GMM-SVM Fingerprint Verification Based on Minutiae Only

Berkay Topcu[1,2], Yusuf Ziya Isik[1,2], and Hakan Erdogan[2]
[1]TUBITAK BILGEM, Kocaeli, Turkey
[2]Sabancı University, Istanbul, Turkey
{berkay.topcu,yusuf.ziya}@tubitak.gov.tr, haerdogan@sabanciuniv.edu

Abstract

Most fingerprint recognition systems use minutiae information, which is an unordered collection of minutiae locations and orientations. Template protection algorithms such as fuzzy commitment and other modern cryptographic alternatives based on homomorphic encryption require a fixed size binary template. However, such a template is not directly applicable to fingerprint minutiae representation which by its nature is of variable size. In this study, we introduce a novel method to represent a minutiae set with a rotation invariant fixed-length vector. We represent each minutia according to its geometric relation with neighbors and use Gaussian mixture model (GMM) to model its feature distribution. A two-class linear SVM is used to create a model template for the enrollment fingerprint sample, which discriminates impressions of the same finger from other fingers. We evaluated the verification performance of our method on the FVC2002DB1 database.

1. Introduction

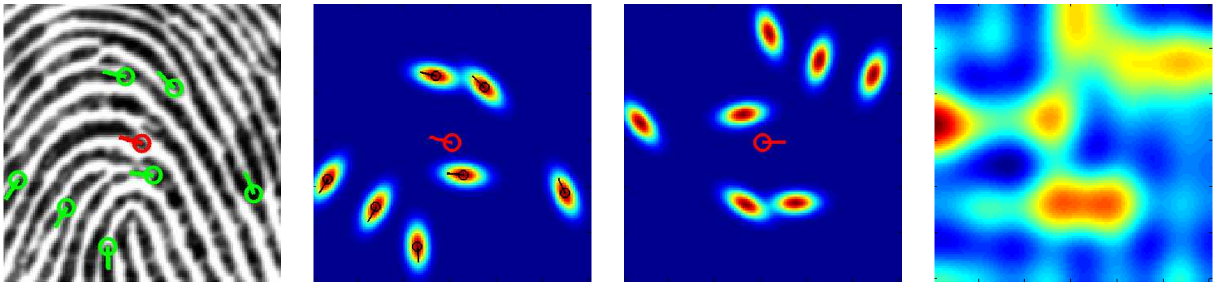
Biometric recognition systems enable fast, reliable, and secure electronic authentication, however, their large scale deployment in real world applications causes privacy and security concerns [10, 23, 24]. In the literature, several biometric template protection methods have been proposed [11] (e.g., fuzzy commitment scheme [15] and biohashing [13]) to overcome these concerns by securing biometric templates (e.g., face and fingerprint).

Recent template protection schemes require either a fixed length feature vector representation or a binarized string as input. Thus, a variable length minutiae representation of a fingerprint cannot be directly used in combination with these schemes. In addition, some template protection schemes designed specifically to work with unordered sets of varying number of minutiae (e.g., fuzzy vault [14]) experience degradation in matching accuracy due to alignment issues and nonlinear distortion [12].

One of the earliest works on fingerprint template protection has secured minutiae information x, y, θ separately [3]. In a later study, FingerCode feature (a texture based fingerprint representation without minutiae information) has been protected via biohashing [13]. Another branch of research has focused on securing each minutia separately. Yang et al. [28, 29] have proposed methods to extract a binary secure hash bit string from each minutia and its vicinity using minutiae information only. A more recent study similarly has used neighboring minutiae information along with texture information around each minutia and secured each minutia feature vector by biohashing [2]. Protected Minutiae Cylinder-Code (P-MCC) [9], one of the most accurate algorithms proposed recently, has secured each MCC structure that corresponds to a single minutia. All these studies have represented each minutia with a fixed length binary string therefore matching between variable length final templates has been addressed as a minutiae pairing problem.

In the spectral minutiae representation [27], each minutia location is coded by an isotropic two-dimensional Gaussian function in the spatial domain. Here, minutiae are represented as a magnitude spectrum and their orientations are incorporated by assigning each Gaussian a complex magnitude. Bringer et al. [4] have characterized a fingerprint in terms of its similarity to each representative local minutiae vicinities in a set of fixed size. This fixed size set has been extracted from a representative database of all existing vicinities in the world of fingerprints. For a fingerprint, a feature vector that contains similarities of its vicinities to those of the representative set has been produced.

In this work, our ultimate goal is to describe an underlying framework that enables the generation of a fixed length feature vector representation for fingerprint minutiae. The framework draws upon the work of Campbell et al. on support vector machines using GMM supervectors for speaker verification [6]. Each minutia and its neighbors within a specified radius are represented as a 2D image by placing two-dimensional Gaussians at the locations of neighbor minutiae. DCT coefficients of this patch image are rearranged based on zig-zag scanning and the first D DCT



(a) A minutia and its neighbors in *R*

(b) Gaussian representation for neighbor minutiae

(c) Neighbor minutiae at the new coordinate system

(d) Reconstructed minutiae patch image from DCT coefficients

Figure 1. DCT representation of a minutia patch image

coefficients of this patch image are used to represent each minutia as a *D*-dimensional feature vector. A single user-independent GMM universal background model (UBM) is trained from a collection of fingerprints to represent the distribution of DCT features. A fingerprint is then represented with its probabilistic alignment into the UBM mixture components and a GMM supervector is created from the stacked first order statistics of the mixture components.

For a given enrollment fingerprint sample, a two-class linear SVM is trained in order to create a model template that discriminates positive samples from negative samples. The matching between a query fingerprint and the model template is performed by computing a single inner product between the target SVM model and the query GMM supervector. The performance of our framework was evaluated in a 1-to-1 fingerprint verification setup and the results on FVC2002DB1 database demonstrated that an EER of 1.63% was achieved. The protection of the model is out of the scope of this study and left as future work.

2. DCT-based minutia patches

2.1. Minutia Patch

A minutia patch is a local representation that encodes a minutia and its geometric relations with other minutiae that are closely located. Each minutia patch consists of a central minutia m_c and its neighboring minutiae within a radius R (Figure 1(a)). In order to directly compare two minutia patches, without any registration for the relative alignment of fingerprints, an absolute representation using m_c as a reference is required. The central minutia m_c can be used to define a new coordinate system where its position would be the center of the system and its orientation would give the direction of the x-axis. In this new coordinate system, the coordinates and orientations of the neighbor points would be translated and rotated accordingly. This representation scheme is inspired from minutiae vicinities described in [4].

In this representation, a global set of minutiae is converted into a collection of several local minutiae sets and for each minutia of a fingerprint, a patch is constructed. This also enables two fingerprints to be matched by locally comparing patches pairwise and calculating their similarity score using the local scores of the best pairs. Although global coherency in the minutiae set is not utilized, the local approach has the advantage of limiting the crucial elastic distortion problem in fingerprint matching. In the local area of a patch, distortion due to the elasticity of the skin is negligible. The radius used in the local approach is of great importance. The neighborhood of the central minutia should contain several minutiae in order to be sufficiently discriminative but at the same time it should be small enough to be considered as a local area.

2.2. Gaussian minutia patch image

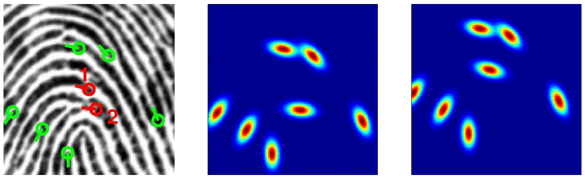
Within a specific radius *R*, the number of neighbor points of a central minutia varies and this leads to a minutia representation of unknown length. In order to obtain a fixed length representation, one can use a rectangular grid of size $(2R + 1) \times (2R + 1)$ where the central minutia is at the center. Each neighbor minutia is then inserted into this grid with respect to its relative location to m_c on the fingerprint.

Representing a minutia with a single point in the spatial domain increases the sensitivity of minutiae positions to small variations and does not maintain direction information. Instead, each neighbor minutia is represented by a two-dimensional multivariate Gaussian function:

$$f(\mathbf{x}, \mu, \Sigma) = \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)} \tag{1}$$

where $\Sigma = diag(\sigma_1, \sigma_2)$ is the covariance matrix. A Gaussian is centered at the minutia location and its covariance matrix is selected such that the major axis coincides with the minutia orientation as illustrated in Figure 1(b). For a neighbor minutia (x_i, y_i, θ_i), a template Gaussian is translated to (x_i, y_i) and rotated with θ_i. This makes the mean of Gaussian as $[x_i, y_i]^T$ and covariance matrix as $\Sigma' = A\Sigma A^T$, where $A = A(\theta_i)$ is a rotation matrix [1]. The patch image is

[1] Please note that, Gaussians are placed prior to the rotation with respect



(a) Two neighbor minutiae (b) Minutia patch for minutia 1 (c) Minutia patch for minutia 2

Figure 2. Selected minutiae patches of two neighbor minutiae from the same fingerprint image (before rotation)

then generated as a sum over these shifted Gaussians:

$$I(\mathbf{x}) = \sum_{i=1}^{N_p} f(\mathbf{x}, \mu_i, \Sigma'_i) \tag{2}$$

where N_p is the number of neighbor minutiae and $\mu_i = [x_i, y_i]^T$ is the position of the neighbor minutia with respect to m_c. Sample minutia patch images selected from a fingerprint are illustrated in Figures 2(b) and 2(c). Please note that, the central minutia is not directly included in this representation, but it defines the new coordinate system and the neighbors of the patch.

2.3. DCT representation for minutia patches

Although minutia patch images capture the required information for fingerprint matching, $(2R+1)^2$-dimensional representation for each minutia brings heavy computation and storage requirements. Discrete cosine transform (DCT) is often used in image processing, especially for lossy compression (e.g., JPEG), due to its strong energy compaction property. It expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. Since most of the signal information tend to be concentrated in a few low-frequency components of the DCT, discarding small high-frequency components results in compact representation of the signal. By keeping only the first D 2D-DCT coefficients after performing zig-zag scanning, each minutia patch image is represented as a D-dimensional feature vector.

We conducted an evaluation to assess the discriminative power of our DCT minutia patch representation. To compare two fingerprints, fp_1 and fp_2, a pairing matrix that contains similarity scores between patches of fp_1 and patches of fp_2 was constructed. The scores were computed using a decreasing function that converted the Euclidean distances between DCT coefficients to a score (i.e., $g(x) = 1/(1+e^{x/\tau})$). A closest neighbor search algorithm was applied to the pairing matrix in order to select the best association of minutiae. At each turn, a minutiae pair from fp_1 and fp_2 with the highest matching score was identified and removed from the matrix. The final score between two fingerprints was computed by accumulating the matching scores of identified pairs during the search.

In the evaluation, the FVC2002DB1 database [20] which has 8 impressions of 100 different fingerprints captured with an optical sensor was used. Following the performance evaluation protocol of FVC2002 [21], 2800 genuine and 4950 imposter comparisons were performed. An Equal Error Rate (EER) of 4.46% was achieved for $D = 50$. Although, the achieved EER was below the state of the art [20, 21], it arguably confirmed the discriminative capability of minutia patches.

3. GMM supervector training

Gaussian mixture models (GMMs) have been dominantly used for modeling in text-independent speaker verification. The distribution of features extracted from speech segments (i.e., frames of an utterance) is modeled by performing background model adaptation of GMMs. First, a universal background model (UBM) is trained from set of frames and then the speaker model for the i^{th} speaker is derived by adapting the universal background model to match the observations of the speaker. Recently, the use of GMM for modeling feature distribution has also become an effective approach for face verification [26].

Similar to the frames of a speech utterance or the blocks of a face image, minutiae points of a fingerprint are separate observations of the same underlying signal. DCT patch representation of minutiae is used to train a universal background minutiae model. The UBM is a large GMM trained to represent the distribution of features. From a huge database of fingerprints, a large number of minutiae patches are extracted as training data and they are pooled to train the UBM via EM (expectation maximization [7]) algorithm[2]:

$$g(\mathbf{x}) = \sum_{i=1}^{N} w_i p_i(\mathbf{x}) \tag{3}$$

where w_i are the mixture weights and $p_i(\mathbf{x})$ is the unimodal Gaussian density with mean $\mathbf{m}_i$ and covariance $\mathbf{\Sigma}_i$ (diagonal covariance is assumed as this requires fewer observations).

Given a fingerprint with T minutiae, $\mathbf{x}_t, t = 1, ..., T$ are the DCT minutia patches for each minutia. The estimates of first order statistics for the fingerprint data are computed for mixture i in the UBM as:

$$E_i(\mathbf{x}) = \sum_{t=1}^{T} Pr(i|\mathbf{x}_t)(\mathbf{x}_t - \mu_{\mathbf{i}}) \tag{4}$$

$$Pr(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{j=1}^{M} w_j p_j(\mathbf{x}_t)}. \tag{5}$$

to the orientation of the central minutia θ_c.

[2] For further details, please refer to [25].

Using only the first order statistics ($E_i(\mathbf{x})$), a GMM supervector is formed by concatenating the first order statistics of each mixture. The GMM supervector maps a fingerprint to a high-dimensional vector of length $D \times N$, where D is the number of DCT coefficients and N is the number of Gaussians in the mixture. Please note that, we do not perform MAP adaptation as done in [6, 18, 25, 26] for adapting a speaker model. Our experiments showed that using first order statistics without MAP adaptation performed better, so we employed first order statistics only.

4. Linear SVM training for template generation

An SVM is a two-class linear classifier constructed from sums of a kernel function K

$$f(\mathbf{x}) = \sum_{i=1}^{L} \alpha_i t_i K(\mathbf{x}, \mathbf{x}_i) + d \tag{6}$$

where t_i are the ideal outputs (either 1 or -1), d is a learned constant, $\sum_{i=1}^{L} \alpha_i t_i = 0$, and $\alpha_i > 0$. The vectors $\mathbf{x}_i$ are support vectors and obtained from the training set by an optimization process. The kernel K is constrained to have certain properties so that K can be expressed as an inner product, $K(\mathbf{x}, \mathbf{x}_i) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x})$, where $\Phi(\cdot)$ is a mapping to a higher dimension.

SVM provides a suitable solution to fingerprint verification problem, since it is fundamentally a two-class problem. We aim to decide whether the fingerprint comes from the user or the fingerprint belongs to someone else. As the number of features is large in our problem ($D \times N$), we do not need to map data to a higher dimensional space and use linear kernel (i.e., $K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x}_i^T \mathbf{x}$). In practice, the linear kernel tends to perform very well when the number of features is large. In addition, GMM supervector has already been employed as a linear kernel with a simple diagonal scaling [5, 6]. The SVM in (6) can be expressed as:

$$f(\mathbf{x}) = \sum_{i=1}^{L} \alpha_i t_i \mathbf{x}_i^T \mathbf{x} + d = \mathbf{w}^T \mathbf{x} + d \tag{7}$$

which reduces two-class classification to an inner product between the classifier model $\mathbf{w}$ and GMM supervector $\mathbf{x}$. The model $\mathbf{w}$ is solved by minimizing:

$$\min_{\mathbf{w},d} \left(\frac{1}{2} \|\mathbf{w}\|_2 + C \sum_i H_1 \left(t_i(\mathbf{w}^T \mathbf{x}_i + d) \right) \right) \tag{8}$$

where $H_1(z) = \max(0, 1 - z)$ is the "Hinge Loss" and C is a regularization parameter that controls a tradeoff between a low error on the training data and the ability to generalize well.

DB Name	#Fingers	#Fingers × #Samples/Finger
FVC2002DB2	800	100 × 8
FVC2002DB3	800	100 × 8
FVC2002DB4	800	100 × 8
FVC2004DB1	800	100 × 8
FVC2004DB2	800	100 × 8
FVC2004DB3	800	100 × 8
FVC2004DB4	800	100 × 8
FVC2006DB1	1648	140 × 12
FVC2006DB2	1680	140 × 12
FVC2006DB3	1680	140 × 12
FVC2006DB4	1680	140 × 12
IN-HOUSE	3520	440 × 8
TOTAL	**15808**	

Table 1. Number of fingerprints used in GMM training

We use SVM to create a model $\mathbf{w}$ (which we also refer to as a template) for an enrollment fingerprint sample $\mathbf{f}_{enroll}$. This is achieved by training an SVM using the GMM supervector of $\mathbf{f}_{enroll}$ as a positive sample (labeled as +1) and GMM supervectors of fingerprints from example imposters as negative samples (labeled as -1). Given a query fingerprint sample $\mathbf{f}_{query}$, its matching score for the subject i is the inner product between $\mathbf{w}_i$ and $\mathbf{x}_{query}$, where $\mathbf{w}_i$ is the SVM classifier model for the subject i and $\mathbf{x}_{query}$ is the GMM supervector of $\mathbf{f}_{query}$. The verification decision is based upon whether the score $\mathbf{w}_i^T \mathbf{x}_{query}$ is above or below a threshold. This approach provides 1-to-1 fingerprint matching since only one single training sample for each class is used to train the template model. It corresponds to comparing a gallery fingerprint to a query fingerprint as done in all other fingerprint verification systems.

5. Experiments and Results

We performed 1-to-1 fingerprint verification experiments on the FVC2002DB1A fingerprint database [20]. For minutiae extraction, a commercial fingerprint minutiae extractor (which participated in FVC-onGoing [19], Ongoing MINEX [17] and FpVTE 2012 [16]) was used to obtain minutiae information in ISO 19794-2 format (x, y, θ) [1].

In order to create patches for each minutia, all neighbor minutiae within a radius $R = 60$ pixels were used. This resulted in minutia patch images of size 121×121 pixels. For DCT representation of patches, the first 50 DCT coefficients after zig-zag scanning were kept (i.e., $D = 50$), which means that a minutia was represented along with its local information via a feature vector of length 50.

We used 15808[3] fingerprints from publicly available FVC databases and an in-house fingerprint database col-

[3] 32 minutiae in FVC2006DB1 have zero neighbors within R, therefore they were not used in GMM training.

# Gaussians	1024	2048	4096
EER	2.23%	1.85%	1.63%

Table 2. Equal error rates for GMMs with different number of Gaussians

lected via an optical reader. The details of these databases (the number of fingers and samples per finger) can be found in Table 1. Our target database, FVC2002DB1, was not included in GMM training to prevent any bias that might favor supervector representation in the advantage of the FVC2002DB1 database. The GMMs were trained for different number of Gaussians (1024, 2048, and 4096) and their results were reported separately. Once the universal models were trained, we extracted first order statistics of fingerprint samples from FVC2002DB1 and produced supervectors for GMMs with different number of Gaussians, which resulted in supervectors of dimensions 51200 (1024×50), 102400 (2048×50), and 204800 (4096×50).

For the enrollment of target fingerprints, we trained an SVM for each fingerprint sample using the target GMM supervector and the set of imposter GMM supervectors labeled as -1, using the first impression of each subject as imposters. The weight vector of the SVM classifier model was the template for the enrolled fingerprint sample. During verification, GMM supervector of the query fingerprint was compared to the template of the claimed identity and their inner product was used to give accept or reject decision.

The verification protocol was as follows:

i) Each impression was matched against the remaining impressions of the same finger. The total number of genuine tests was 5600 ($8 \times 7 \times 100$).

ii) The first impression of each finger was matched against the first impression of the remaining fingers. The total number of imposter tests was 9900 (99×100).

For both cases, symmetric matching (i.e., fp_1 vs fp_2 and fp_2 vs fp_1) was executed as $\mathbf{w}_{fp_1}^T \mathbf{x}_{fp_2}$ is different from $\mathbf{w}_{fp_2}^T \mathbf{x}_{fp_1}$.

Equal error rates (EERs) for GMMs with different number of Gaussians are shown in Table 2. The optimal C value for training SVMs corresponding to different number of Gaussians were found by grid search and best C values were 10, 0.001, and 1 for 1024, 2048, and 4096 Gaussians, respectively. As the number of Gaussians in the GMMs increases, our method performs better in representing feature distribution which eventually leads to lower error rates.

In order to provide comparison with our system, we also performed direct minutiae matching[4] with the commercial algorithm which we also used for minutiae extraction. It obtained 0.50% EER on FVC2002DB1 and performed better than our proposed method. This difference stems from the facts that we can neither perform minutiae pair search, which is a crucial step for minutiae matching, nor include singular point information. However, the main purpose of this study is to present a fixed length fingerprint representation and this performance drop was expected. Two other fixed length approaches that can be compared with our system are the spectral minutiae representation [27] and binary feature vector representation in [4]. However, they do not report error rates for the FVC2002DB1 database. When we analyzed their reported results on the FVC2002DB2 database (2.48% [27] and 3.88% [4] EERs compared to minutiae matching 1.0% on FVC2002DB2), we also observed similar performance drops[5].

6. Conclusion

The GMM-SVM based feature representation is a novel method to create a fixed length feature vector for fingerprint minutiae. Although minutiae-based matching is the most widely used technique in fingerprint verification/identification, the increasing security and privacy concerns make minutiae template protection one of the most crucial tasks. The main motivation of this study is to obtain a fixed length feature vector for fingerprints so that minutiae based fingerprint verification can be combined with template protection schemes. In addition, our method avoids the difficulties of minutiae registration by representing minutiae patches on a normalized coordinate system defined by the orientation of the central minutia. Also, the major problem of elastic distortion in fingerprint matching is compensated with the local representation of the minutiae neighborhoods.

This paper introduces a fixed length feature representation for variable length minutiae of a fingerprint. In order to combine our method with the cryptographic primitives for template protection, such as [8], one should extract bits that are stable for genuine users and completely random for arbitrary users. Another possible direction for template protection might be random projection-based biometric hashing [13], which cannot be directly applied to minutiae templates. In the future, we will work towards protecting feature vectors that are created by our approach and include the binarization of the GMM-SVM feature vectors. We also conjecture that enriching the database that is used in training GMMs and using random resampling ([22]) for addressing data-imbalance problem in SVM will be possible improvements to our GMM-SVM minutiae representation.

[4] Additional fingerprint features that are not defined in ISO minutiae template were not used in any of the experiments.

[5] We plan to analyze the performance of our system and make comparison on the FVC2002DB2 database in the future.

References

[1] Information technology – Biometric data interchange formats – Part 2: Finger minutiae data. ISO 19794-2:2011, International Organization for Standardization, Geneva, Switzerland, 2011.

[2] R. Belguechi, E. Cherrier, C. Rosenberger, and S. Ait-Aoudia. Operational bio-hash to preserve privacy of fingerprint minutiae templates. *IET Biometrics*, 2(2):76–84, 2013.

[3] R. M. Bolle, J. H. Connell, and N. K. Ratha. Biometric perils and patches. *Pattern Recognition*, 35:2727–2738, 2002.

[4] J. Bringer, V. Despiegel, and M. Favre. Adding localization information in a fingerprint binary feature vector representation. In *Proceedings of SPIE 8029, Sensing Technologies for Global Health, Military Medicine, Disaster Response, and Environmental Monitoring; and Biometric Technology for Human Identification VIII*, volume 8029, pages 80291O–80291O–10, 2011.

[5] W. M. Campbell. Generalized linear discriminant sequence kernels for speaker recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, Orlando, Florida, USA*, pages 161–164, 2002.

[6] W. M. Campbell, D. E. Sturim, and D. A. Reynolds. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13:308–311, 2006.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[8] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.

[9] M. Ferrara, D. Maltoni, and R. Cappelli. Noninvertible minutia cylinder-code representation. *IEEE Transactions on Information Forensics and Security*, 7(6):1727–1737, 2012.

[10] T. Ignatenko and F. M. J. Willems. Information leakage in fuzzy commitment schemes. *IEEE Transactions on Information Forensics and Security*, 5(2):337–348, 2010.

[11] A. K. Jain, K. Nandakumar, and A. Nagar. Biometric template security. *EURASIP Journal on Advances in Signal Processing*, 2008:113:1–113:17, 2008.

[12] A. K. Jain, K. Nandakumar, and A. Nagar. *Security and Privacy in Biometrics*, chapter Fingerprint Template Protection: From Theory to Practice, pages 187–214. Springer London, London, 2013.

[13] A. T. B. Jin, D. N. C. Ling, and A. Goh. Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recognition*, 37(11):2245–2255, 2004.

[14] A. Juels and M. Sudan. A fuzzy vault scheme. *Designs, Codes and Cryptography*, 38(2):237–257, 2006.

[15] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM Conference on Computer and Communications Security*, CCS '99, pages 28–36, New York, NY, USA, 1999. ACM.

[16] N. I. T. Laboratory. Fingerprint Vendor Technology Evaluation. `http://www.nist.gov/itl/iad/ig/fpvte2012.cfm`, 2012. [Online; accessed 23-March-2016].

[17] N. I. T. Laboratory. Ongoing MINEX. `http://www.nist.gov/itl/iad/ig/ominex.cfm`, 2016. [Online; accessed 23-March-2016].

[18] S. Lucey and T. Chen. A GMM parts based face representation for improved verification through relevance adaptation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 855–861, 2004.

[19] D. Maio, D. Maltoni, R. Cappelli, A. Franco, and M. Ferrara. FVC-onGoing: on-line evaluation of fingerprint recognition algorithms. `https://biolab.csr.unibo.it/fvcongoing/UI/Form/Home.aspx`, 2016. [Online; accessed 23-March-2016].

[20] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain. FVC2002: Second Fingerprint Verification Competition. In *16th International Conference on Pattern Recognition, ICPR 2002, Quebec, Canada.*, pages 811–814, 2002.

[21] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain. FVC2002: Fingerprint Verification Competition. `http://bias.csr.unibo.it/fvc2002/`, 2016. [Online; accessed 23-March-2016].

[22] M. Mak and W. Rao. Acoustic vector resampling for GMMSVM-based speaker verification. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan*, pages 1449–1452, 2010.

[23] S. Prabhakar, S. Pankanti, and A. K. Jain. Biometric recognition: Security and privacy concerns. *IEEE Security & Privacy*, 1(2):33–42, 2003.

[24] N. K. Ratha, J. H. Connell, and R. M. Bolle. An analysis of minutiae matching strength. In *Third International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 223–228, London, UK, 2001. Springer-Verlag.

[25] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1 - 3):19 – 41, 2000.

[26] R. Wallace, M. McLaren, C. McCool, and S. Marcel. Intersession variability modelling and joint factor analysis for face authentication. In *International Joint Conference on Biometrics*, pages 1–8, Los Alamitos, CA, USA, 2011.

[27] H. Xu and R. N. J. Veldhuis. Complex spectral minutiae representation for fingerprint recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 1–8, June 2010.

[28] B. Yang, C. Busch, P. Bours, and D. Gafurov. Robust minutiae hash for fingerprint template protection. In *Media Forensics and Security*, volume 7541 of *SPIE Proceedings*, page 75410. SPIE, 2010.

[29] B. Yang, D. Hartung, K. Simoens, and C. Busch. Dynamic random projection for biometric template protection. In *Fourth IEEE International Conference on Biometrics: Theory Applications and Systems, BTAS 2010, Washington, DC, USA*, pages 1–7, 2010.