CRYPTOGRAPHICALLY SECURE IDENTITY CERTIFICATES

Darko Kirovski and Nebojša Jojić
Microsoft Research, One Microsoft Way, Redmond, WA 98052

ABSTRACT

We present FACECERTS, a simple, inexpensive, and cryptographically secure identity certification system. A FACECERT is a printout of person's portrait photo, an arbitrary textual message, and a 2D color bar-code which encodes an RSA signature of the message hash and the compressed representation of the face encompassed by the photo. The signature is created using the private key of the party issuing the ID. Verification is performed by a simple, intelligent, and off-line scanning device that contains the public key of the issuer. The system does not require smart cards. More interestingly, the ID does not need to be printed by a high-end printer, it can be printed anywhere. We present a novel algorithm for compressing faces and investigate the reliability of the crucial components of the system.

1. INTRODUCTION

A typical identity certification such as a driver's licence, passport, or visa, consists of a personal portrait photo, an arbitrary message, and one or more features whose purpose is to guarantee authenticity. Commonly, authenticity is assured using sophisticated printing procedures that are difficult to replicate: holograms, watermarks, micro-printing and threading, special print paper, and chemical coating [1]. Modern printing technologies have made high-quality printing devices relatively inexpensive. The availability of such printers has rendered forging most personal ID documents a relatively simple task with results often perceptually comparable to the originals. Authentication of imprinted features via electronic devices is complex and most importantly, expensive [1].

On the other hand, authenticating all-digital IDs such as smart cards or lasercards can be made highly reliable using off-the-shelf public cryptography [2]. Typically, the stored photograph and the textual message are hashed and then signed using the private key of the issuer. In-field authentication is performed using the public key of the issuer by a verification device, which also must display the signed data. While the security of such systems can be made to follow strict security standards, their cost makes them undesirable for widespread applications. A smart card costs about $10–35, while a lasercard reader costs about $2400.

1.1. FACECERTS

In this paper, we present FACECERTS, an inexpensive, paper-based but cryptographically secure, identity certification system. FACECERTS rely on public-key cryptography for security, while deploying a standard-quality low-cost color printing process which keeps the cost of printing a FACECERT two orders of magnitude lower than that of a smart card. Issuing and verification of FACECERTS is illustrated using Figure 1.

Information that is certified on a FACECERT is both photographic and textual. The digital photo is a portrait of the FACECERT holder. The textual data can be of arbitrary length and is printed on the ID. The ID is certified in the following way. First, the textual data is hashed using a cryptographically secure hashing algorithm such as SHA1 [7]. The resulting 160-bit hash is denoted as t. Next, the facial features on the photo are compressed using an algorithm that identifies the facial structure and compresses its features. A novel symbiotic eigenface-DCT based algorithm for face compression is a crucial component of this system detailed in Section 3. The best-effort output of the face compression step, denoted as f, is constrained to 3Kbits. Compression quality affects system performance for two main reasons: first, to impose low likelihood of a false negative or positive during detection and second, to set the level of detail to which an adversary, whose photo has not been taken for the ID, must resemble the facial features of the person on the authentic FACECERT.

Messages f and t are merged into a message $m = f \triangle t$ using a reversible operator $\triangle$ such that $(\exists \triangle^{-1}) f = m \triangle^{-1} t$. This operator ensures that each bit of m is dependent upon at least one bit from f and all bits of t. An example of such an operator is stream encryption [7], where message m is encrypted using t as a key. Note that the purpose of using an encryption function is not related to security, it is rather a way to make the final message m dependant upon t, which is read error-free at the verifier, without using bits to store t in m. Then, m is divided into 1023-bit parts, where each of them is signed with the private 1024-bit RSA-key of the FACECERT issuer. The signature is encoded and printed as a 2D bar-code onto the FACECERT. Two aspects of printing and scanning are important: *degradation of printed color* and *scanning reliability*. Studies have shown that state-of-the-art inks have an estimated life of 65 years on a cotton paper in average indoor display without noticeable fading and several years of corresponding outdoor lifetime [8]. The second requirement has been already addressed in modern bar-code standards such as PDF417.

A FACECERT verifier initially scans all three printed components: the photo, the text, and the bar-code. The bar-code is decoded into the originally printed signature. The scanned textual data is also converted into a text-string using reliable optical character recognition. For successful verification of a FACECERT, the text and the barcode need to be read without errors. Next, after encrypting the signature with the public RSA key of the issuer [2], the verifier obtains the signed message m. After the verifier hashes the text to obtain t, it computes $f = m \triangle^{-1} t$. Then, the verifier decompresses f into a subimage of the original photo that contains the facial features. Finally, the verifier quantifies the level of similarity between the decompressed and scanned face. If the two images are similar within the maximum tolerable compression-print-scan noise, only then the FACECERT is reported as authentic.

The security of FACECERTS stems from the fact that changing a single bit of the textual message or altering the photo beyond the compression-print-scan noise causes a global change in the bar-code that appears to be random without the knowledge of the issuer's private key.

In this manuscript, we focus on the two crucial components of the system, a novel face compression algorithm and a statistical metric for computing the similarity between an original and corresponding compressed face in the presence of the print-scan noise. The basic requirement for the face compression algorithm in the FACECERT system is to compress an image of a face into only several thousand bits with preserved sharpness of the main facial characteristics. We present a novel face compression technology based on eigenfaces [4] and improved variants of principal component analysis [5]. The PCA-based compression algorithm is combined with DCT based compression of the portion of the image represented by the low-energy subspace dimensions. This step drastically reduces the storage requirement at the verifier at marginal reduction in compression quality. We show that our technology can achieve the desired compression rates even when the component analysis is trained on a small database of images.

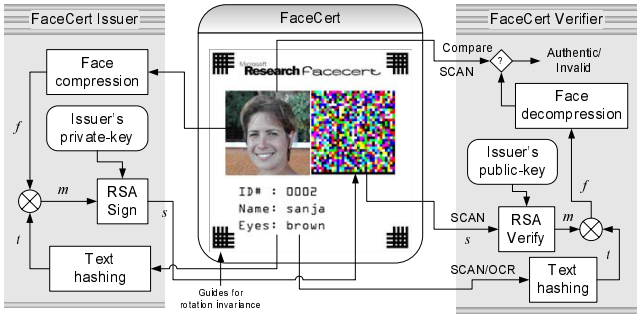

Fig. 1. Functional block diagram of the actions taken at the issuer and verifier of FACECERT IDs.

2. RELATED WORK

The idea of using digital technology and cryptography as key to enabling low-cost photo identification is not new. The system presented by O'Gorman and Rabinovich [3] is the most related to our work as it aims at the same goal - however, it relies on signing *image digests* which are tolerant to scanning errors *instead* of actual compressed images. We have shown a successful attack on this system that manipulates an image using a simple procedure so that its digest equals the digest of another distinct facial photograph [13]. By using a compressed version of the facial structure within an image, such attacks are reduced to seeking perfect human look-a-likes. This is a limitation of the distinctiveness of a human face, not only the FACECERT system. To address this issue, we have added biometric information in one version of the FACECERT system which is out of the scope of this paper.

Another alternative to FACECERTS is biometric recognition. The three most important disadvantages of almost all biometric recognition systems are: (*i*) *reliability* does not stay constant as the system scales up, which commonly renders these systems highly prone to false alarms [6], (*ii*) *centralized decision making* – the verifier needs to be connected to a central trusted server which actually performs the identification, which implies: (*iii*) *high cost* – the equipment performing the verification is costly. For most applications, such solutions are inconvenient, costly, and most importantly, unreliable. Finally, FACECERTS achieve the same level of security as smart cards, at a significantly lower cost of maintaining the issuing and verification infrastructure.

3. FACECERTS - FACE COMPRESSION

As faces form a class of images with substantially smaller variability then the class of all natural images, they can be compressed better by using a class-specific compression scheme than using general-purpose compression algorithms, such as JPEG. To develop such a scheme, we need to model the variability of facial images, i.e., the probability distribution $p(\mathbf{g})$, where $\mathbf{g}$ denotes the vector of pixel intensity in a facial image. Then, according to Shannon's coding theorem, the code length for the image g is bounded bellow by $-\log_2 p(\mathbf{g})$ bits. To build this distribution, we focus on 2D subspace models.

The problem of subspace learning can be elegantly defined in terms of a generative model that describes joint generation of the subspace coordinates, or factors, $\mathbf{y}$ and the image $\mathbf{g}$ by linearly combining image components in the factor loading matrix $\mathbf{\Lambda}$:

$$p(\mathbf{g}, \mathbf{y}) = N(\mathbf{g}; \boldsymbol{\mu} + \mathbf{\Lambda}\mathbf{y}, \mathbf{\Phi}) N(\mathbf{y}; \mathbf{0}, \mathbf{I}) \quad (1)$$

where $\mathbf{\Phi}$ constitutes the non-uniform image noise, i.e., the variability not captured in the subspace model. $\mathbf{\Lambda}$ is an nxk matrix used to expand from the k-dimansional subspace into a full n-dimensional one, where n is the number of pixels in the image $\mathbf{g}$. The parameters $\mathbf{\Lambda}$, $\mathbf{\Phi}$, and $\boldsymbol{\mu}$ can be learned by maximizing the likelihood of a set of images $\{\mathbf{g}_t\}$,

$$\log p(\{\mathbf{g}_t\}) = \log \sum_t \int_{\mathbf{y}_t} p(\mathbf{g}_t, \mathbf{y}_t), \quad (2)$$

and a good low-dimensional representation of the image tends to be $E[\mathbf{y}|\mathbf{g}]$. The above probability model, called factor analysis (FA), also allows for the design of the optimal encoding strategy for the factors $\mathbf{y}$. A realted method, principal component analysis, was used by Moghaddam and Pentland for face crecognition and compression [9]. By limiting their representation to the central part of the face they were able to represent each image in a carefully manually preprocessed database, with only 85 bytes describing 100 face factors $\mathbf{y}$. In our case, we need a more robust coding scheme that does not require precise manual registration of images, and can encode slightly more than just the central region of the face. We also include hair and the face shape, in order to lower the probability of false positive matches.

Recently, an extension of the subspace models that takes into account the possible transformation of the facial image, such as translations, rotations and scale has been proposed in [10]. In this model, called transformed component analysis (TCA), an additional random transformation variable T is applied to the image expanded from $\mathbf{y}$, and a new image $\mathbf{h}$ is observed:

$$p(\mathbf{h}, \mathbf{g}, \mathbf{y}) = N(\mathbf{h}; \mathbf{T}\mathbf{g}, \mathbf{\Psi}) N(\mathbf{g}; \boldsymbol{\mu} + \mathbf{\Lambda}\mathbf{y}, \mathbf{\Phi}) N(\mathbf{y}; \mathbf{0}, \mathbf{I}) p(\mathbf{T}).$$

Such a model, when trained on an image set tends to automatically align all images to create the most compact subspace representation. The regular subspace models, in presence of tranformational variability in the training data will tend to create blurry models, while TCA creates sharper components.

A hierarchical generative model like this is naturally suited to efficient compression, as it decomposes the variability in the data. To develop the coder, the model is first trained on a large number of face images, i.e., the subspace origin $\boldsymbol{\mu}$ and subspace vectors $\mathbf{\Lambda}$ are estimated together with the pixel noise levels $\mathbf{\Phi}$ and distribution over the used transformations (rotations, scales, shifts and deformations) $p(\mathbf{T})$. Then, for a particular image to be encoded,

the hidden variables are inferred and each of the conditional probability distributions, i.e., $p(\mathbf{T})$, $p(\mathbf{y})$, $p(\mathbf{g}|\mathbf{y})$, $p(\mathbf{h}|\mathbf{g},\mathbf{T})$, is used in an appropriate entropy coder to create codewords for describing the geometric position and deformation of the image, as well as its subspace coordinates and error vector. As the model distributions are either multinomial or Gaussian, this procedure is straightforward (for example, for a Gaussian source a non-uniform quantization is used that is fine close to the mean of the Gaussian and coarse in the unlikely areas of the subspace).

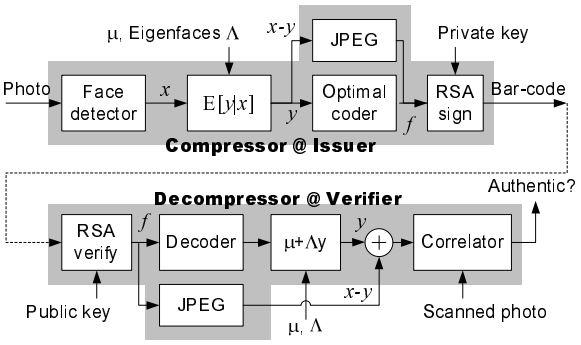

Fig. 2. Block diagram of the face compression and decompression algorithm encapsulated within the FACECERT issuing and verification system. The $\mathbf{\Lambda}$-subspace model $\mathbf{y}$ follows a Gaussian distribution and thus can be encoded close to its rate-distortion limit.

The transformation information is then combined with the face cropping information needed to capture the face from the scanned ID and encoded in the barcode, while the subspace encoding is illustrated in Figure 2. First, given an ID photograph, we identify the facial structure to be modelled $\mathbf{x} = N(\mathbf{\Lambda y}+\boldsymbol{\mu}, \mathbf{\Phi})$ with eigenfaces using a face detection algorithm [11]. Vector $\boldsymbol{\mu}$ denotes the first order statistics of the input image $\mathbf{x}$. As the posterior $p(\mathbf{y}|\mathbf{x})$ can be computed using the Bayesian rule, hence we compute:

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{x}) \quad = \quad & -\log p(\mathbf{x}) - \frac{\mathbf{1}}{\mathbf{2}}\mathbf{y}\mathbf{y}' - \frac{\mathbf{1}}{\mathbf{2}}\log(\mathbf{2}\pi\mathbf{I}) \\ & -\frac{1}{2}(\mathbf{x}-\mathbf{\Lambda y}-\boldsymbol{\mu})'\mathbf{\Phi}^{-\mathbf{1}}(\mathbf{x}-\mathbf{\Lambda y}-\boldsymbol{\mu}) \\ & -\frac{1}{2}\log(2\pi\mathbf{\Phi}) \qquad (3)\end{aligned}$$

which points to: $E[\mathbf{y}|\mathbf{x}] = \hat{\mathbf{y}} = (\mathbf{I}+\mathbf{\Lambda}'\mathbf{\Phi}^{-\mathbf{1}}\mathbf{\Lambda})^{-\mathbf{1}}\mathbf{\Lambda}'\mathbf{\Phi}^{-\mathbf{1}}(\mathbf{x}-\boldsymbol{\mu})$. Assuming $\mathbf{\Phi} = \sigma^2 I, \sigma \to 0$, we conclude that $E[\mathbf{y}|\mathbf{x}] = \hat{\mathbf{y}} = (\mathbf{\Lambda}'\mathbf{\Phi}^{-\mathbf{1}}\mathbf{\Lambda})^{-\mathbf{1}}\mathbf{\Lambda}'\mathbf{\Phi}^{-\mathbf{1}}(\mathbf{x}-\boldsymbol{\mu})$ which in the case when the basis vectors are orthogonal (e.g., $\mathbf{\Lambda}$ has been derived using PCA [5]) results in a simple least-squares approximation $\hat{\mathbf{y}} = (\mathbf{\Lambda}'\mathbf{\Lambda})^{-\mathbf{1}}\mathbf{\Lambda}'(\mathbf{x}-\boldsymbol{\mu})$. In the $\mathbf{\Lambda}$-subspace, $\hat{\mathbf{y}}$ follows a Gaussian distribution, and thus can be efficiently encoded using codes with long block lengths (for analysis see [12]), so as to approach the theoretical rate-distortion limit for the distribution illustrated in Figure 5.

The main disadvantage of the above compression method is excessive storage requirement at the decoder predominantly used to store the subspace vectors $\mathbf{\Lambda}$. For $|\mathbf{\Lambda}| = 1000$ and resolution 100×66 pixels at 1B/pixel, the verifier needs to store about 6.5MB of raw data. In order to reduce this requirement, we reduce $\mathbf{\Lambda}$ to its first $K \ll |\mathbf{\Lambda}|$ subspace vectors and encode the representation error, $\mathbf{x}-\hat{\mathbf{y}}$, using a DCT-based image codec. The decision on selecting a particular K is obtained using an exhaustive search for a desired storage/quality balance. In our experiments, we have used $K \approx 300$. An example of replacing the 700 lowest energy $\mathbf{\Lambda}$ subspace vectors with a "classic" DCT-based image codec, forced to use an equivalent number of bits, is presented in Figure 3. The details of the DCT compression mechanism and search for best K are omitted in this manuscript for brevity. Slight improvement in performance can be obtained using wavelet-based codecs.

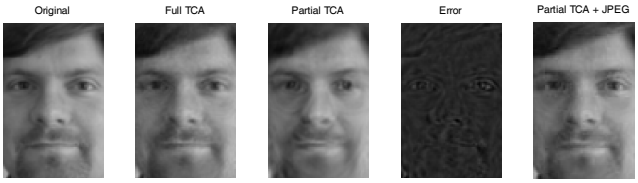

Fig. 3. Image quality comparison: original, compressed with $|\mathbf{\Lambda}| = 1000$ subspace dimensions in 3Kb, with $|\mathbf{\Lambda}| = 300$ in ≈1.5Kb, the resulting error, and the image compressed using $|\mathbf{\Lambda}| = 300$ and additional 1.5Kb for DCT-based error compression.

3.1. Face Compression Experiments

We conducted several experiments in order to evaluate system performance. We trained $\mathbf{\Lambda}$ using 400 images of 64x64 faces extracted from personal photo collections of Microsoft Research employees using a face detection algorithm that follows the work of Viola et al. [11]. The former set is noisy due to errors in alignment and about 5-10% of false positives. We tested the coding performance on the Yale and Rockefeller face databases.

Fig. 4. Five faces extracted from the Yale face database and the compressed images using JPEG (second row), PCA (third) and TCA (fourth). TCA achieved an RMSE of about ten intensity levels, considerably bellow the difference between any two images in the set. Both TCA and PCA were trained on a separate unrelated database of 400 images derived from personal digital photo collections.

In Figure 4 we show comparison between the JPEG, PCA and TCA coders on several faces in the test set. On average, at low bitrates, we were able to make JPEG encode the gray level images with 255 levels with 360 bytes and a root mean square error $rmse_{jpeg} = 36$, while both PCA and TCA performed better, with $rmse_{PCA} = 17$, $rmse_{TCA} = 10$, and with significantly lower bit rates of about 200 bytes for a 200-dimensional representation of images. TCA models used only shifts as the set of possible transformations $\mathbf{T}$. The $rmse$ differences among the images in the test set were between 35 and 65, even for images of the same people with slightly different expressions. Thus, the TCA result is well beyond the error of random photo replacement.

Figure 5 shows in red the distribution of component strengths over the coordinates in the subspace. For this distribution, the optimal rate-distortion function indicates that for the error of standard

deviation of 0.5 intensity levels (out of the 255), the number of bits needed to encode the image is about 500[1]. In other words, at 500 bits per face image, the coding error is expected to be smaller than 0.5% of the dynamic range of the image. This value is far bellow the scanning error of the system. On the same plot, in blue we plot the distribution over the subspace coordinates of images in a separate small face dataset (165 images), using the derived subspace vectors (first ten of which are shown at the bottom of the figure).

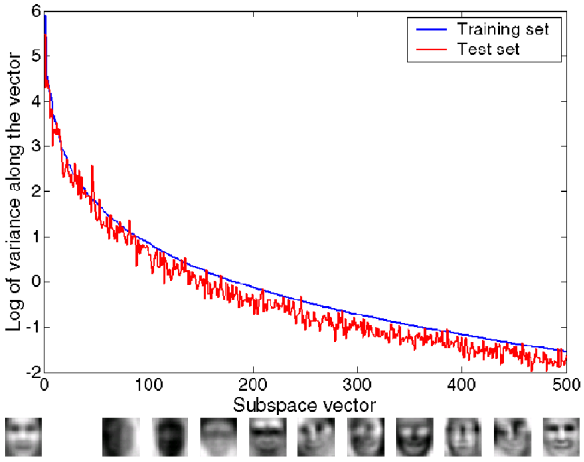

Fig. 5. The distribution over the coordinates (strengths of the subspace vectors, or principal components) for the training set (blue), and a test set (red). According to the rate-distortion analysis of the blue distribution computed on the training set of 10000 images, for errors of roughly one intensity level out of 255, the image code would be only about 500 bits long. Bellow, we show the mean and the first ten subspace vectors.

4. FACECERTS - VERIFICATION

FACECERT verification consists of simple template matching. To be in accordance with the models in the previous section, a likelihood over the windows in the image can be used as the cost instead of the template differences. For example, to use the likelihood as the similarity measure, one would take the message $\mathbf{f}$, extract the window size and detection threshold thr as well as the subspace parameters $\mathbf{y}$ to compute:

$$\log p(\mathbf{h}|\mathbf{y}) = \int_{\mathbf{T},\mathbf{g}} p(\mathbf{h}, \mathbf{g}, \mathbf{T}|\mathbf{y}), \qquad (4)$$

for all windows of appropriate size. If $\max_{\mathbf{h}} \log p(\mathbf{h}) > thr$, then the ID does contain the face encoded in the bar code. If the position of the isolated face are stored in the bar-code, the integration over transformation $\mathbf{T}$ is not necessary. Hence, our system *does not* depend on either face detection or face recognition technologies, which currently have much higher error rates. The verification simply depends on the Euclidean distance between the encoded face image in the barcode and the one on the photograph.

4.1. Empirical Error Analysis

The detection threshold thr models the tolerance of FACECERTS to a certain level of compression-print-scan noise. Large thr is likely to accommodate various classes of printers, however it also introduces higher likelihood of a false positive. Using a non-linear filter that reverses the change of the expected value of a particular gray color after print-and-scan [13], the expected Euclidean distance of the pre-processed scanned original from the decompressed photo was well within 5% of the dynamic range of a photograph in 1000+ FACECERT demo-tests. In our experiments, the likelihood of a false negative ν conditioned on a given input photo was contained within $\nu \leqslant 10^{-4}$ for $thr = 0.07$ and using a χ^2-distribution model for $\log p(\mathbf{h}|\mathbf{y})$.

The distribution over the $rmse$ distances computed over all possible pairs of six test photos paired with all the faces from our learning database of 3400 photos, is shown in Figure 6. For the adopted detection threshold $thr = 0.07$, the fitted parametric pdf to this histogram, shows numerically that the probability of a false positive is at $\psi \leqslant 10^{-6}$. Intuitively, this is a strong result as it is expected that a person finds several look-a-likes among a group of million people.

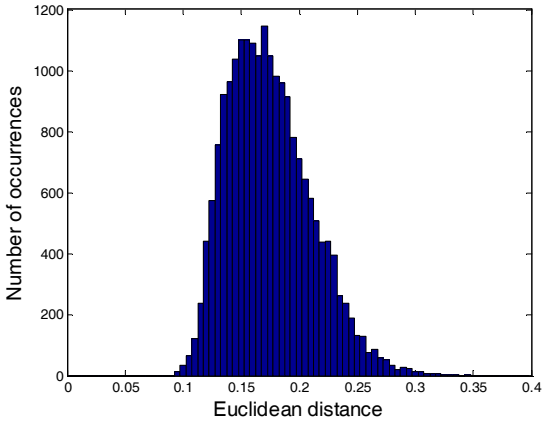

Fig. 6. The distribution of the Euclidean distance for 20,400 different pairs of facial photos. The average pixel differences are given in fractions of the dynamic range. For a detection threshold set at $thr = 0.07$, the likelihood of a false positive is $\psi \leqslant 10^{-6}$.

5. REFERENCES

[1] R.L. Van Renesse. Optical Document Security. Artech House, 1998.

[2] R.L. Rivest, et al. A method for obtaining digital signatures and public-key cryptosystems. *CACM*, vol.21, no.2, pp.120–6, 1978.

[3] L. O'Gorman, I. Rabinovich. Secure identication documents via pattern recognition and public-key crypto. *PAMI*, pp.1097–102, 1998.

[4] M.A. Turk and A.P. Pentland. Face Recognition Using Eigenfaces. *CVPR*, pp.586–91, 1991.

[5] I.T. Jolliffe. Principal Component Analysis. Springer Verlag, 1986.

[6] Biometric Consortium. `http://www.biometrics.org`.

[7] A.J. Menezes, et al. Applied Cryptography. CRC Press, 1996.

[8] Wilhelm Rsrch. `http://www.wilhelm-research.com`.

[9] B. Moghaddam, A. Pentland. Probabilistic visual learning for object representation. *Early Visual Learning*, pp.99–130, 1996.

[10] B.J. Frey, N. Jojic. Transformed Component Analysis. *ICCV*, pp.1190–6, 1999.

[11] P. Viola et al. A unified framework for face datection and recogntion. *Learning workshop, Snowbird*, 2002.

[12] A. Gersho and R. Gray. Quantization and Signal Compression. Kluwer, Boston, 1992.

[13] D. Kirovski, N. Jojic. FaceCerts. *Technical report, Microsoft Research*, 2001.

[1] Result reported for Yale database. Images in the Rockefeller database required about 1600 bits for similar performance.