Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney

Handheld Person Verification System Using Face Information

Yao-Hong Tsai[1], Raymond B.C. Fu[2], Li-Wu Huang, Ching-Chun Huang,
and Cheng-Yi Liu

Advanced Technology Center,
Computer & Communication Research Laboratories
Industrial Technology Research Institute
Chutung, Hsinchu, Taiwan 310, R.O.C.
{tyh[1], raytina[2]}@itri.org.tw

Abstract. This paper presents our newly developed wireless system for person verification based on the face verification technology. In our system, a handheld device cooperating with a client-server wireless network is used. Due to the limitation of the handheld device's computing power, most computations will be distributed to a remote control server. There are two main problems for implementing the system. One is the variations in lighting and background conditions. The other is that the camera on the handheld device can not be calibrated with the system in advance. For calibration on line, we use a three-point localization scheme for extracting appropriate face region according to the information of eyes and mouth. Furthermore, statistic based illumination normalization is used in preprocessing to decrease illumination influence under variant lighting conditions. Experiment shows the proposed system provides users a more flexible and feasible way to interact with the verification system through handheld device.

1 Introduction

Computation should be human-centered and completely integrated to human life as the Oxygen in the air that we breathe. MIT's Oxygen project [1] brings us the vision of computing in the future and Moore's law gave us the faith to make it truly possible. Through the use of handheld and embedded systems in the environment, they help us to do more while doing less.

In the scenario above, an essential component is the natural human-computer interface that provides the ability of a computer to be aware of its environment, its user, and the userís community. Face verification technology is a good choice to be the human-computer interface of the handheld computers. Since a personís face is with the person anywhere and anytime. Unlike other biometric verification requiring special hardware, face verification can be done with a low-resolution camera that is the standard equipment in handheld device. It is the reason for us to develop a handheld-based person verification system using face image.

For implementing such systems, there are two main problems when the handheld device is used. One is the variations in lighting and background conditions since the handheld device may be used in anywhere and the environment is not under well control. The other is that the camera on the handheld device can not be calibrated with the system in advance since the camera is not fixed in somewhere. Actually, it is just hold by the hand of the user.

The Project Oxygen has worked on implementing an infrastructure to allow face identification technology to be incorporated into pervasive computing applications. There are some initial efforts [2] at experimenting with face identification in handheld computing environment. They use a face detector and a face recognizer to accomplish the face identification technology. The face detector developed at Compaqís Cambridge Research Laboratory is trained by a boosted cascade of simple features [3]. The face recognizer [4] is developed at the MIT Artificial Intelligence Laboratory and the Center for Biological and Computational Learning. They try to use the powerful classifiers, Support Vector Machine (SVM) to overcome the problems for handheld face identification system.

Face detection plays an important role in many applications such as face recognition and video surveillance. It should automatically locate the positions and regions of human faces in images or videos. Since face detection is usually the first step of the above systems, it is essential that a face detection algorithm can be executed fast and accurately. As a result, the performance of face detection influences the performance of the entire systems. However, due to many kinds of variations such as lighting condition, image scale and image quality, face detection is still a very challenging problem, which requires much more research effort to improve its current performance. Over the past decade, a lot of face recognition papers have been published. Most previous face recognition systems have been implemented using stationary cameras with fixed background and lighting conditions. Face recognition will also be a very challenging problem due to the two main problems described above when handheld devices are used to capture face image.

This paper presents our newly developed handheld system for person verification based on the face verification technology. In our system, a handheld device cooperating with a client-server wireless network is used. Due to the limitation of the computing power for handheld devices, most computations will be distributed to a remote control server. In addition, we design two approaches to solve the two main problems when the handheld device is used. First, the illumination normalization based on statistic histogram equalization is used in preprocessing to decrease illumination influence under variant lighting conditions. Then, an interface for human and the handheld device is designed to calibrate the camera on the handheld device with the system dynamically. For calibration on line, we use a three-point localization scheme for extracting appropriate face region according to the information of eyes and mouth.

This paper is organized as follows. In section 2, the computing platform of the proposed system is depicted. In section 3, user interface design and preprocessing for illumination are described. Section 4 illustrates the concept of Support Vector Machine (SVM) and the training process for classifiers. Section 5 shows the algorithm of the proposed system and experimental results. Finally, the conclusion remarks will be in section 6.

2 Computing Paltform

The computing platform that we used in the implementation includes a handheld device for image acquisition and a remote server for person verification purpose. The handheld device is a Compaq iPAQ 3970 with a 400MHz StrongARM processor and 64MB of DRAM, running PocketPC 2002 OS. A CF camera is attached to the handheld device for image acquisition. Fig. 1 shows an example for the handheld device and the corresponding interface of our system. The camera we choose is VEO photo traveler for PocketPC which is capable of 640×480 resolution while the 160×120 resolution is used for our implementation.

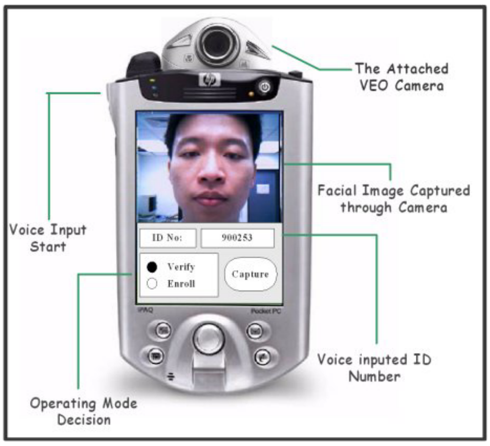

Fig. 1. The handheld device including a Compaq iPAQ 3970 and a VEO photo traveler for PocketPC

After image acquisition, the acquired image is sent through 802.11 wireless network to the remote computing server that equipped with Pentium III 800MHz processor with 512MB SDRAM, running Windows 2000 professional.

However, the speed bottleneck of the entire system is mainly on the transmission limit of wireless network. The computing time consumed on the handheld device and remote server is relatively small compared with the time we spend on the transmission between them. Thus, only a coarse face region instead of the whole image is captured to transmit to the server for person verification purpose.

3 User Interface Design and Preprocessing

In this section, the user interface design is first described then some preprocessing steps are introduced. These preprocessing steps are all performed on the server including the fine face detector and the illumination normalizer. After these preprocessing, more stable features are obtained and they are used to be the input of the face verification step.

3.1 User Interface for Calibration Online and face components Detection

In the system, the handheld device including the CF camera and the PocketPC is hold by the hand of the user. It is impossible to have calibration for the camera and the system in advance. There are three basic camera operations (Camera pan, tilt, and roll) that will influence the performance of face verification heavily. To reduce the verification errors that usually caused by the basic camera operations, a three-points face localization scheme is developed in the limit of the computing power of the handheld devices.

The centers of two eyes and mouth in the face are set to be the three points of the face localization scheme. Three points must be fitted by user manually. The distance between each pair of two points solves the problems of camera pan and tilt. The position of points solves the problems of camera roll. In the design, user sees his face on the screen of the handheld device and tries to fit each cross to its corresponding feature points as shown in Fig 2(a).

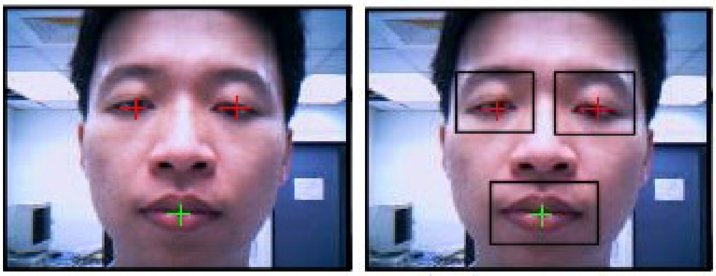
(a) (b)

Fig. 2. Three points must be fitted by user to control the basic camera operations. (a) the fitting result and (b) the face components detection result

Three cross points are drawn on the screen of the handheld device by two different colors to represent fitting positions for eye pairs (red) and mouth (green). The user will first be asked to adjust his eye's position to fit with the two red crosses on the screen. After then, the user must centers his mouth to the green cross for tilt angle adjustment. Although the precision of the fitting process above may not be well controlled, it does provide a rough region of the face image to be the input from the PDA to the remote server. It is noted that it does not need to transmit the whole image since the rough face image is already obtained.

The verification system on the server will further detect the face components in the 3×3 neighboring area to have the more precise position of the face. It includes an eye detector and a mouth detector. The training process of component classifiers is in Section 4. The result of face components detection is shown in Fig. 2(b).

3.2 Illumination Normalization

It is well known that the image gray level (or image color) is very sensitive to the lighting variation. A same object with different illumination may produce considerably different images. Psychophysical experiments show that the human visual system is difficult to identify the images of the same face that are due to considerable changes in illumination [5]. For face verification system, it is also difficult to produce good classification accuracy if image samples in the training and testing sets are taken from different lighting conditions.

The general purpose of illumination normalization is to decrease lighting effect when the observed images are captured in different environment. A common idea is trying to adjust observed images to approximate the one captured under a standard lighting condition. Most of the past works tried to define the standard lighting condition in statistically and modify the observed images to match these statistic properties. In our system, we extract the statistical histogram feature of standard lighting condition from the training images in the database. Each testing image will be adjusted based on the extracted histogram information of the standard lighting condition. Fig. 3 shows the flow chart of the illumination normalization algorithm.

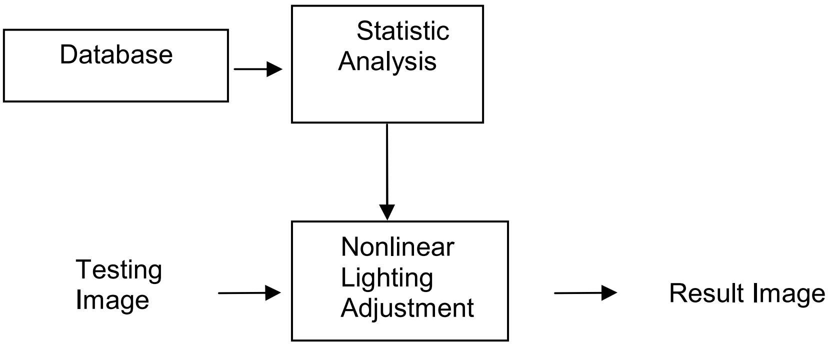

Fig. 3. The flow chart of the illumination normalization algorithm

For example, if R_a is the observed image and $h^{\acute{\imath}}_s$ is the histogram feature of standard lighting condition we get, the modification strategy [6] we used is to transform the statistic histogram of region R_a to the histogram $h^{\acute{\imath}}_s$ so that R_a has similar statistic histogram to $h^{\acute{\imath}}_s$. The transform s is the one to one function T which can be expressed as follow equation.

$$T = G^{-1} \circ H . \qquad \textbf{(1)}$$

where G is the empirical cdf of $h^{\acute{\imath}}_s$ and H is the empirical cdf of region R_a if we treat the intensity histogram at the region R_a as probability density function.

Each pixel in the region R_a is normalized by using the transfer function T. Let $R^{\hat{\imath}}_a$ be the normalized region, then

$$R^{\hat{\imath}}_a (x, y) = T(R_a (x, y)) = G^{-1} \circ H(R_a (x, y)). \qquad \textbf{(2)}$$

Ideally, $R^{\hat{\imath}}_a$ will have similar histogram distribution to the one $R^{\acute{\imath}}_a$ has. Fig. 4 shows some results of the illumination normalization algorithm.

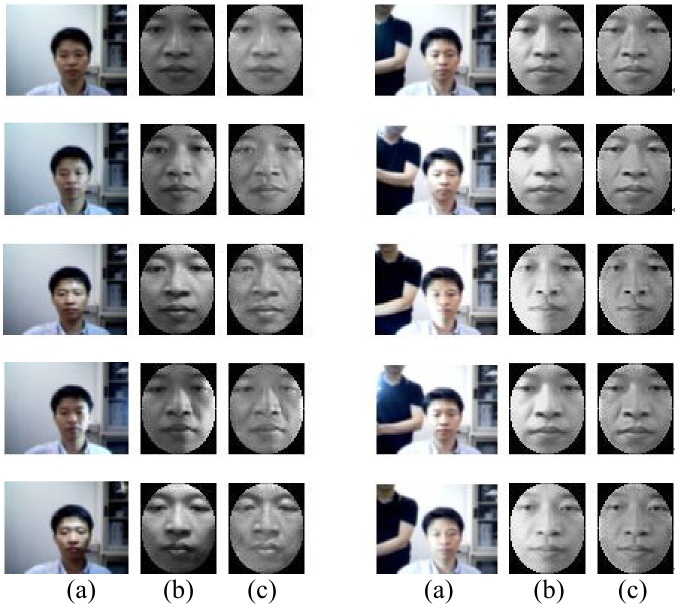

Fig. 4. Some results after the illumination-normalization mechanism. (a) The original images (b) The face images (c) The images after illumination normalization

4 SVM Calssifier

In this section, we first outline the basic theory of SVM [7]. Then, the training process for SVM classifier is described.

4.1 Support Vector Machine

The SVM has powerful capabilities in solving pattern recognition for two-class classification problems. Assume that we have *N* feature vectors $X_i \in R^n$, *i*=1, Ö , *N* with the associated label $y_i \in \{-1,1\}$, such that the training data set $\{(X_i, y_i)\}_{i=1}^{N}$ is obtained. The goal in training a SVM is to find the separating hyper-plane with maximum distance to the closest points of the training set and minimum training errors. These points are called support vectors. Generally, SVM has the following form to represent the optimal separating hyper-plane:

$$f(X) = \text{sign}(\sum_{i=1}^{N} \lambda_i^* y_i K(X_i, X) + b^*), \tag{3}$$

where $K(X_i, X) = \phi(X_i)\phi(X)$ is a kernel function and parameters are obtained by maximizing the following function

$$W(\lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j \phi(X_i) \phi(X_j), \tag{4}$$

subject to the constrains:

$$\sum_{i=1}^{N} \lambda_i y_i = 0 \tag{5}$$

for $0 \le \lambda_i \le C$ for i=1,Ö , N.

C is a parameter chosen by the user that controls the tradeoff between the margin of data points in classes and the misclassification errors. Such criterion can be solved by the use of standard quadratic programming method. According to the statistical learning theory developed by Vapnik [7], the following risk holds $\forall \alpha \in \Lambda$

$$R(\alpha) \le R_{emp}(\alpha) + \sqrt{\frac{h(\log(2l/h)+1) - \log(\eta/4)}{l}} \tag{6}$$

with probability $1-\eta$ where $0 \le \eta \le 1$, Λ is the parameter space and h is the Vapnik-Chervonenkis (VC) dimension.

4.2 Training

The training of face components detector is described first. Suppose that the image of the coarse face region is transmitted to the server. We have two (one) initial points of the eyes (mouth). The SVM-based eye (mouth) classifier is used to find the highest score of eye (mouth) component in the 3×3 neighboring area. To train the SVM model for eye (mouth) component, we begin with an eye (mouth) database. Each contains 5000 positive images and 7000 negative images to calculate the separating hyper-plane of classifiers. Each image in the database is of size 9×7 (13×7) for eye (mouth). Linear SVM classifier is used here. The negative examples are taken from scenes that do not contain any faces and a bootstrapping technique is applied to increase the number of negative examples.

Next the training process of face verification is described. To build the SVM user models, positive and negative sample features should be assigned to each user model. In our implementation, positive samples are acquired from the current user's enrolled images, negative samples are collected from all the other (exclude the current user) enrolled user's images in the database. For each user, 20 face images are captured by PDA for the training process. Polynomial kernel function with degree two is used in the training and verification process. As a user is properly enrolled, he should be able to pass the verification process guarded by the system he enrolled while invalidate users are blocked away. Some examples of the enrolled images are shown in Fig. 5. The images in the first row are captured from the handheld device. Then, they are normalized to gray images with 256 gray levels of size 25×25 to be the training samples. Some examples of them are shown in the third row of Fig. 5.

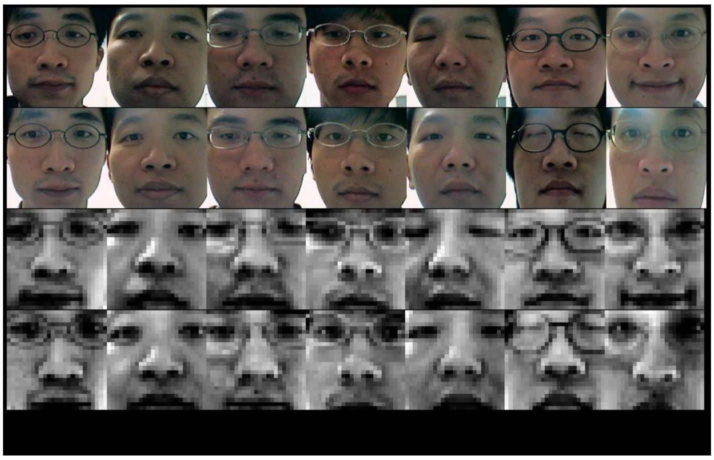

Fig. 5. Examples of images in our database for training process and verification test

5 Algorithm and Experiment Results

5.1 Algorithm

The detailed system flow chart is shown in Fig. 6.

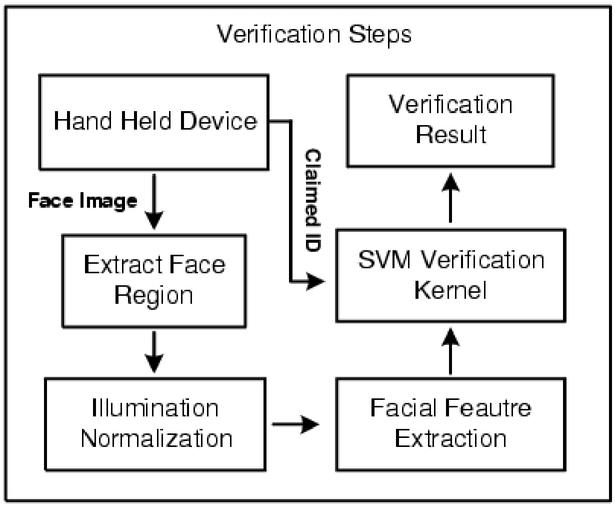

Fig. 6. The flow chart for person verification using facial image in our system

The proposed algorithm is list in the following:

Step 1. Image acquisition through the handheld device:

When a user wants to pass the verification system, he must enroll his face image through his hand-held device in advance, for example, a PDA. In addition, the system will also ask him to enter his ID number on line.

Step 2. Transfer the acquired face image and user claimed ID to the server:

When the user's face images and ID number are enrolled, the information will be further transmitted to the remote control server through wireless network.

Step 3. Extract effective face region

When the enrolled images are received by the remote control server, the image will further divided into effective and ineffective face regions according to the given eyes coordinates. To improve the precision of eyes and mouth positions, a fine face component detection algorithm is also applied before face verification to adjust the coordinates of eyes and mouth more accurately.

Step 4. Illumination normalization:

As depicted in Section 3.2.

Step 5. Facial Feature Extraction:

Normalize the gray image to size 25×25 for SVM verification kernel.

Step 6. SVM Verification Kernel:

In the verification phase, ID number and facial feature information are necessary inputs. Our system will invoke the user model corresponding to the claimed ID from the database and calculate the distance between the model and input user facial feature by SVM. If the score value is positive, it means that the user can pass the system.

5.2 Experimental Results

In our experiment, 10 individuals are enrolled in the training database and 10 images are used for each person in the testing process. The examples of testing images are shown in the second row and fourth row in Fig. 5. It is obviously that the training data is much different from the testing data, which are captured in different environment. The experimental results are shown in Table 1, which is the confusion matrix of the experiment.

Table 1. The confusion matrix of the experiment

	A	B	C	D	E	F	G	H	I	J
A	10	0	0	0	0	0	0	0	0	0
B	0	10	0	0	0	0	0	0	0	0
C	0	0	10	0	0	0	0	0	0	0
D	0	0	0	10	0	0	0	0	0	0
E	0	0	0	0	10	0	0	0	0	0
F	0	0	0	0	0	10	0	0	0	0
G	0	0	0	0	0	2	9	0	0	0
H	0	0	0	0	0	0	0	10	0	0
I	0	0	0	0	0	0	0	0	10	0
J	0	0	0	0	0	0	0	0	0	9

6 Conclusion

A biometric verification system with handheld device is proposed in this paper. Our system has a client-server framework with a handheld device to be the client and a PC to be the server. In the client side, only image acquisition is processed due to the computing and storage limitations of the handheld device. After that, the remote control server performs the fine face component detection and the illumination normalization of the extracted face regions. In the assist of our newly developed three-point face localization scheme, the possible verification error induced by basic camera operations is minimized. It indeed increases the robustness of verification. The facial features are finally feed into the SVM verification kernel to decide if the user is accepted or not. In the near future, handheld devices will have a lot more impact to our daily life. Multi-functional handheld devices will flood everyone's pockets. Its combination with the security system is an unavoidable tendency and can be expected in the future. Therefore, more research efforts should be paid to facilitate this demand. Actually, a real demo system is running at the door of our lab for on-line test. It will take three months to get enough data for performance evaluation and further research on this topic.

References

1. ìMIT Project Oxygen,î http://oxygen.lcs.mit.edu/.
2. Weinstein, E., Ho, P., Heisele B., Poggio, T., Steele, K. and Agarwal, A.: Handheld Face Identification Technology in a Pervasive Computing Environment. In Proceedings of Pervasive 2002, 48-54
3. Viola, P. and Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple Features. In Proceedings of CVPR, Kauai, Hawaii, 2001.
4. Heisele, B., Ho, P. and Poggio, T.: Face Recognition with Support Vector Machines: Global versus Component-based Approach. In Proceedings of ICCV, Vancouver, Canada, 2 (2001), 688ñ694.
5. Moses, Y., Edelman, S. and Ullamn, S.: Generalization on Novel Images in Upright and Inverted Faces. Perception. 25 (1996) 443-461
6. Phillips, P.J., and Vardi, Y.: Efficient Illumination Normalization of Facial Images. Pattern Recognition Letters. 17 (1996), 921-927
7. Vapnik, V.: Statistical Learning Theory. John Wiley and Sons, New York (1998)