Using data compression to enhance the security of identification document

Guoqin Cui[1], Aleksandar M. Ivanovic[2], Thomas S. Huang[2] , and Lizhen Chen[1]
[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China
[2]Beckman Institute at University of Illinois, Urbana, IL61801, USA
(cgq@ ict.ac.cn, gqcui@ uiuc.edu)

***Abstract*--Forging the special documents such as driver's license, visa and passport, has reached unprecedented level in recent years. In order to stop such crimes, various methods have been presented. One typical way is to do human identification if face picture is present in the documents. But it's often found that criminals modify the face picture in the documents, thus card validation purely based on face picture may also not be reliable. In this paper we present a novel method for such document authentication problem, we propose to do face image compression and store the data as barcode in an identification document, both of the barcode and original photo are printed in the document. By comparing the face image scanned from the document and the picture reconstructed from barcode, using different image processing methods to enhance the image and Gabor wavelet features, we can make a decision whether the document is valid or not.**

***Index Terms* -- Image enhancement, Identification, Data compression, Matching.**

1. INTRODUCTION

THE Identification Cards, such as visa, passport, and driver's license often use face image as important biometric features to identify the owner. Solutions to protect physical identification cards often use the information such as age, birthplace, etc. to compare with the photos in the document. But it's often found that criminals change the face image printed in the card, then if the other information are similar to the holder, it's difficult to judge whether the card is valid or not.

Barcodes are the typical way to store information. Since standard 2-D barcode, such as PDF417 [1], can store and transfer large amounts of data. It's possible to store the compressed face image data in the barcode.

In this paper we discuss a method--using compressed face data, which is stored in barcode or other place to match the image scanned from the card. Since the compressed data is difficult to be forged, and it is printed in the document, it can be used to compare with the printed face image in the card directly.

After reading the data stored in barcode or in other place, and scanning the face image printed in the card, the authentication algorithm can verify if the face image has been altered or not. Since we just compare the compressed data and the scanned face image, it avoids many difficulties of face identification, such as occlusion of hair, facial expression and illumination divergence, etc.

Certainly, since the scanned photo is often contaminated by noise introduced while scanning, and the compressed face is degraded due to loss compression, in order to make them be comparable, some image enhancement and nomination should be done at first.

The procedure is visualized at Figure 1. In next section, the compression methods will be discussed; the experiment database and the scanned images are presented in the third section. For different type of images, the pre-preprocessing steps are explained in the last part of the third section. In the last the matching methods will be introduced and conclusion will be drawn.

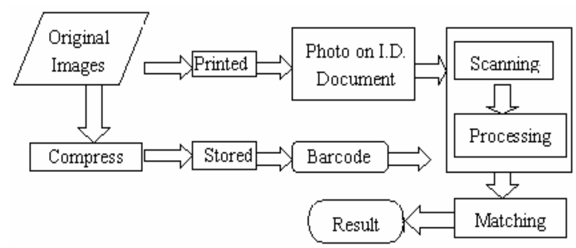

Figure 1: The procedure of authentication of the face image on the identification card

2. IMAGE COMPRESSION ALGORITHMS

For face image compression or recognition, we often select principal component analysis (PCA) or Eigenface [5] to compress face images. PCA is often taken as the most effective method for image compression, but it has two limitations: First, the Eigen-features based methods need to be trained with a large database as input; second, normalization of the face images such as eye localization is required. JPEG and JPEG2000 avoid such problems, it needn't consider eye localization, and can also restore high quantity image with high rate of compression.

In order to know which one is best as a compressed method, we will first do some experiments to compare these three compression methods.

2.1. Some definitions

One standard distortion measure of the quality of a reconstructed image compared with an original image is peak signal to noise (PSNR). Reconstructed images with higher metrics are considered better quality in a way. Assume we are given a source image X that contains N pixels and a reconstructed image Y where it's reconstructed by decoding the encoded version of X. Error metrics are computed on the luminance signal only so the pixel values X range between black (0) and white (255). Suppose the two image vectors are $(x_1, x_2, \cdots, x_N)$ and

$(y_1, y_2, \cdots, y_N)$, the formula for the PSNR compression efficiency in dB, is shown in Eq (1), where b is the bit depth of the original image. RMSE expresses the root mean square error.

$$PSNR = -20 * \log\left(\frac{RMSE}{2^b - 1}\right) \quad (1)$$

The RMSE and standard deviation (std) are

$$RMSE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}\left(x_i - y_i\right)^2}$$

$$std = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^2}, \text{ with } \bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

2.2. Original images and the corresponding Jpeg2000, Jpeg and Eigenface

We first use jpeg [3] [7], jpeg2000 [4], and Eigenface [5] to get different face images from the same original ORL database [2].

The sizes of a single image in average of different formats are as table 1; the original bmp image is 11,382 bytes. From the sizes and the face compressed visual results, we can find Eigenface need only a few bytes, jpeg can get high quality, but it still need more bytes to store the image data. Jpeg 2000 is a good-compressed method, it can get high quality visual image, but the compressed rate is between jpeg and eigenface. Besides that JPEG 2000 needn't normalize the image such as eye location, and train the large database to get the base matrix. So we choose JPEG 2000 as the compression method in our experiments.

Table 1 Average size and PSNR of different images to the original image

	Average Size (Bytes)	Average PSNR (dB)	Std of PSNR
Eigen,Dim=64	576	32.1126	0.6133
Eigen,Dim=100	900	32.8341	0.6026
Eigen,Dim=120	1,080	33.2094	0.6369
Eigen,Dim=240	2,160	35.7959	1.4480
JPEG	2,354	36.5472	0.7841
JPEG2000	984	35.8867	0.9748

3. EXPERIMENT SETUP AND FACE IMAGE PROCESSING

Original image (see figure 3(a)), after being printed in the document, and using scanner to digitalize, has changed great. We want to use the compressed data (see figure 3(b)) from the original image stored in barcode, web or other place to compare with the digitalizing data gotten by different scanners.

Figure 2 selected 9 original images from FERET database

In order to know the influence to the experiment result of different scan equipments. We first select 9 pictures shown in figure 2 from FERET face database [6] as original images. Two printers with different technologies print these images: Tektronix Phaser 450 (tk450) uses dye sublimation, Hewlett Packard (hp) uses color laser technology, then, we get eighteen printed images, then scan them by two different scanners: Canon scan Lide 30 (ca) of CIS technology and Microtek ScanMaker X6 (x6) of CCD technology.

After printing and scanning, we get 36 different images; they are labeled with "hp_ca", "hp_x6", "tk450_ca", and "tk450_x6". The JPEG 2000 compressed images such as figure 3(b) from the original nine images and the original 36 scanned images formed our test images database.

To the scanned data, the first problem tackled is copy location in a noisy environment due to the document and scanning process. We suppose the location of the photos printed in the document has a prominent notation, e.g. there is a decisive blue border around the passport photo, then the alignment reduces to finding the rotation angles, and scaling according to the length and width of the original image.

(a) (b)

Figure 3: Face images and its compressed images using Jpeg 2000(a) Original Image, BMP, 1153KB,(b) Jpeg 2000, 3.80 KB

(a) HP-CANNON (b) HP-X600 (c) TK450-X6 (d) TK450-CANNON

Figure 4: Images obtained using different scanner and printer combination

(a) (b)

Figure 5: (a) Scanned image and (b) the image after histogram equalization.

Using the special label to compare features of the sub-image with features of the full-image, the images can be found and got as figure 4.

In order to facilitate comparison between the scanned images and the compression data, it's necessary to do some preprocessing to enhance the image quality. The goal is to compare the two images from different source; we convert the color images into gray-scale first and then do processing as follows.

3.1.Image Enhancement by Gauss average filter

To the scanned image, first processing is doing the Gauss 5x5 average filtering. The central pixel is replaced by the average of itself and its neighborhood. The neighborhood is multiplied by a set of integer weights. These weights are then multiplied by pixels surrounding the central pixel and divided by the sum of the weights for normalization. Since there is usually some noise incorporated, the averaging filter can smooth the scanned image. The next steps are to both of the compressed image (as figure 3(b)) and the scanned image (as figure 4) after the above gauss filtering.

3.2. Gauss-Laplace transforms.

Performs the Gauss-Laplace filter, the edge value of the neighboring pixels in both the scanned images and the compressed images is subtracted from the edge value of central pixels, the weights are to be multiplied by pixels surrounding the central pixel and dividing by the sum of the positive weights normalizes this sum. Consequently, for regions of the image that are uniform in brightness, the result of applying this transformation is to reduce the gray level to zero. When a discontinuity is present within the neighborhood in the form of a point, line or edge, the result of the Laplacian is a non-zero value.

3.3. Gamma correction.

Using Gamma correction to adjustment the nonlinear brightness adjustment. Brightness for darker pixels is increased, but it is almost the same for bright pixels.

3.4. Histogram equalization

After the above transforms, in order to create images with equally distributed brightness levels over the whole brightness scale. We use histogram equalization to enhance the contrast for brightness values histogram near maximum, and decreases contrast close to minima. Figure 5(b) shows the image after histogram equalization.

4. Feature Evaluation and Image Matching

We use Gabor wavelets to extract the image features for the image matching [10]. The response image of the Gabor filter can be written as a correlation of the input image $I(x)$, with the Gabor kernel $g_k(x)$,

$$G_k(x_0) = \iint I(x)\, g_k(x - x_0)\, dx\,,$$

Where the Gabor filter is:

$$g_k(\vec{x}) = \left(\frac{|k|}{\sigma}\right)^2 \exp\left(-\frac{|k|^2}{2\sigma^2}|\vec{x}|^2\right)\left(\exp(ik\vec{x}) - \exp\left(-\frac{\sigma^2}{2}\right)\right),$$

$$with \quad k = \begin{pmatrix} k_v \cos\varphi_u \\ k_v \sin\varphi_u \end{pmatrix}, \quad k_v = 2^{-\frac{v+2}{2}}\pi\,, \quad \varphi_u = u\frac{\pi}{m}.$$

The above formula can be regarded as two parts, one is a symmetrical Gaussian function centered by origin point with bandwidth $\sigma' = \sigma / k_v$. We use parameter $\sigma = \sqrt{2}\pi$ by experience. Parameter σ' represents the spatial analysis resolution, the lower of σ' the higher of spatial resolution. Its real and imaginary components are 2D cosine and sine wave respectively. Parameter m refers to the number of orientations from 0 to π. We choose m equates to 8, four values of orientation are used: $0, \frac{\pi}{8}, \frac{\pi}{4} \cdots, \frac{7\pi}{8},$, then parameter u is from 0 to 7. Parameter v is related to wavelength.

In our implementation, 40 Gabor wavelet coefficients are calculated in 20 locations, which are automatically defined based on the geometric features in the upper face wave numbers. Therefore, for each location, we have 40 Gabor wavelet coefficients. After initializing the templates of the permanent features in the first frame, geometric facial features and Gabor wavelets coefficients are automatically extracted the whole images sequence.

We use MSE, PSNR and normalized correlation measures to match the images; from the experiments we have done, the normalized correlation proved to perform best over all other methods considered. Normalized correlation is calculated using:

$$S(\alpha,\beta) = \frac{\sum_{i=1}^{n} \alpha_i \cdot \beta_i}{\sqrt{\sum_{i=1}^{n} \alpha_i^2} \cdot \sqrt{\sum_{i=1}^{n} \beta_i^2}}$$

where $\alpha = (a_1, a_2, \cdots, a_n)$, $\beta = (b_1, b_2, \cdots, b_n)$ are two image data vectors after preprocessing.

5. Experiments

We use normalized correlation to compute matching scores between nine compressed images from the original images by jpeg 2000, name bj1, bj2…bj9, and 36 printed and scanned images. The matching scores of the image of "bj" with "hp_ca" can be seen in table 1, the meaning of "bj" and "hp_ca" can be found in paragraph 3. Figure 6 are the comparison of the similarity of the printed and scanned images with the images after jpeg 2000 compressed. In figure 6(a), for an example, the first column is the similarity value of the image bj1 with hp_ca01, they are from same original image which is labeled with "_sam", the second column is the maximum similarity value of bj1 with the other image hp_ca and it is labeled with "_dif", the others have same meaning.

In table 1 and figure 6(a), it's easy to know the matching scores of the images through printed and scanned and from jpeg 2000 compressed are much high than from the different original image: the lowest matching scores from the same image is 0.644876 (bj5 with hp_ca05), the highest of the value from different original image is 0.515849(bj9 with hp_ca08), thus if we choose 0.60 as the threshold, the bj1 can be right matched with the printed and scanned photos in the document. To the different scanned and printed image we have the same conclusions as shown in figure 6(b), (c), and (d).

Table 1 Matching scores of compressed data with the image after hp printed and scanned.

	bj1	bj2	bj3	bj4	bj5	bj6	bj7	bj8	bj9
hp_ca01	**0.686329**	0.409254	0.39606	0.402071	0.218212	0.448875	0.422771	0.398833	0.410577
hp_ca02	0.395568	**0.725725**	0.425169	0.418063	0.298324	0.415238	0.431484	0.409622	0.426095
hp_ca03	0.286497	0.34283	**0.700383**	0.262988	0.314675	0.362509	0.323158	0.325113	0.290154
hp_ca04	0.434418	0.447976	0.385578	**0.760972**	0.24306	0.407554	0.4698	0.403905	0.471233
hp_ca05	0.335504	0.407718	0.401915	0.35323	**0.644876**	0.401746	0.372988	0.371015	0.368877
hp_ca06	0.375214	0.378255	0.407958	0.336775	0.315539	**0.685015**	0.37121	0.375779	0.364192
hp_ca07	0.451809	0.449286	0.432436	0.47016	0.220035	0.433671	**0.774586**	0.431328	0.515849
hp_ca08	0.360716	0.413959	0.401546	0.362534	0.261525	0.417964	0.381992	**0.684855**	0.394153
hp_ca09	0.379437	0.40279	0.370419	0.422667	0.224366	0.393738	0.448855	0.385184	**0.699005**

6. CONCLUSION

In this paper we proposed an authentication technique for protect the security of identification card by compressed data. Since a new 2-D barcode standard can store more bytes, it's possible to use the information printed in the card directly to protect its' own security. The image by scanning the identification card always has much noise, and the quality is always very low, thus we must use a series of image enhancement methods. Next step we will do more experiments in different database, and find other applications of these methods.

ACKNOWLEDGEMENTS

This work was partially supported by China NSF grant 60473043, 60303018, CSC grant 22491025 and Proximex Corporation, USA.

REFERENCES

[1] http://www.mecsw.com/specs/pdf417.html.

[2] http://www.uk.research.att.com/facedatabase.html

[3] G. K. Wallace, "The JPEG Still Picture Compression Standard", Communications of the Association for Computing Machinery, Vol. 34, No. 4, 1991, pp. 30-44

[4] C. Christopoulos, A. Skodras, and T. Ebrahimi. "The JPEG2000 Still Image Coding: An Overview", IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, pp. 1103-1127, November 2000

[5] M.Turk and A.Pentland,"Eigenface for recognition", Journal of Cognitive Neuroscience, Vol.3, pp71-86, 1991.

[6] P. J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi, "The FERET evaluation methodology for face recognition algorithms", IEEE Transactions on Pattern Analysis and Machine In-telligence, Vol. 22, No. 10, October 2000.

[7] A.N. Netravali and B.G. Haskell, Digital Pictures: Representation, Compression, and Standards (2nd Ed), Plenum Press, New York, NY (1995).

[8] Milan Sonka, Vaclav Hlavac, Roger Boyle: Image Processing, Analysis, and Machine Vi-sion, Brooks/Cole Publishing Company, 1998, Second Edition.

[9] Gradshteyn, I. S. and Ryzhik, I. M. Tables of Integrals, Series, and Products, 6th ed. San Diego, CA: Academic Press, pp. 1114-1125, 2000.

[10] Turner, M.R. "Texture discrimination by Gabor functions", *Biological Cybernetics* 55, pp. 71-82, 1986.

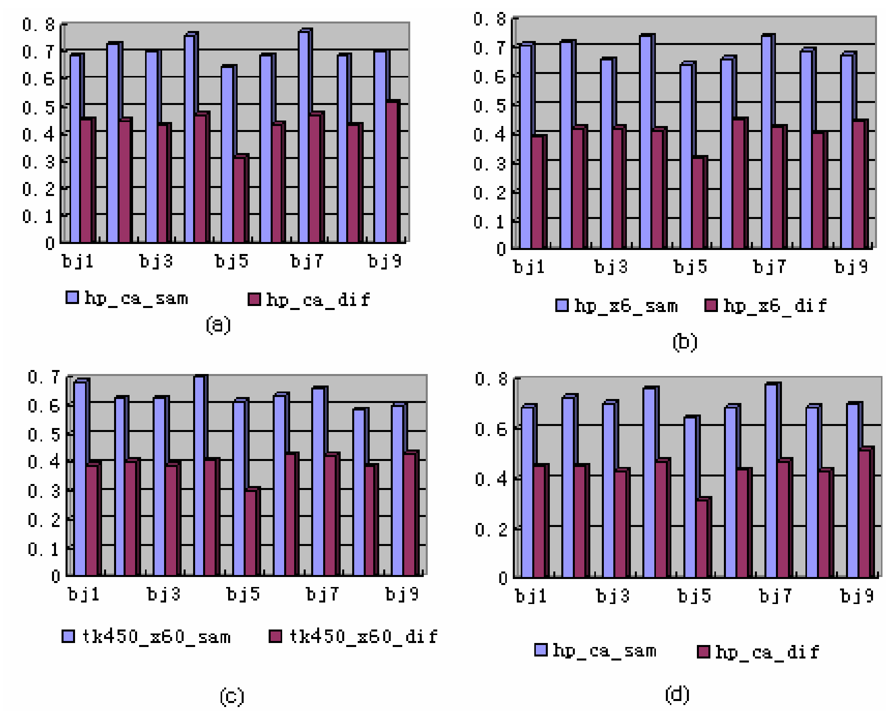

Figure 6. Comparison of the similarly of the printed and scanned image, the high is to the same number, the low is of the top value with the different number.