and architecture of our proposed online social network with face recognition functions. After that, we will reveal how the face recognition technology can be used to design a new friendship search algorithm, to generate friend recommendation list, and to design photo-based search and match functions. Finally, we will discuss the implementation of the system and demonstrate our prototype social network.

Manuscript received October 22, 2009; revised December 29, 2009.

Ray K. C. Lai is a research master student at University of Macau, Av. Padre Tomás Pereira, Taipa, Macao, China (e-mail: chonchondotcom@gmail.com).

Jack C. K. Tang is a research student at Macao Polytechnic Institute, Rua de Luis Gonzaga Gomes, Macao (e-mail:jactang@ipm.edu.mo).

Angus K.Y. Wong is an associate professor at Macao. Polytechnic Institute, Rua de Luis Gonzaga Gomes, Macao (e-mail: kywong@ipm.edu.mo).

Philip I.S. Lei is an associate professor at Macao. Polytechnic Institute, Rua de Luis Gonzaga Gomes, Macao (e-mail: philiplei@ipm.edu.mo).

TABLE I.
COMPARISON OF ONLINE SOCIAL NETWORKS WITH AND WITHOUT FACE RECOGNITION TECHNOLOGY

	Existing social networks	Social networks using face recognition technology
Forming a relationship	The personal information of a friend is needed	Any face of a friend appearing in any pictures of the user is sufficient
Searching and recommending a friend	Using text-based search algorithm	Using face-recognition algorithm
Tagging a person	Manually	Automatically

II. SYSTEM ARCHITECTURE

As shown in Figure 1, the system consists of the following three components:

- User Database
- Face recognition Web service
- Friendship algorithm

A. User Database

In addition to the basic information of users, the information related to the face recognition feature is needed. The formation includes:

- Face - the user face identity, type and subject.
- Photo - photo information, ID and subject.
- Photocomment - the information of each user's comment on each photo.
- Friendship - the relationship information between each user.
- Recognition - the relationship information between the faces of photo and each friend.
- Invite - the relationship information of inviting people between user and friendship tables, inviting message and status of invitation.

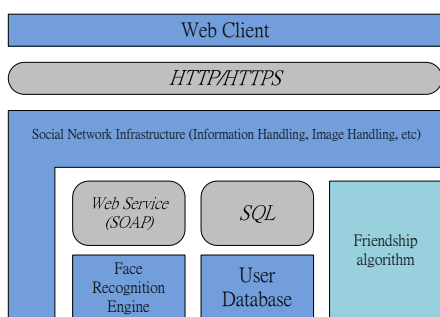


Figure 1. System architecture

B. Face Recognition Web Service

The face recognition web service handles the tasks involving face recognition. It implements with face recognition engine that face localization, enrollment and matching using digital image processing algorithm. The process of face recognition consists of a number of steps, which are shown in Figure 2.

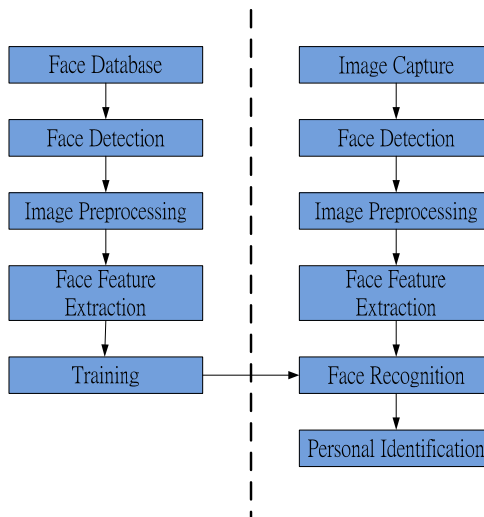


Figure 2. Common face recognition process

The normal procedures include two parts: the reference part and the identification part. In the reference process, the face recognition engine will first detect the faces for the raw image from the photo. It will then convert the photo to grayscale so as to eliminate the lighting effect and make some image process (PCA or ICA) to get a lower resolution from the face. After that, it will extract the face feature and store the face to the training database. In the face identification process, the engine will go through the same procedures to extract the face feature and make comparison with the training database. But in our implementation, we will do some normalization before store to the training database. We will normalize all face to the same resolution and size that can enhance

TABLE II.
CASES CONSIDERED IN THE FRIENDSHIP ALGORITHM

Case	Description	Unit	Weighting
My album	People appeared in my albums	No. of appearance	10
Indirect friend	Friend of my friend	No. of common friend	8
Friend's album	People appeared in my friend's albums	No. of appearance	6
Photo with me	Someone has photo that includes myself	No. of appearance	4
Co-photo	Someone and me in the same photo	No. of appearance	2

the speed of the identification face. After the face recognition step, it will provide a score for the face identity.

The face recognition service is provided by the way of Web services, which can make the system more scalability. When the system becomes larger, the face recognition process could become the performance bottleneck. Because the face recognition service is provided by web services, more face recognition engines can be added easily.

C. Friendship Algorithm

The friendship algorithm is used to provide the recommendation list for the users. The algorithm makes use the faces recognized in different cases to generate recommendation list. The cases used and the corresponding weighting for each case are shown in Table 2. Note that the exact values the weighting can be revised if necessary.

The friendship algorithm considers different case, and in each case there is a weighting associated. From the values of the weighting, it can be seen that priority has been considered in the friendship algorithm. That is, higher priority is given to the owner's direct relationship and lower priority to the indirect relationship.

As a result, a final score for each face can be generated by using the following formula:

$$\text{Final score} = \text{no. of appearance (my album)} \times 10 + \text{no. of common friend (indirect friend)} \times 8 + \text{no. of appearance (friend's album)} \times 6 + \text{no. of appearance (photo with me)} \times 4 + \text{no. of appearance (co-photo)} \times 2$$

The highest score of the face will appear the top of the recommend list, and the lower score faces will appear below the highest recommend friend according to the score.

Based on these final scores, the friendship algorithm will recommend the face to the user when the score higher than a threshold.

III. IMPLEMENTATION

A. Information Handling

The information handling process is the core development of the social network service, as shown in Figure 1. We used the PHP script language to develop this part. The functions are handling the user accounts, photo albums, searching and messaging. We use MySQL to implement our user database.

B. Face recognition process

During the face detection process, we make some optimizations:

- Convert image to grayscale.
- Extract the image with face detection.
- Resize the face to a suitable size.
- Store the face with base64-encoded string to face database.

The above optimizations can reduce the memory usage of each face and improve the performance of face matching. It can also reduce the process of storing the raw faces as the faces have been downsized. The based64-encoded string will be used in the implementation of the face reorganization web service.

Each user can store multiple face identities -- one main identity and other supplementary identities. So the user can store the current face as the main identity and some old faces as supplementary identities. These faces will store in the trained database for face matching uses.

After the comparison, the engine will give the face identity a score. We choose the score 50 as the basic recognition standard. Score 50 means nearly 0.01% of false acceptance rate (FAR). It is a very acceptable figure for the face recognition. Once the score of matching face is larger than 50, we would expect the matching face is the same as the reference face.

C. Face Recognition Engine

To implement the face recognition engine in our online social network, we use VeriLook SDK [8]. VeriLook SDK is a software development kit for face detection and face recognition. It supports multiplatform, Windows, Linux and MacOS. It also can support C/C++, C#, VB, Java and Delphi as development language. It contains two major components -- extractor and matcher. It also provides a camera manager library to support simultaneous capture from multiple cameras. The matching speed of the engine is 100,000 faces per second in 1:N identification. It supports live face detection, multiple face processing.

Our effort on this part is to make use the VeriLook SDK and to build a Web service for face recognition. We design this engine using Web service that can provide backend service with one or more servers. The advantage of making it a backend service is that if the face recognition technology changes in the future, the backend service can be replaced without the need to modify the other components.

Actually this prototype will only employ some simple procedures to detect and recognize faces. So the remote service is limited to five methods, DetectFaces, CompareFace, CompareMultipleFaces, CompareMultipleScore and CompareMultipleToMultipleFaces as shown in Figure 3. They covered all the necessary functions for face detection and recognition for our web service.

```

- <wsdl:definitions targetNamespace="http://www.ipm.edu.mo/">
+ <wsdl:types></wsdl:types>
+ <wsdl:message name="DetectFacesSoapIn"></wsdl:message>
+ <wsdl:message name="DetectFacesSoapOut"></wsdl:message>
+ <wsdl:message name="CompareFaceSoapIn"></wsdl:message>
+ <wsdl:message name="CompareFaceSoapOut"></wsdl:message>
+ <wsdl:message name="CompareMultipleFacesSoapIn"></wsdl:message>
+ <wsdl:message name="CompareMultipleFacesSoapOut"></wsdl:message>
+ <wsdl:message name="CompareMultipleScoreSoapIn"></wsdl:message>
+ <wsdl:message name="CompareMultipleScoreSoapOut"></wsdl:message>
+ <wsdl:message name="CompareMultipleToMultipleFacesSoapIn"></wsdl:message>
+ <wsdl:message name="CompareMultipleToMultipleFacesSoapOut"></wsdl:message>
+ <wsdl:portType name="ServiceSoap"></wsdl:portType>
- <wsdl:binding name="ServiceSoap" type="tns:ServiceSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http">
+ <wsdl:operation name="DetectFaces"></wsdl:operation>
+ <wsdl:operation name="CompareFace"></wsdl:operation>
+ <wsdl:operation name="CompareMultipleFaces"></wsdl:operation>
+ <wsdl:operation name="CompareMultipleScore"></wsdl:operation>
+ <wsdl:operation name="CompareMultipleToMultipleFaces"></wsdl:operation>
</wsdl:binding>
+ <wsdl:binding name="ServiceSoap12" type="tns:ServiceSoap"></wsdl:binding>
+ <wsdl:service name="Service"></wsdl:service>
</wsdl:definitions>

```

Figure 3. Web service definition

D. System Operation

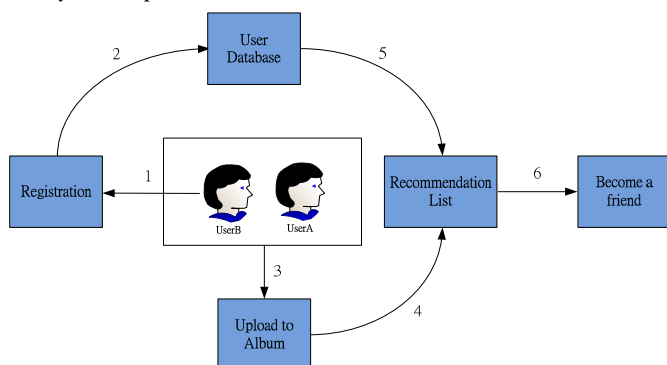


Figure 4. Establishing a friendship with the face recognition process

Figure 4 illustrate the system operation of how a user (UserB) meets another (UserA) using the face recognition technology.

1. Firstly, when UserB registers to the system, besides his basic personal information, he needs to submit a photo containing his face clearly.
2. When the system receives his face, it will look up the face in the database about UserB. If the face does not exist, it will add the face to the database.
3. Suppose UserA has taken a photo with UserB in a social event, and after the event, UserA uploaded the photo to his album in the system.
4. The system will try to detect the faces on the photos UserA just uploaded.
5. Since the system has the faces of UserA and UserB, it will put their relationship to the friendship algorithm to rank and make the recommendation lists for both UserA and UserB.
6. As a result, UserA can find UserB in his recommendation list, and vice versa. After the friend invitation by either UserA or UserB, they can become friend.

E. Testing

In the final stage of the development, a set of sample data was used for testing the functionality, acceptance and accuracy of the system. The sample data will include some text input, photo upload and comparison test. Some sample accounts and qualified photos are required for this test. We have collected the sample data of 30 users from existing social network application. The size of photo will be adjusted to around 30k for faster web page response and lower network latency. But the quality of the photos and faces must be maintained to an acceptable ratio.

The following testing procedures were run to ensure the correctness of the system:

- Create multiple user accounts.
- Upload the user photo that the face will be used as a face identity.
- Create albums with multiple accounts.
- Upload photos to multiple albums.
- Run the friendship algorithm scripts to create recommend lists.
- Make friends with or without the recommend lists.

- Click in the photo to examine the face tagging in album.

After testing the correct operations of the system, we have invited about 70 new users to use the system for testing user interface and performance. In the above testing stage, we have set up one machine for handling social network infrastructure and user database to serve web client, and set up another machine for handling face recognition engine to provide Web service. The configurations of the machines are:

- Core 2 Duo 3.0 GHz
- 1GB DDR2 Memory
- 250GB SATA II Hard Drive

IV. DEMONSTRATIONS

In this section, we will demonstrate how our online social network provides the features of recommendation list from photos, face identity database and face tagging.

A. Testing

Figure 5 shows the first page after the user logged in the system. The recommend list is shown on the lower left hand side. By clicking on the “Recognize my faces”, the user can upload photo to the face identity database. User may upload photo with the user face only or with other friends. The system will detect the faces and let the user choose the right face for the database. After this process, the user will have the face identity for the system. The user may have more than one face identity in the system.

When the user clicks the “edit” link beside the “Recognize my faces”, the corresponding face database will appear, as shown in Figure 6, so that the user can delete or choose the right head for the profile. This face database is actually a trained database of face recognition process. It stores all the possible face identities for face matching process in the friendship algorithm processing.

B. Face Tagging

Face tagging is a very useful feature in the system. The existing social network web sites commonly require a user to manually select and type name in the appropriate area of a picture. As our system employs the face recognition technology, it can recognize faces automatically without user interaction. If the user forgot the name of his friend, he can use the feature to list the name of his friend from the name tagging.



Figure 5. Web interface - my desktop

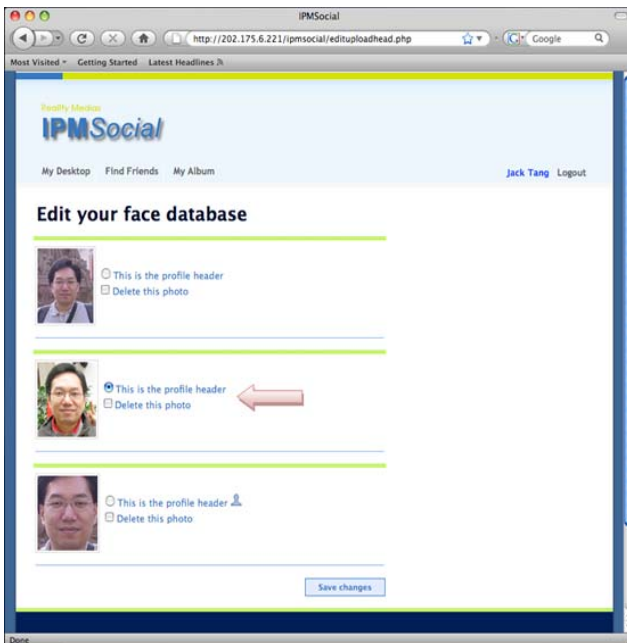


Figure 6. Web interface – face database

V. CONCLUSION

We implement a prototype to demonstrate the basic function of user registration, create and edit albums, upload and delete photos, edit some text fields, and make friends with/without recommend list.

Our online social network with face recognition technology provides features that improve user

experience and user friendly than the existing social networks. The features include recommendation list by photos, face identity database and face tagging.

ACKNOWLEDGMENT

The work described in this paper is supported by Macao Polytechnic Institute Research Grant (No.: RP/ESAP-1/2009).

REFERENCES

- [1] Barry, W. (1996). For a social network analysis of computer networks: a sociological perspective on collaborative work and virtual community.
- [2] W.W. Bledsoe, The model method in facial recognition, Panoramic Research, Inc., Palo Alto, CA PRI:15, August 1966.
- [3] M. Kirby, L. Sirovich, Application of the Karhunen-Loeve procedure for the characterization of human faces, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1990) 103–107.
- [4] L. Sirovich, M. Kirby, A low-dimensional procedure for the characterization of human faces, Journal of the Optical Society of America 4 (1987) 519–524.
- [5] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1991) 71–86.
- [6] M.S. Bartlett, H.M. Lades, T.J. Sejnowski, Independent component representations for face recognition, presented at SPIE Symposium on Electronic Imaging: Science and Technology, Conference on Human Vision and Electronic Imaging III, San Jose, CA, 1998.
- [7] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face recognition by independent component analysis, IEEE Transaction on Neural Networks 13 (2002) 1450–1464.
- [8] VeriLook SDK: <http://www.neurotechnology.com/>.

Ray K. C. Lai received the B.Sc degree in Computer Studies from Macao Polytechnic Institute, and is currently working toward the M.Sc degree in Software Engineering from University of Macau.

Jack C. K. Tang received the B.Sc degree in Computer Studies from Macao Polytechnic Institute. His research interests include online social networks and computer networks.

Angus K.Y. Wong is an associate professor at Macao Polytechnic Institute. His research interests include Internet systems, network infrastructure security, and human–computer interactions. Wong has a BSc and a PhD in information technology from the City University of Hong Kong.

Philip I.S. Lei is an associate professor at Macao Polytechnic Institute and a PhD candidate in computer science at Sun Yat-Sen University. His research interests include user interface design, social network analysis, and data mining.