FUSION DECISION FOR A BIMODAL BIOMETRIC VERIFICATION SYSTEM USING SUPPORT VECTOR MACHINE AND ITS VARIATIONS

A. Teoh, S.A. Samad and A. Hussain

Department of Electrical, Electronic and Systems Engineering,
Universiti Kebangsaan Malaysia

ABSTRACT

This paper presents fusion decision technique comparisons based on support vector machine and its variations for a bimodal biometric verification system that makes use of face images and speech utterances. The system is essentially constructed by a face expert, a speech expert and a fusion decision module. Each individual expert has been optimized to operate in automatic mode and designed for security access application. Fusion decision schemes considered are linear, weighted Support Vector Machine (SVM) and linear SVM with quadratic transformation. The conditions tested include the balanced and unbalanced conditions between the two experts in order to obtain the optimum fusion module from these techniques best suited to the target application.

1. INTRODUCTION

Automatic access of eligible persons to services is becoming common, for example accessing a computer account, an automatic teller machine or even a website. Traditionally, a user can be verified using their ID card and/or password, but these approaches have several drawbacks. The cards can be stolen or misplaced while the passwords can be forgotten. Hence, new verification methods have emerged, where the ID card or password has either been replaced by, or used in addition to, biometrics such as the person's speech, face image or fingerprints. The use of biometrics is attractive since they cannot be lost or forgotten and varies significantly between individuals.

However, a major problem with biometrics is that the physical appearance of a person tends to vary with time. In addition, correct authentification may not be guaranteed due to sensor noise and limitations of feature extractor and matcher. One solution to cope with these limitations is to combine several biometrics in a multi-modal identity verification system[1].

Some work on multi-modal biometric identity verification systems has been reported in literature. Brunelli and Falavigna[2] have proposed a person identification system based on acoustic and visual features, where they use a HyperBF network as the best performing fusion module. Dieckmann *et al.* [3] have proposed a decision level fusion scheme, based on a *2-out-of-3* majority voting, which integrates face and voice, analyzed by three different

experts: face, lip motion, and voice. Duc *et al.*[4] proposed a simple averaging technique and compared it with the Bayesian integration scheme presented by Bigun *et al.*[5]. In this multi-modal system, the authors use a face identification expert, and a text-dependent speech expert. Kittler *et al.*[6] proposed a multi-modal person verification system, using three experts: frontal face, face profile, and voice. The best combination results are obtained for a simple sum rule. Hong and Jain[7] proposed a multi-modal personal identification system which integrates face and fingerprints that complement each other. The fusion algorithm operates at the expert *(soft)* decision level, where it combines the scores from the different experts under statistically independent hypothesis, by simply multiplying them. Ben-Yacoub[8] proposed a multi-modal data fusion approach for person authentication, based on Support Vector Machines (SVM) to combine the results obtained from a face identification expert, and a text-dependent speech expert. Pigeon[9] proposed a multi-modal person authentication approach based on simple fusion algorithms to combine the results coming from the frontal face, face profile, and voice modal. Choudhury *et al.*[10] proposed a multi-modal person recognition using unconstrained audio and video. The combination of the two experts is performed using a Bayes Net.

A bimodal biometric verification system based on facial and vocal modalities is described in this paper. It differs from the systems that are mentioned above in the sense that this system is targeted for applications involving automatic verification using personal computers and their multimedia capturing devices. Thus each module of the system has been fine-tuned to deal with the problems that may occur in this type of application, such as poor quality images obtained from using a low cost PC camera and the problem of using various types of microphones that may cause channel distortion or convolution noise. In addition, the system is designed to keep the rate as low as possible for the case when an imposter is accepted as being a genuine. Each module of the system, i.e. the face and voice, is developed separately and several fusion decision schemes are compared with the aim to obtain the optimum technique for this application. In addition, an 'unbalanced' case has been created to further investigate the robustness among the decision techniques.

2. VERIFICATION MODULES

2.1 Face verification

In personal verification, face recognition refers to static, controlled full frontal portrait recognition. There are two major tasks in face recognition: (i) face detection and (ii) face verification.

In our system as shown in Figure 1, the Eigenface approach[11] is used in the face detection and face recognition modules. The main idea of the Eigenface approach is to find the vectors that best account for the distribution of face images within the entire image space and define the face-space. Face-spaces are eigenvectors of the covariance matrix corresponding to the original face images, and since they are face-like in appearance they are so called eigenfaces as shown in Figure 2.

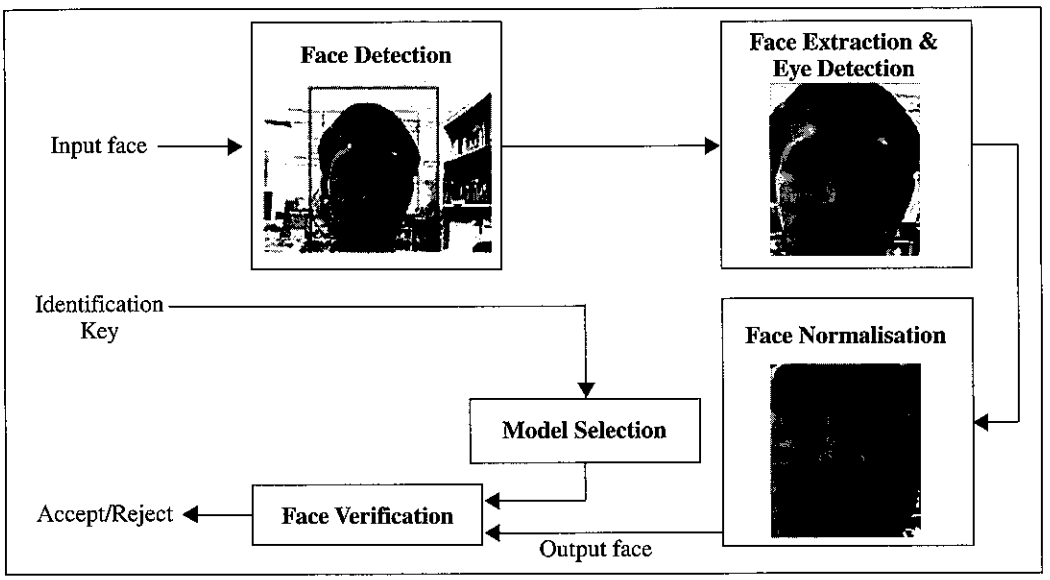

Figure 1: Face Verification System.

Figure 2: Eigenfaces.

Now let the training set of face images be $i_1, i_2, \ldots, i_m$, the average face of the set is defined as:

$$\bar{i} = \frac{1}{M}\sum_{j=1}^{M} i_j \tag{1}$$

where M is the total number of images.

Each face differs from the average by the vector $\phi_n = i_n - \bar{i}$. A covariance matrix is constructed where:

$$\begin{aligned} C &= \sum_{j=1}^{M} \phi_j \phi_j^T \\ &= AA^T \end{aligned} \tag{2}$$

where $A = [\phi_1 \phi_2 \ldots \phi_M]$.

Then, eigenvectors, v_k and eigenvalues, λ_k with symmetric matrix C are calculated. v_k determine the linear combination of M difference images with ϕ to form the eigenfaces:

$$u_l = \sum_{k=1}^{M} v_{lk}\phi_k \qquad l = 1,...,M \tag{3}$$

From these eigenfaces, $K(<M)$ eigenfaces are selected to correspond to the K highest eigenvalues.

Face detection is accomplished by calculating the sum of the square error between a region of the scene and the Eigenface, a measure of Distance From Face Space (DFFS) that indicates a measure of how face-like a region is. If a window, ϕ is swept across the scene, to find the DFFS at each location, the most probable location of the face can be estimated. This will simply be the point where the reconstruction error, ε has the minimum value.

$$\varepsilon = \left\| \phi - \phi_f \right\| \tag{4}$$

where ϕ_f is the projection into face-space.

From the extracted face, eye co-ordinate will be determined with the hybrid rule based approach and contour mapping technique[12]. Based on the information obtained, scale normalization and lighting normalization are applied for a *head in box* format.

The Eigenface-based face recognition method[13] is divided into two stages: (i) the training stage, (ii) the operational stage. At the training stage, a set of normalized face images, $\{i\}$ that best describe the distribution of the training facial images in a lower dimensional subspace (eigenface) is computed by the operation:

$$\varpi_k = u_k\left(i_n - \bar{i}\right) \tag{5}$$

where $n = 1, \ldots, M$ and $k=1, \ldots, K$.

Next, the training facial images are projected onto the eigenspace, Ω_i, to generate the representations of the facial images in eigenface.

$$\Omega_i = \left[\varpi_{n1}, \varpi_{n2}, \ldots, \varpi_{nK}\right] \tag{6}$$

where $i=1,2, \ldots, M$.

At the operational stage, an incoming facial image is projected onto the same eigenspace and the similarity measure which is the Mahalanobis distance between the input facial image and the template is, thus, computed in the eigenspace.

Let φ_1^0 denote the representation of the input face image with claimed identity C and φ_1^C denote the representation of the Cth template. The similarity function between φ_1^0 and φ_1^C is defined as follows:

$$F_1(\varphi_1^0, \varphi_1^C) = \left\| \varphi_1^0 - \varphi_1^C \right\|_m \tag{7}$$

where $\|\bullet\|_m$ denotes the Mahalanobis distance.

2.2 Speaker verification

Anatomical variations that naturally occur amongst different people and the differences in their learned speaking habits manifest themselves as differences in the acoustic properties of the speech signal. By analyzing and identifying these differences, it is possible to discriminate among speakers[14]. Our front end of the speech module aims to extract the user dependent information.

The system includes three important stages: end-point detection, feature extraction and pattern comparison. The end-point detection stage aims to remove silent parts from the raw audio signal, as this part does not convey speaker-dependent information.

Noise reduction techniques are used to reduce the noise from the speech signal. Simple spectral subtraction[15] is first used to remove additive noise prior to end-point detection. Then, in order to cope with the channel distortion or convolution noise that is introduced by a microphone, the zero'th order cepstral coefficients are discarded and the remaining coefficients are appended with delta feature coefficients. In addition, the cepstral components are weighted adaptively to emphasize the narrow-band components and suppress the broadband components[16]. The cleaned audio signal is converted to a 12th order linear prediction cepstral coefficients (LPCC), using the autocorrelation method that leads to a 24 dimensional vector for every utterance. The significant improvement of verification rate by using this combination can be found in paper[17]. Figure 3 shows the process used in front-end module.

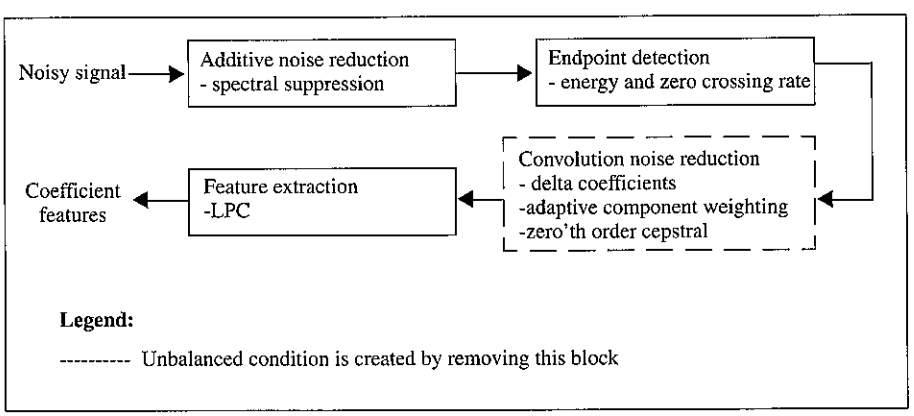

Figure 3: The front-end of the speaker verification module.

As with the face recognition module, the speaker verification module also consists of two stages: (i) the training stage and (ii) the operational stage. At training phase, two sample utterances with the same words from the same speaker are collected and trained using the modified *k*-Mean algorithm[18]. The main advantages of this algorithm are the statistical consistency of the generated templates and their ability to cope with a wide range of individual speech variations in a speaker-independent environment.

At the operational stage, we opted for a well-known pattern-matching algorithm — Dynamic Time Warping (DTW)[19] to compute the distance between the trained template and the input sample.

Let φ_2^0 represent the input speech sample with the claimed identity C and φ_2^C the *Cth* template. The similarity function between φ_2^0 and φ_2^C is defined as follows:

$$F_1\left(\varphi_2^0, \varphi_2^C\right) = \left\|\varphi_2^0 - \varphi_2^C\right\| \tag{8}$$

where $\|\bullet\|$ denotes the distance score result from DTW.

3. FUSION DECISION MODULE

3.1 Linear support vector machine

Support Vector Machines (SVM) is a type of machine-learning technique that learns the decision surface to separate the two classes through a process of discrimination. It has good generalization characteristics. SVMs have been proven to be successful classifiers on several classical pattern recognition problems[20].

In conventional pattern classification problem, empirical risk minimization (ERM) is the most commonly used optimization procedure in machine learning. In this regime, the goal is to arrive at a parameter setting that gives the smallest value called risk, R_{emp}. The risk computation can take other forms such as the sum-squared error. Neural network training, back-propagation in particular, is a direct consequence of a similar optimization process. There are no probability computations involved in the definition of risk.

Another form of risk commonly used is the expected risk or estimated risk, *R*. Vapnik[21] proved that bounds exist for this expected risk such that,

$$R \leq R_{emp} + f(h) \tag{9}$$

where *h* is Vapnik Chervonenkis (VC) dimension. Finding a learning machine with the minimum upper bound on the estimated risk leads to a method of choosing an optimal machine for a given task. This is the essential idea of Structural Risk Minimization (SRM). The SVM is based on the principle of SRM.

3.2 Linearly separable data in linear SVM

Figure 4 shows a typical 2-class classification example where the classes are perfectly separable using a linear decision region. Let w be normal to the decision region and let the N training examples be represented as the pairs $\{x_i, y_i\}$, $i = 1,2,...,N$ where $-1 \leq y_i \leq 1$. The points that lie on the hyper plane to separate the data satisfy,

$$w \cdot x + \gamma = 0 \tag{10}$$

where γ is the distance of the hyper plane from the origin. Let the margin of the SVM be defined as the distance between closest positive and negative example from the hyper plane. The SVM looks for the separating hyper plane, which gives the maximum margin. Once the hyper plane is obtained, all the training examples satisfy the following inequalities.

$$w \cdot x_i + \gamma \geq +1 \qquad \text{for } y_i = +1 \tag{11}$$

$$w_i \cdot x_i + \gamma \leq -1 \qquad \text{for } y_i = -1 \tag{12}$$

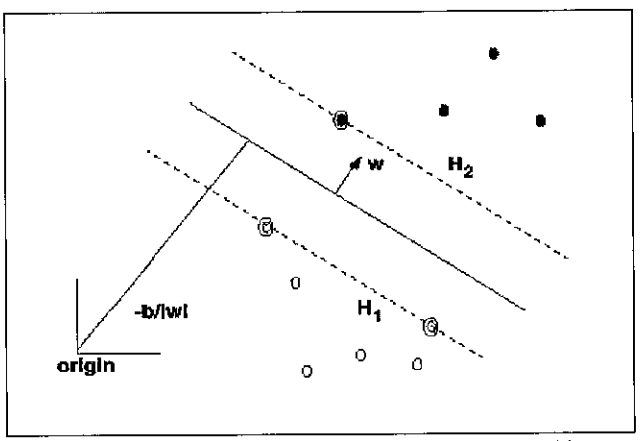

Figure 4: Linear separating hyper planes for the separable case.

Looking at the above equations with respect to Figure 4, the distance between H_1 and H_2 is $2/\|w\|$. Note that for a completely separable data set, no points fall between H_1 and H_2. Thus for maximizing the margin, we need to minimize $\|w\|^2$. In Figure 4, they are indicated by concentric circles. This leads to a Quadratic Programming problem, which can make use of the theory of Lagrange multipliers.

3.3 Non-linearly separable data in linear SVM

Most of the classification problems in real world involve non-separable data. The optimal-margin classifier can be extended to this non-separable case by using a set of slack variables. In this situation, the inequality constraints become:

$$w \cdot x_i + \gamma \geq +1 - \xi_i \qquad \text{for } y_i = +1 \tag{13}$$

$$w \cdot x_i + \gamma \leq -1 + \xi i \qquad \text{for } y_i = -1 \tag{14}$$

$$\xi_i \geq 0 \qquad \forall_i \tag{15}$$

A close look the above inequalities (15) shows that for an error to occur, the corresponding ξ_i needs to be greater than 1. This implies that the upper bound on the number of errors on the training data is $\Sigma_i \xi_i$. In addition, the optimization process in the new data setting needs to minimize this quantity. The new term that is added to the objective is as follows:

$$C(\Sigma_i \xi_i)^2 \tag{16}$$

where C is used to control the penalty for a training error.

3.4 Linear support vector machines with quadratic transformation

In general, the linear separation of points in the feature space does not suffice and the non-linear separation hypersurface should be used instead. In some cases, it is an advantage to re-map the original feature space non-linearly to a new space where the separation by a hyperplane is again possible. The new feature space has often higher dimensions.

$$q = f(l) \tag{17}$$

where $f(.)$ is a transformation function, l is feature matrix with size N x K in the original N-dimensional feature space with K points. q is a matrix of images of the point set l with size M x K.

In this case, the quadratic transformation has been applied. This leads to N = 2 and thus M = 2N+N(N-1)/2 = 5. Hence a new classifier can be constructed by using transformation[17]. A 5 dimensional linear separator w and a bias γ are then constructed for the set of transformed vectors. Classification of an unknown vector, x is done by first transforming the vector to the feature space and then computing

$$\text{sgn}(wx + \gamma) \tag{18}$$

The above formulation is based on the fact that among all hyperplanes separating the data, there exists a unique one that maximizes the margin of separation between the classes. Figure 5 shows the example of linear SVM and transformed SVM classifier.

4. EXPERIMENTS AND DISCUSSION

4.1 Distance score normalization

The similarity measure values from equations (7) and (8) have different ranges and hence cannot be fused directly. They have to be mapped into a common score interval between [0 1].

A high score indicates the person is genuine, while a low opinion suggests the person is an imposter. The opinions from the modality experts are used by a fusion stage also referred to as a decision stage. It considers the opinions and makes the final decision to either accept or reject the claim. The bimodal biometric system here is designed as shown in Figure 6.

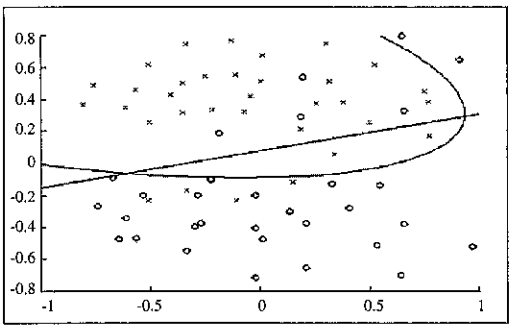

Figure 5: Linear SVM and its quadratic transformation classifier for two classes classification.

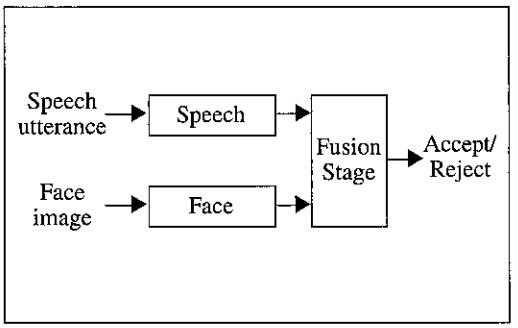

Figure 6: The building block of the bimodal face and speech verification system.

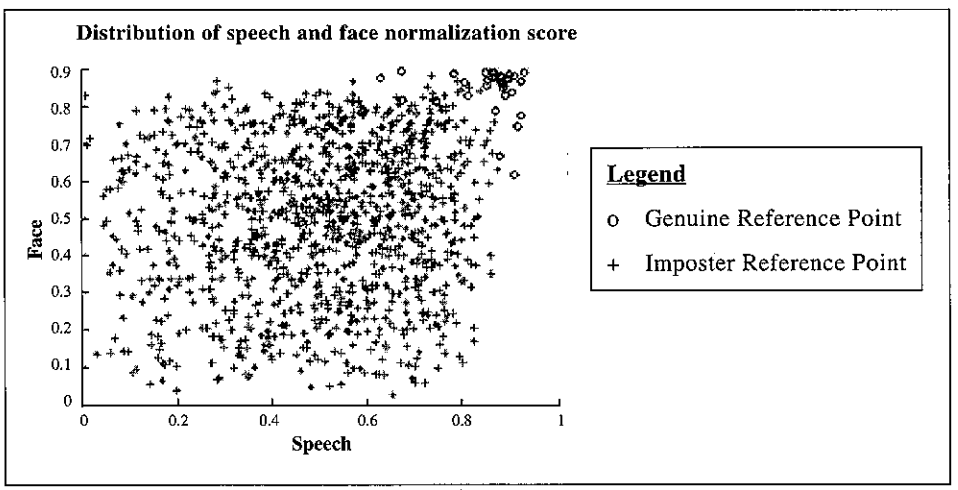

Figure 7: The distribution plot for the genuine and imposter reference points.

From the distance scores, x that is produced by the speech and face databases, the mean, μ, sand the variance, σ^2, of the distance values of the speech and face expert, respectively are found by performing validation experiments on the database. The distance score is then normalized by mapping it to the range [-1,1] using:

$$y = \frac{x - \mu}{\sigma} \tag{19}$$

The [-1,1] interval corresponds to the approximately linear changing portion of the sigmoid function

$$f(y) = \frac{1}{1 + \exp(-y)} \tag{20}$$

used to map the values to the [0,1] interval. Figure 7 shows the distribution plot for the genuine and imposter reference points obtained for the system of mapping.

4.2 Performance criteria

The basic error measure of a verification system are false acceptance rate (FAR) and false rejection rate (FRR) as defined in equations (21) and (22).

$$\text{FAR} = \text{Number of accepted imposter claims/total number of imposter accesses} \quad (21)$$

$$\text{FRR} = \text{Number of rejected genuine claims/total number of genuine accesses} \quad (22)$$

A unique measure can be obtained by combining these two errors into the total error rate (TER) or total success rate (TSR) where:

$$\text{TER} = (\text{FAR} + \text{FRR})/(\text{Total number of accesses x } 100) \quad (23)$$

$$\text{TSR} = 1 - \text{TER} \quad (24)$$

In this targeted application, the Minimum Total Misclassification Error (MTSE) criterion is used, which means the system always tries to minimize ε as shown in equation (25):

$$\varepsilon = \min(\text{FA} + \text{FR}) \quad (25)$$

In order to apply this criterion, we set FAR <1% while keeping the FRR to a minimum possible value.

4.3 Experimental setup

All experiments are performed using a face database obtained from Olivetti Research Lab[22] and speech database contributed by Otago speech corpus[23]. Three sessions of the face database and speech database are used separately. The first enrolment session is used for training. This means that each access is used to model the respective genuine, yielding 34 different genuine models.

In the second enrolment session, the accesses from each person are used to generate the validation data in two different manners. The first is to derive a single genuine access by matching the shot or utterance template of a specific person with his own reference model, and the other is to generate 34 imposter accesses by matching it to the 33 models of the other persons of the database. This simple strategy thus leads to 34 genuine and 1122 imposter accesses, which are used to validate the performance of the individual verification system and to calculate the thresholds for the equal error rate (EER) criterion.

The third enrolment session is used to test these verification systems, using the thresholds calculated with the validation data set.

4.4 Experimental results

4.4.1 Balanced (ordinary)

Two cases are considered to represent a balanced and an unbalanced system. For the balanced system, which is the ordinary results obtained from the system, the performances of the speech and face expert are as shown in Table 1.

Table 1: Individual performance of the face and speech expert.

Expert	FAR	FRR	TSR
Speech	8.38%	0%	91.87%
Face	8.02%	8.82%	91.96%

Table 2: Results for ordinary linear SVM and linear SVM fusion with quadratic transformation technique.

SVM	FAR	FRR	TSR
Linear	0%	5.88%	99.83%
Quadratic	0%	2.94%	99.91%

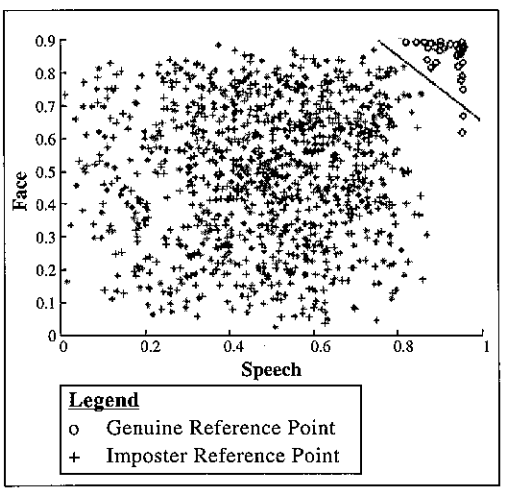

Figure 8: Distribution test points for speech population and face population for the ordinary linear SVM classifier.

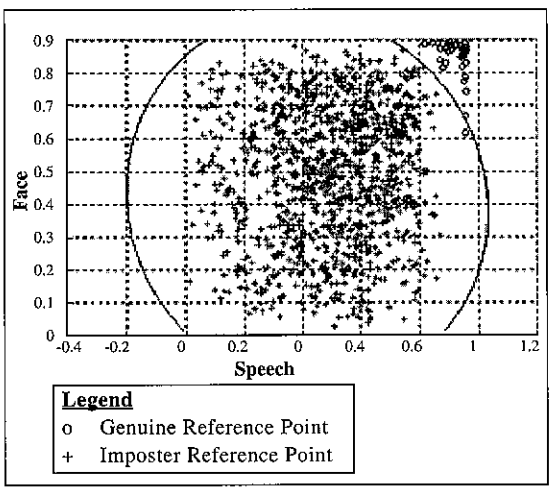

Figure 9: Distribution test points for speech population and face population for the linear SVM classifier with quadratic classifier.

From the values of TSR in Table 1, we can observe that the experts are working equally well individually.

Table 2 shows the results obtained with the linear SVM and linear SVM with quadratic transformation by combining the two experts. The comparison results can be visualized as shown in Figures 8 and 9.

Among these two techniques considered, linear SVM with quadratic transformation performed the best as it introduces zero FAR and low FRR, while linear SVM is also adequate for this application.

4.4.2 Unbalanced case

To approximate real conditions, an unbalanced version of the biometric system is created with the two experts performing at unequal TSE, unlike that of the balanced system. Since the speech expert requires an extensive pre-processing stage compared to the face expert, it is used to create an unbalanced case.

The unbalanced version is created by eliminating the pre-processing steps for convolution noise reduction that have been applied in speech expert as indicated in Figure 3. In addition, the training procedure is omitted in the training phase. This causes the performance to be reduced in the speech expert as shown in Table 3. The distribution plot for the validation set is as shown in Figure 10.

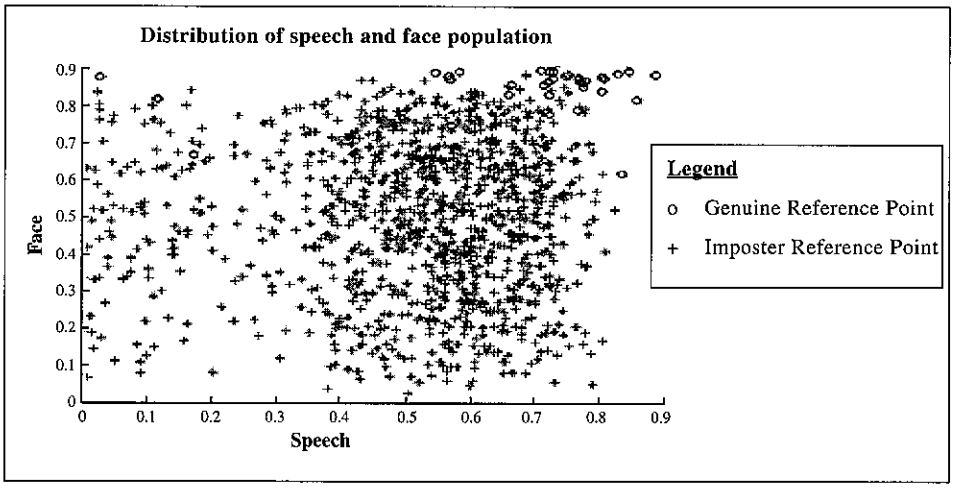

Figure 10: The distribution plot for validation set in the unbalanced system.

Table 3: Results from the unbalanced face and speech expert.

Expert	FAR	FRR	TSR
Speech	29.59%	5.88%	71.11%
Face	8.02%	8.82%	91.96%

Table 4: Results for ordinary and weighted Linear SVM fusion technique in the unbalanced case.

SVM	FAR	FRR	TSR
Ordinary	0%	32.35%	99.05%
Weighted	0.36%	11.77%	99.31%

The fusion module is again tested using the methods used for the balanced case. When using linear SVM classifier, the performance drops significantly especially for FRR as shown in Table 4. This is undesirable in our application.

In linear SVM, the only parameter that could be adjusted is C as indicated in equation (16). It is responsible for achieving the MTSE criterion in equation (25). Gutschovan *et al.*[24] suggested adding a priori knowledge in TSR as weights to maximize the margin. It is indeed more useful to have a bigger margin for a reliable expert with high TSR than for a less reliable expert with lower TSR. Expression $\min(1/2\|w\|^2)$ shows that maximizing the margin is equivalent to minimizing the L_2 - norm of w. To attribute different weights to each component, we can weigh each component as follows:

$$w = \sqrt{\sum_{i=1}^{n} g_i w_i} \tag{26}$$

where $g_i = 1/(TSR_i)$ (27)

Table 4 is the results obtained before and after applying weighted w.

Clearly, it can be seen that the weighted LSVM improves the results significantly; the FRR is reduced significantly. The comparison results can be visualized as shown in Figures 11 and 12.

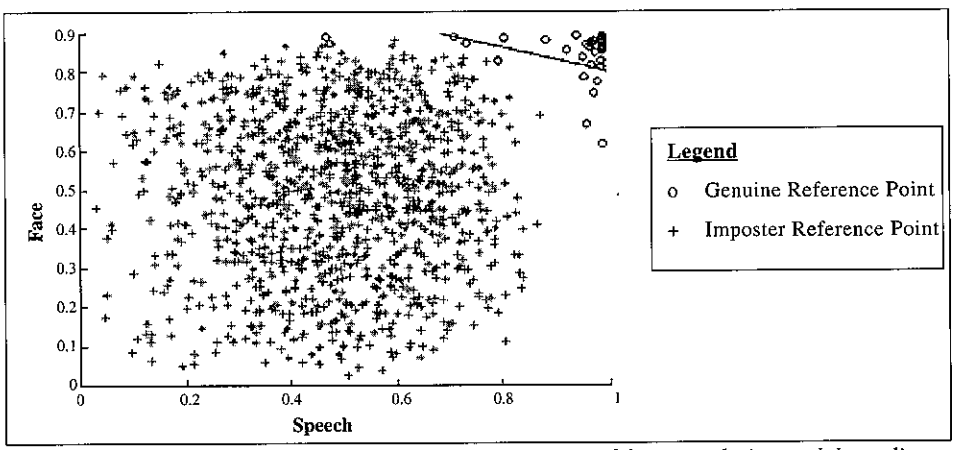

Figure 11: Distribution test points for speech population and face population and the ordinary linear SVM classifier.

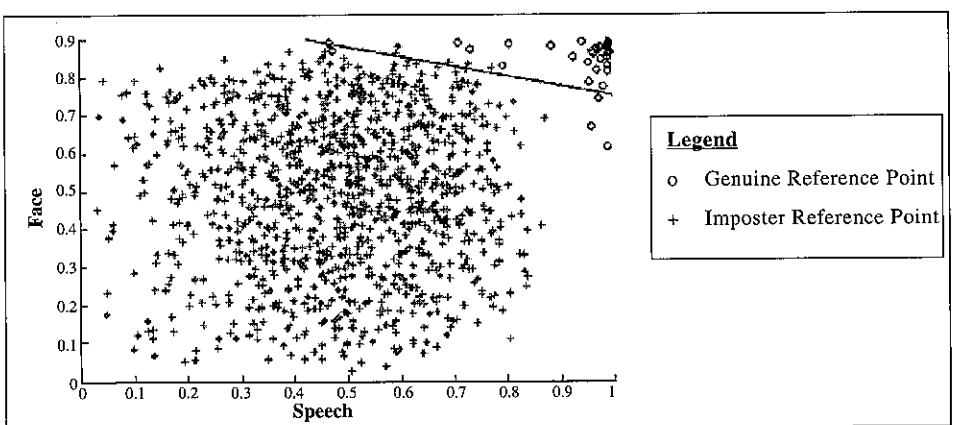

Figure 12: Distribution test points for speech population and face population and the weighting linear SVM classifier.

Table 5: Result for Linear SVM with quadratic transformation in the unbalanced case.

SVM	FAR	FRR	TSR
Quadratic	0.1782%	5.88%	99.65%

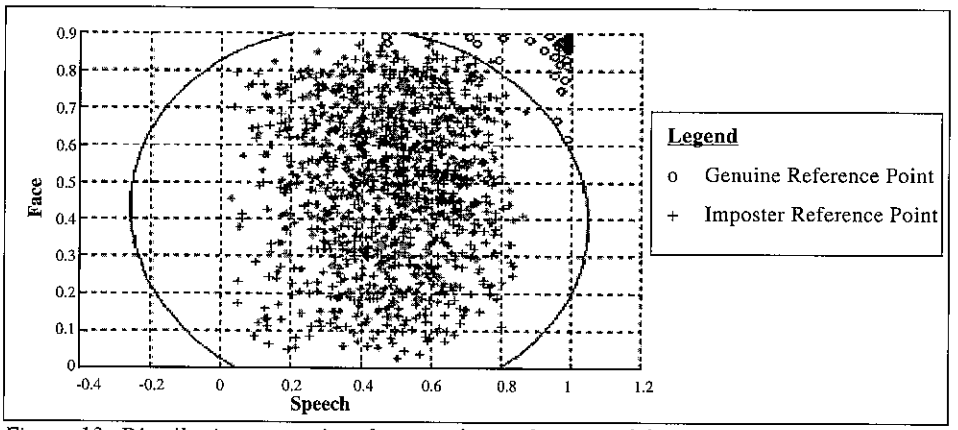

Figure 13: Distribution test points for speech population and face population for the linear SVM with quadratic transformation classifier.

By using the linear SVM with quadratic transformation, the results as shown in Table 5 is obtained.

The result can be visualized as shown in Figure 13.

As with the case of the balanced version, the linear SVM with quadratic transformation still outperformed the other techniques. This is because the transformation to higher dimension enables the hyper plane to separate the data more efficiently.

5. CONCLUSION

The paper has shown fusion decision technique comparisons based on SVM and its variations for a biometric verification system. The system consists of speech and face experts developed separately and targeted for applications involving automatic verification using personal computers and their multimedia capturing devices. In addition, the system is designed to keep the rate as low as possible for the case when an imposter is accepted as being a genuine. The fusion decision schemes considered are both the ordinary and weighted linear Support Vector Machine and also its quadratic transformation.

From the experiments, it is found that the best result is obtained using the linear vector machine with quadratic transformation as it introduces zero FAR and low FRR consistently, either in the balanced or unbalanced versions of the system, compared to both the ordinary and weighted linear Support Vector Machine classifier.

6. REFERENCES

1. Jain, A., Bolle, R. and Pankanti, S., "Biometrics, Personal identification in networked society", 2nd Printing, Kluwer Academic Publishers, 1999.
2. Brunelli, R. and Falavigna, D., "Personal identification using multiple cues", *IEEE Trans. on Pattern Analysis and Machine Intelligence,* 17(10), 955-966, 1995.
3. Dieckmann, U., Plankensteiner, P. and Sesam, T. W., "A biometric person identification system using sensor fusion", *Pattern recognition letters,* 18(9), 827-833, September 1997.
4. Duc, B., Maýtre, G., Fischer, S and Bigun, J., "Person authentication by fusing face and speech information", *In Proceedings of the First International Conference on Audio- and Video-based Biometric Person Authentication,* Lecture Notes in Computer Science. Springer Verlag, 1997.
5. Bigun, E., Bigun, J., Duc, B. and Fisher, S., "Expert conciliation for multi-modal person authentication systems by Bayesian statistics", *In Proceedings of the first international conference on Audio- and Video-based Biometric Person Authentication,* pages 327-334, Crans-Montana, Switzerland, March 1997.
6. Kittler, J., Hatef, M., Duin, R. P. W. and Matas, J., "On combining classifiers", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20(3), 226-239, March 1998.
7. Hong, L. and Jain, A., "Integrating Faces and Fingerprints for Personal Identification", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20(12), 1295-1307, December 1998.
8. Yacoub, S. B., "Multi-Modal Data Fusion for Person Authentication using SVM", IDIAP-RR 7, IDIAP, 1998.
9. Pigeon, S., "Authentification multimodale d'identit´ e", PhD thesis, Universit´e Catholique de Louvain, February 1999.
10. Choudhury, T., Clarkson, B., Jebara, T. and Pentland, A., "Multimodal Person Recognition using Unconstrained Audio and Video", *In Second International Conference on Audio- and Video-based Biometric Person Authentication,* pages 176-181, Washington D. C., USA, March 1999.
11. Moghaddam, B. and Pentland, A., "Probabilistic Visual Learning for Object Detection", M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 326, *The 5th International Conference on Computer Vision, Cambridge,* M.A., June 1995.
12. Samad, S. A., Hussein, A. and Teoh, A., "Eye Detection Using Hybrid Rule Based Method and Contour Mapping", *In Sixth International Symposium on Signal Processing and Its Applications,* pages 631-634, Kuala Lumpur, Malaysia, August 2001.
13. Turk, M. and Pentland, A., "Eigenfaces for Recognition", *Journal of Cognitive Neuro science,* 3(1), 71-86, 1991.
14. Campbell, J., "Speaker Recognition", A tutorial, Proceeding of the IEEE, 85(9), 1437 - 1462, September 1997.
15. Martin, R., "Spectral Subtraction Based on Minimum Statistics", *Proc. Seventh European Signal Processing Conference,* pp. 1182-1185, 1994.

16. Zilovic, M. S., Ramachandran, R. P. and Mammone, R. J., "A fast algorithm for finding the adaptive component weighting cepstrum for speaker recognition", *IEEE Transactions on Speech & Audio Processing,* vol. 5, pp. 84-86, Jan. 1997.
17. Samad, S. A., Hussein, A. and Teoh, A., "Increasing Robustness In A Speaker Verification System With Template Training And Noise Reduction Techniques", to appear in *Proceedings of the International Conference on Information Technology and Multimedia,* 2001.
18. Wilpon J. G. and Rabiner, L. R., "A modified K-Means clustering algorithm for use in isolated word recognition", *IEEE Trans, Acoustics Speech, Signal Proc.,* ASSP-33(3), 587-597, June, 1985.
19. Sakoe H. and Chiba S., "A dynamic programming approach to continuous speech recognition", *in Proc. 7th Int. Congress Acoustics,* Paper 20, C13, 1971.
20. Burges, C. J. C., "A Tutorial on Support Vector Machines for Pattern Recognition", Bell Laboratories, Lucent Technologies. Data Mining and Knowledge Discovery, 2(2), 121-167, 1998.
21. Vapnik, V. N., "The Nature of Statistical Learning Theory", Springer, 1995.
22. Database of faces, http://www.cam-orl.co.uk/facedatabase.html
23. Otago Speech Corpus, http://kel.otago.ac.nz/hyspeech/corpusinfo.html
24. Gutschoven, B. and Verlinde, P., "Decision Fusion using Support Vector Machines (SVM)", *Proceedings of the 3rd International Conference on Information Fusion,* Paris, France, July 2000.