Feature Distribution Modelling Techniques for 3D Face Verification

Chris McCool, Jordi Sanchez-Riera and Sébastien Marcel

Idiap Research Institute, PO Box 592, CH-1920 Martigny, Switzerland

Abstract

This paper shows that Hidden Markov Models (HMMs) can be effectively applied to 3D face data. The examined HMM techniques are shown to be superior to a previously examined Gaussian Mixture Model (GMM) technique. Experiments conducted on the Face Recognition Grand Challenge database show that the Equal Error Rate can be reduced from 0.88% for the GMM technique to 0.36% for the best HMM approach.

Key words: Feature Distribution Modelling, Hidden Markov Models, Gaussian Mixture Models, 3D Face Verification

1. Introduction

Early research into face recognition was inspired by the fact that humans can recognise a face from a photograph (gray-scale) image (Bledsoe, 1966) and it took over 20 years, in 1989, before research using 3D face data to identify humans was first published by Cartoux et al. (1989). However, there are several advantages to performing verification on a 3D face image.

The 3D face data encodes the structure of the face and so is inherently robust to pose and illumination variations. Pose variations can be fully recovered as the 3D coordinates are available, this means in-plane and out-of-plane rotations can be normalised whereas for 2D faces only in-plane rotations can be normalised. Also, illumination variations are greatly reduced because the scanning procedure usually involves the projection of an external energy source in order to measure the depth of the face, this is the case for laser scanners and structured light scanners.

A recent advance for 3D face verification has been to show the applicability of the Gaussian Mixture Model (GMM) parts-based approach (McCool et al., 2008). This state-of-the-art 2D face verification technique (Cardinaux et al., 2004; G. Heusch and Marcel, 2006) was shown to be applicable and effective for performing 3D face verification achieving an Equal Error Rate (EER) half that of a standard Principal Component Analysis (PCA) based approach, the EER was reduced from 1.99% to 0.88% for cross-validation experiments performed on the Face Recognition Grand Challenge (FRGC) database.

The GMM parts-based approach is a feature distribution modelling approach that works by dividing the face into parts and considering each part to be an independent observation of the face. The assumption of independence between each observation is a strong assumption, particularly for 3D faces as there is obviously a structural dependence. Since the face obviously has a spatial relationship it is logical to explore feature distribution modelling techniques that incorporate this information, such as Hidden Markov Models (HMMs).

Applying HMMs to 3D face verification was first attempted in 1997 by Achermann et al. (1997). They found that applying the HMM technique of Samaria and Fallside (1993) to depth data (instead of pixel intensities) produced a system that had similar performance to a PCA based system. This initial attempt at applying HMMs to 3D face data showed some promise but was never extended, by contrast the research into applying HMMs to 2D faces has made significant progress.

Cardinaux et al. (2004) described two Hidden Markov Model (HMM) methods for performing 2D face verification. These two feature distribution modelling approaches, referred to as HMM 1D and HMM 2D, were shown to be more accurate than a GMM parts-based approach and were capable of describing the spatial relationship of the face. However, these two state-of-the-art HMM techniques have never been applied to 3D faces.

The aim of this article is to examine the effectiveness of state-of-the-art feature distribution modelling techniques for 3D face verification. The two techniques under consideration are the HMM 1D and HMM 2D approaches which have both been successfully applied to 2D face recognition (Cardinaux et al., 2004) but not to 3D face verification. We also examine different methods of encoding the feature vector, such as including delta coefficients and performing feature vector normalisation.

In the next section we will provide an overview of feature distribution modelling and the feature vectors used for these experiments. The experimental procedure will then be outlined and we will present and discuss the results. Conclusions will then be drawn and future work will be outlined.

2. Feature Distribution Modelling and Classification

Feature distribution modelling techniques for face verification aim to describe the face (signal) in a generative manner. This means that the probability density function (pdf) of the face (signal) is modelled. For face verification two main methods for feature distribution modelling have been explored: GMMs (Sanderson and Paliwal, 2002a) and HMMs (Samaria and Fallside, 1993; Cardinaux et al., 2004).

A major obstacle in applying feature distribution modelling techniques to face verification is the lack of observations. In particular, there are very few client observations to be able to derive an effective model. Two solutions have been used, normally in conjunction, to alleviate this problem: to perform background model adaptation and to apply a parts-based approach or topology.

In the following section we provide a detailed overview of two distribution modelling techniques (GMMs and HMMs). We then provide an overview of background model adaptation followed by a description of the parts-based approaches (topologies) that have been used for face verification.

2.1. Distribution Modelling and Parameter Estimation

GMMs and HMMs are both statistical models (Bishop, 2006) that aim to compute $P(X \mid \Omega)$ which is the probability of observing some data X given model parameters Ω. The model parameters are estimated using the expectation maximisation (EM) (Dempster et al., 1977) and in general the choice between a GMM or a HMM is driven by the assumptions made about the data. In the case of the GMM parts-based approach (Sanderson and Paliwal, 2002a) the observations are assumed to be independent while for the HMM approaches (Samaria and Fallside, 1993; Cardinaux et al., 2004) there is assumed to be a dependence between consecutive observations. A more detailed description of both the GMM and HMM is given below.

2.1.1. Gaussian Mixture Model

A GMM assumes an independence between the observations derived from X. For instance, if we consider X as a finite set of observations x_t with $t = 1..T$ and x_t is a k-dimensional feature vector ($x_t \in \mathbb{R}^k$). Then $P(X \mid \Omega)$ can be computed as $P(X \mid \Omega) = \prod_{t=1}^{T} p(x_t \mid \Omega)$ where $p(x_t \mid \Omega)$ is modelled by a mixture of M Gaussians (GMMs) as follows:

$$p(x_t \mid \Omega) = \sum_{m=1}^{M} \lambda_m p(x_t \mid m) \tag{1}$$

and where λ_m is the weight of Gaussian m ($\sum_{m=1}^{M} \lambda_m = 1$) and $p(x_t \mid m) = \mathcal{N}(x_t; \mu_m, \sigma_m)$ is given by the Normal distribution:

$$\mathcal{N}(x_t; \mu_m, \sigma_m) = \frac{1}{\sqrt{(2\pi)^k \mid \sigma_m \mid}} e^{-\frac{1}{2} {}^T (x_t - \mu_m) \sigma_m^{-1} (x_t - \mu_m)} \tag{2}$$

parametrised by the mean vector $\mu_m \in \mathbb{R}^k$ and the diagonal[1] covariance matrix $\sigma_m \in \mathbb{R}^k$ for each Gaussian m. Hence, Ω is defined by the set of all the parameters $\Omega = \{\lambda_1, ..., \lambda_m, ..., \lambda_M, \mu_1, ..., \mu_m, ..., \mu_M, \sigma_1, ..., \sigma_m, ..., \sigma_M\}$.

Parameter estimation is achieved using the EM algorithm and consists of iteratively:

1. estimating (E-step) the posterior probability

$$p(m|x_t) = \frac{\lambda_m p(x_t|m)}{\sum_{m=1}^{M} \lambda_m p(x_t|m)} \tag{3}$$

---

[1]It is common to use a diagonal covariance matrix instead of a full covariance matrix.

2. maximizing (M-step) the likelihood by applying the following update rules:

$$\lambda_m = \frac{\sum_{t=1}^{T} p(m|x_t)}{\sum_{m=1}^{M} \sum_{t=1}^{T} p(m|x_t)}$$

$$\mu_m = \frac{\sum_{t=1}^{T} p(m|x_t)x_t}{\sum_{t=1}^{T} p(m|x_t)}$$

$$\sigma_m = \frac{\sum_{t=1}^{T} p(m|x_t)x_t^2}{\sum_{t=1}^{T} p(m|x_t)} - \mu_m^2 \tag{4}$$

until the E-step (step 1) converges.

2.1.2. Hidden Markov Model

A HMM assumes a dependence between consecutive observations within X and so is capable of describing the spatial relationship between blocks (observations). The underlying assumption is that the system that generated the observations is a Markov process with unobserved state. Therefore, X is considered as a sequence of observations $x_1^T = x_1...x_T$ and $P(X \mid \Omega) = P(x_1^T)$ is computed by introducing a discrete random variable $q = 1..N$ to represent the unobserved state (q is also called the hidden state variable) as follows $P(x_1^T) = \sum_{j=1}^{N} p(x_1^T, q_T = j)$. This probability is obtained using the recursive formulation:

$$\alpha_{j,t} = p(x_1^t, q_t = j)$$

$$= p(x_t \mid q_t = j) \sum_{i=1}^{N} p(q_t = j \mid q_{t-1} = i) \, \alpha_{i,t-1} \tag{5}$$

The term $p(x_t \mid q_t)$ is called the emission probability associated to a state q and is modelled by a mixture of Gaussians (Eq.1). The term $p(q_t \mid q_{t-1})$ is called the transition probability and is modelled by a $N \times N$-matrix of transition probabilities between states q_{t-1} and q_t.

Hence, Ω is defined by the set of all transition probabilities and of all the parameters Ω_q: the weights $\lambda_{q,m}$, the mean vectors $\mu_{q,m} \in \mathbb{R}^k$ and the diagonal covariance matrices $\sigma_{q,m} \in \mathbb{R}^k$ for each Gaussian component m from each state q.

Parameter estimation is also achieved using the EM algorithm and consists of iteratively:

1. estimating (E-step) the posterior probability

$$p(q_t = j, q_{t-1} = i|x_t) = \frac{\beta_{j,t} \, p(x_t \mid q_t = j) \, p(q_t = j \mid q_{t-1} = i) \, \alpha_{i,t-1}}{\sum_{j=1}^{N} \alpha_{j,T}} \tag{6}$$

using the following recursive formulation for computing $\beta_{j,t}$:

$$\beta_{j,t} = p(x_{t+1}^T \mid q_t = j)$$

$$= \sum_{i=1}^{N} p(x_{t+1} \mid q_{t+1} = i) \, \beta_{i,t+1} \, p(q_{t+1} = i \mid q_t = j) \tag{7}$$

2. maximizing (M-step) the likelihood by applying update rules for the transition probabilities:

$$p(q_t = j \mid q_{t-1} = i) = \frac{\sum_{t=1}^{T} p(q_t = j, q_{t-1} = i | x_t)}{\sum_{k=1}^{N} \sum_{t=1}^{T} p(q_t = k, q_{t-1} = i | x_t)} \qquad (8)$$

and for each state q by applying the update rules of the GMM (Eq.4) with the posterior probability given by Eq.6.

These two steps are repeated until the E-step (step 1) converges.

2.2. Background Model Adaptation

Background model adaptation has been used extensively in speaker recognition (Reynolds et al., 2000) and in 2D face recognition (Cardinaux et al., 2004). The basis of the technique is to derive a world or background model Ω_{world} with much more data than would be available to train a single client model. Client models are then derived by adapting the background model (the starting point) to better match the client observations.

There are two main methods for performing background model adaptation: mean only adaptation (Reynolds et al., 2000) and full adaptation (Lee and Gauvain, 1996). Full adaptation adapts (or changes) all of the parameters of the background model to better match the client observations. If we consider the case of a multi-component GMM then this means that the weight λ_m, mean vector μ_m and co-variance matrix Σ_m of each component m ($m = [1...M]$) could be updated. By contrast, mean only adaptation adapts (or changes) only the means of the background model, for the case of a GMM this means that only μ_m (the mean vector) for each component m could be updated. For this work mean only MAP adaptation is used as it requires fewer samples to derive an effective model and it has been shown to be effective for GMMs and HMMs for 2D faces (Cardinaux et al., 2004) and for GMMs for 3D faces (McCool et al., 2008). The mean of each component can then be updated using the following equation,

$$\hat{\mu}_m = \alpha \mu_m + (1 - \alpha) \frac{\sum_{t=1}^{T} p(m|x_t) x_t}{\sum_{t=1}^{T} p(m|x_t)}. \qquad (9)$$

To score using these two models, the background model and the client model, the log-likelihood ratio (LLR) is used. Therefore, given an observation x it is scored against both the client model (Ω_{client}) and world or background model (Ω_{world}) model to produce a LLR score,

$$h(x) = \ln(p(x \mid \Omega_{client})) - \ln(p(x \mid \Omega_{world})). \qquad (10)$$

This score is used to assign the observation to the world class of faces (not the client) or the client class of faces (it is the client) and consequently a threshold τ has to be applied to the score $h(x)$ to declare (verify) that x matches to the ith client model Ω_{client}^{i} when $h(x) \geq \tau$.

2.3. Parts-Based Approaches and Topologies

Feature distribution modelling using a parts-based approach has been applied to 2D face verification by several researchers. The first feature distribution modelling technique for 2D faces was applied in 1993 by Samaria and Fallside (1993) and utilised HMMs, it was extended in 1994 by Samaria and Young (1994). This technique obtained features from blocks of the 2D faces using a sliding window approach (the pixel intensities were used as the feature vector). The approach of Samaria and Young predefined the HMM to have five hidden states that were forced to encode the following facial regions, the: forehead (q=1), eyes (q=2), nose (q=3), mouth (q=4) and chin (q=5). In 2002 a more general framework for feature distribution modelling of 2D faces was introduced by Sanderson and Paliwal (2002a).

The general parts-based framework of Sanderson and Paliwal (2002a) divides the 2D face into parts (or blocks), an example of this procedure is given in Figure 1 (a). Several researchers have used this topology to consider each part (block) as a separate observation which can then be modelled using GMMs (Cardinaux et al., 2004; G. Heusch and Marcel, 2006). A similar framework was used by Cardinaux et al. (2004) to define two HMM topologies which are more general than that of Samaria et al., these two topologies are more general as they do not predefine the regions for the hidden states. The two topologies are the HMM 1D topology and the HMM 2D topology which can be seen in Figure 1 (b) and (c) respectively.

The HMM 1D is a HMM with a particular topology allowing only self-transitions and transitions to the next state. In speech recognition, for instance, this topology is referred to as a left-right HMM. The representation of a face image for an HMM 1D is obtained by decomposing the face into blocks and concatenating the blocks from the same row t to form a single feature vector x_t, see Figure 1 (b). This topology can only account for vertical transitions.

The HMM 2D (also called a Pseudo-2D HMM) is a HMM 1D, referred to as the "main HMM", where the emission probability is modelled by a HMM 1D, referred to as the "embedded HMM". The representation of a face image for a HMM 2D is obtained by decomposing the face into blocks and forming a sequence (along the rows) of feature vector sequences (where a feature vector is obtained from each block in the row) as illustrated in Figure 1 (c). The vertical sequence is modelled using the "main HMM" and the horizontal sequence is modelled using the "embedded HMM". This topology can account for vertical and horizontal transitions.

These three topologies (for the GMMs, HMM 1D and HMM 2D) have been shown to provide state-of-the-art performance for 2D face images (Cardinaux et al., 2004) when using DCT based feature vectors. However, only one of these topologies, the GMM parts-based approach, has been successfully applied to 3D face data (McCool et al., 2008). In that work, the local blocks from the 3D faces were encoded using two DCT based feature vectors. In the following section we outline several other possible DCT based feature vectors which can used to obtain the feature vectors from 3D faces using the above topologies.
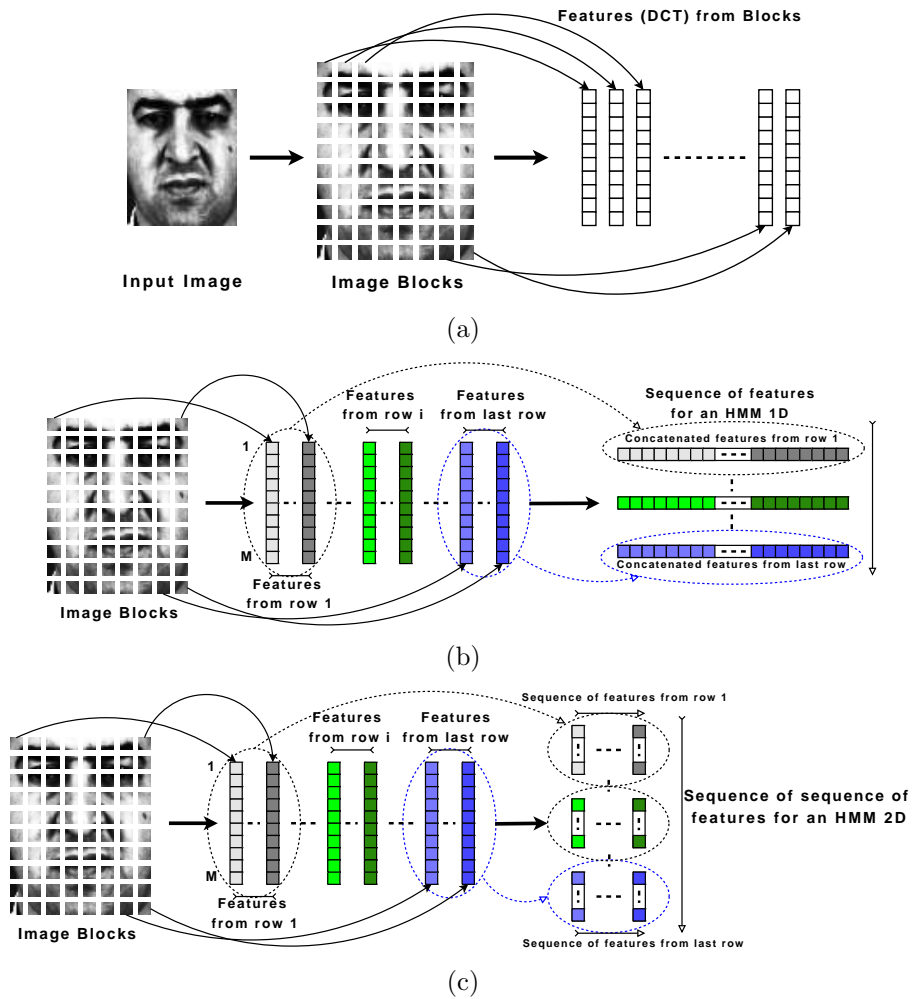
Figure 1: Three different topologies that have been used for feature distribution modelling techniques. The topology in (a) breaks the face into separate blocks. In (b) the topology collates feature vectors from a row (to form a super feature vector) and in (c) the topology first consists of vertical transitions (down the face) and then of a second set of horizontal transitions (across the face).

2.3.1. DCT Based Feature Vectors

A significant amount of work has been conducted to find the optimal feature vectors to use with the parts-based GMM approach for 2D face data (Sanderson and Paliwal, 2002a,b). Of these, the most commonly considered features are those based on the DCT because of their efficiency and relative accuracy (Sanderson and Paliwal, 2002b; Cardinaux et al., 2004).

7

Sanderson and Paliwal (2002b) presented several variants of the DCT feature vectors. These variants are based on using delta coefficients which represent the transitional information, in this case in the frequency domain. As an example the DCTdelta feature vector (Sanderson and Paliwal, 2002a) consists of only vertical(Δ^v) and horizontal(Δ^h) delta coefficients. The delta coefficients are obtained by applying a window to a particular block (but considering it as a feature vector) and taking the adjacent blocks (feature vectors) in the following manner:

$$\Delta^v c_n^{(b,a)} = \frac{\Sigma_{w=-W}^W w h_w c_n^{(b+w,a)}}{\Sigma_{w=-W}^W h_w w^2}, \tag{11}$$

and

$$\Delta^h c_n^{(b,a)} = \frac{\Sigma_{w=-W}^W w h_w c_n^{(b,a+w)}}{\Sigma_{w=-W}^W h_w w^2}. \tag{12}$$

Where h_w is the window, b is the position of the block (feature vector) in the vertical axis, a is the position of the block (feature vector) in the horizontal axis, W is the size of the window and $c_n^{(b,a)}$ is the nth DCT coefficient. The delta coefficients are obtained by using a symmetric window around the central block (feature vector), similar to (Sanderson and Paliwal, 2002b) we use a window of $W = 1$ with a uniform function for h_w. In addition to the delta coefficients, Sanderson and Paliwal also defined the DCTmod feature vectors.

The DCTmod feature vectors are DCT feature vectors who have their first r coefficients removed. In (Sanderson and Paliwal, 2002a) the first three coefficients were removed as they were considered to be susceptible to noise. In (McCool et al., 2008), although not referred to as a DCTmod feature, the first (0th) coefficient was removed from DCT feature vectors obtained from 3D faces. For consistency both forms of the DCTmod features are considered in this work, removing the first ($r = 1$) and removing the first three ($r = 3$) coefficients. In addition to the DCT-based feature vectors defined by Sanderson and Paliwal a third set of feature vectors is explored.

The third and final set of feature vectors considered in this work is to perform mean and standard deviation normalisation. Mean and standard deviation normalisation is performed in order to provide a common basis for comparison. It is performed on an image by image basis such that the mean vector μ_I and the standard deviation vector σ_I are calculated from all the feature vectors (blocks) from the image I. The mean vector and standard deviation vectors are then applied to each feature vector from that particular image I. The list of DCT-based feature vectors that are examined in this work are presented in Table 1, mean and standard deviation normalisation is applied to all of these feature vectors and so it is not specified in this table.

DCT	$c_0 c_1 .. c_{n-1}$
DCTzero	$c_1 c_2 .. c_{n-1}$
DCTmod	$c_3 c_4 .. c_{n-1}$
DCTmod2	$\Delta^v c_0 \Delta^v c_1 \Delta^v c_2 \Delta^h c_0 \Delta^h c_1 \Delta^h c_2 c_3 c_4 .. c_{n-1}$
DCTdelta	$\Delta^v c_0 \Delta^v c_1 .. \Delta^v c_{n-1} \Delta^h c_0 \Delta^h c_1 .. \Delta^h c_{n-1}$
DCTmoddelta	DCTmod + DCTdelta

Table 1: Table listing the family of DCT feature vectors.

3. Database and Experimental Design

The database used for these experiments is the subset of the Face Recognition Grand Challenge (FRGC) database (Phillips et al., 2005) containing 3D face images. The same database, experimental design and normalisation procedure used in (McCool et al., 2008) is used here, a brief summary of the experimental protocol is given in Section 3.1.

The normalisation procedure consists of cropping and normalising the image. First the image is normalised using the procedure defined in (Chang et al., 2005) and then an additional cropping step is performed to ensure that none of the masked regions are included when extracting the face data in a block-based manner. The procedure of (Chang et al., 2005) can be summarised as follows: i) geometric normalisation is performed using four landmark points (the two eye corners, nose tip and center of the chin), ii) invalid data points are removed and replaced using interpolation of the valid data points, and iii) each image is mean and standard deviation normalised. Following from the work in (McCool et al., 2008) the final 3D face region is a square region consisting of 108×108 pixels (ensuring that none of the masked regions are included), an example of the full normalisation and cropping procedure is provided in Figure 2.
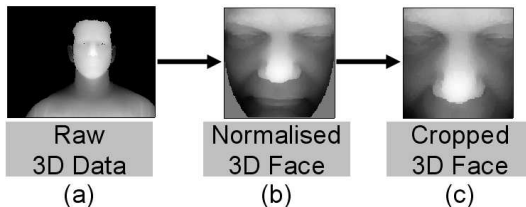

Figure 2: An example of the normalisation procedure applied to the FRGC data. The raw 3D face data (a) is normalised to produce depth values with an elliptical mask (b) which is then cropped to retain the most discriminative features of the face (c).

3.1. Experimental Protocol

A set of cross-validation experiments is used on the FRGC 3D face data. The database is evenly and randomly divided, based on ID, into four disjoint splits to form $split1$, $split2$, $split3$ and $split4$. These splits are then used to

form *Train*, *Tune* and *Test* sets with a ratio of 2:1:1, where: i) the *Train* set is used to train the background (or world) models, ii) the *Tune* set is used to determine parameters such as the number of components M and Hidden states, and iii) the *Test* set is used to produce final results (separately from the *Tune* set). Each experiment is then defined by assigning a split to either the *Train*, *Tune* or *Test* set, each split consists of approximately $1,150$ images. Using these splits four cross-validation experiments are conducted by cycling through the possible combinations, this procedure is illustrated in Figure 3 and more details can be found in (McCool et al., 2008).
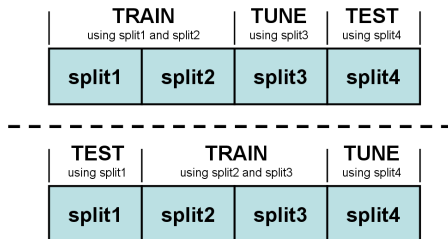

Figure 3: An example of how different *Train*, *Test* and *Tune* sets were designed for the cross-validation experiments.

There are four conditions that can be used with this cross-validation protocol. This is because the FRGC database was captured in three distinct periods, the Spring of 2003, the Fall of 2003 and the Spring of 2004, therefore, the four possible conditions are *All*, *Spring2003*, *Fall2003* and *Spring2004*. The particular condition refers to the period from which the enrollment data is randomly obtained. For example *Spring2003* means that the enrollment data is randomly drawn from data captured during *Spring2003* and the test data is drawn from the other periods (*Fall2003* and *Spring2004*). For the *All* condition the enrollment data is randomly drawn from all of the periods. When using the *Tune* set only the *All* condition results are used.

For these experiments the *Test* results obtained using the *All* condition are considered to be the most useful. This is because they consist of the largest number of client and impostor accesses with $3,061$ client accesses and $314,877$ impostor accesses. When presenting results the Equal Error Rate (EER) is used, this is the point at which the False Rejection Rate equals the False Acceptance Rate.

4. Results and Discussion

The main interest of this work is to compare the performance of a GMM parts-based system to the two HMM systems described in Section 2. Work conducted in (McCool et al., 2008) has provided some baseline results for a PCA system and a tuned GMM system which are used as a reference point, referred to as the PCA Baseline and the GMM Baseline respectively. However,

for these experiments a different implementation of GMMs is used and so for consistency a second set of GMM results is presented, referred to simply as GMM when presenting results.

To simplify our experiments a reasonable set of constraints were placed on the parameters that were examined when performing optimisation on the *Tune* set. This is because there are a range of parameters to optimise and certain factors are kept constant to ease the computational burden. First, the block size is set to a constant of $B = 16$ with an overlap of $S = 12$ based on work in (McCool et al., 2008). Second, the MAP adaptation factor is fixed to a constant value of $\alpha = 0.5$ for all of the GMM and HMM experiments; this alpha factor is significantly different to any of those examined in (McCool et al., 2008) because a different implementation is used. Third, the base DCT feature vector is kept to a constant dimension of $d = 15$. This value is chosen because it is often selected when using DCT feature vectors (for parts-based systems) for 2D face recognition (Cardinaux et al., 2004) and also because this keeps the dimensionality of the DCT variants within a reasonable limit, for instance for an $d = 15$ DCT feature vector a DCTmoddelta feature vector (the largest DCT variant examined) will have $d = 42$ coefficients.

4.1. Performance Comparison GMMs versus HMMs

Initial experiments were conducted on the *Tune* set to determine the optimal parameters. The optimal parameters were found for each system (GMM, HMM 1D and HMM 2D) by initially considering only the DCT feature vectors ($d = 15$). For the GMM system only the number of mixture components had to be derived, whereas for the HMM 1D system the number of hidden states H and the number of mixture components per state M both had to be derived. For the HMM 2D the number of hidden states for the "main HMM" H_{main}, the number of hidden states for the "embedded HMM" $H_{embedded}$ and the number of mixture components per state M all had to be derived. The optimal parameters on the *Tune* set were found to be:

- **GMM** $M = 512$,

- **HMM 1D** $M = 64$, $H = 10$, and

- **HMM 2D** $M = 64$, $H_{main} = 10$, $H_{embedded} = 10$.

Using these parameters the performance of the three systems were compared.

It was found that the performance of both HMM systems was consistently better than the GMM system. Examining the results in Table 2 it can be seen that the HMM 2D has half the EER of the GMM system with the EER reducing from 1.80% for the GMM system to 0.88% for the HMM 2D system. This result is greater than expected and shows the importance of being able to model the spatial relationship of the face blocks, which is something that the GMM system is not capable of doing.

The results in Table 2 provide an initial guide as to the performance difference between the GMM and HMM systems. However, to ensure that these

	GMM	HMM 1D	HMM 2D
EER	1.80%	1.21%	0.88%

Table 2: Table with the Equal Error Rates of the GMM, HMM 1D and HMM 2D systems for the *All* condition on the *Test* set.

results were consistent a more optimal set of DCT-based feature vectors was explored for the GMM and HMM systems. The results of these further experiments are summarised below.

4.2. Performance of DCT-based feature vectors

Several DCT feature vector variants were described in Section 2.3.1 and these are all analysed. Using the previous GMM classifier with $M = 512$ mixture components it was found that all of the DCT variants provided improved performance when compared to the basic DCT feature vector. Two consistent trends were found from these experiments, that: i) DCTmoddelta and DCTdelta feature vectors provide the best performance, and ii) mean and standard deviation normalisation provides a significant improvement in performance for all of the DCT based feature vectors. These results are presented in Figure 4.

Testing the GMM system with the DCT variants it was found that the DCTdelta and DCTmoddelta feature vector consistently performed well. In Figure 4(a) it can be seen that the DCTdelta and DCTmoddelta feature vectors perform the best and that when mean and standard deviation normalisation is applied the DCTmoddelta feature vector performs the best, see Figure 4(b). From these two figures it can also be seen that mean and standard deviation normalisation provides a significant improvement for all of the feature vectors, for example the performance of the DCTmoddelta features improves from 1.15% to 0.75% (*Tune* set results). A similar set of experiments were conducted for both of the HMM systems and it was found that the mean and standard deviation normalised DCTmoddelta feature vectors provided the optimal performance.
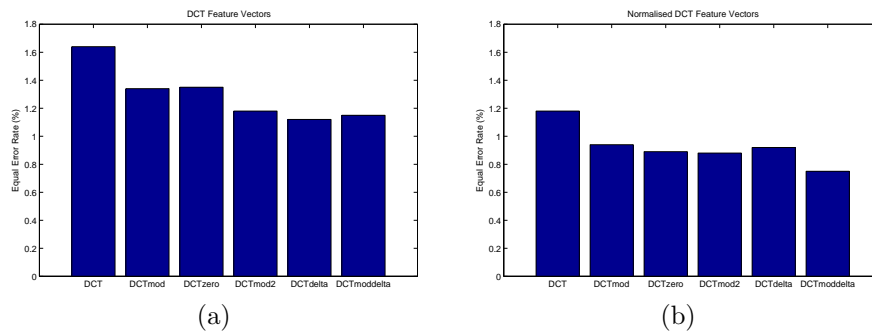

Figure 4: The Equal Error Rate (a) all of the DCT variants and (b) all of the DCT variants with mean and standard deviation normalisation for the GMM system on the *Tune* set.

12

In Figure 5 a summary of the performance of each system, GMM, HMM 1D, and HMM 2D, is presented on the *Test* set using the *All* condition. This figure shows the impact of using the HMM systems and also the effect of using a more appropriate feature vector (with mean and standard deviation normalisation). Both HMM systems clearly outperform the GMM system using the same feature vectors with the HMM 2D system providing significantly improved performance. For example when using the optimal DCTmoddelta feature (with mean and standard deviation normalisation) the GMM system has an EER of 0.88%, the HMM 1D has an EER of 0.72% and the HMM 2D has an EER of 0.36%.
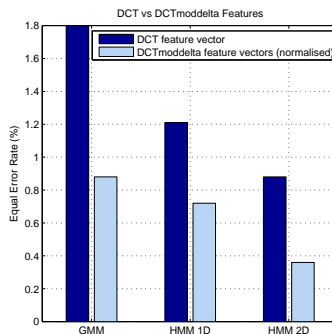
(a)

Figure 5: The Equal Error Rate of the DCT feature vectors versus the normalised DCT-moddelta feature vectors are presented for the *Test* set. It can be seen that the normalised DCTmoddelta feature vectors significantly outperform their DCT counterparts.

4.3. Summary

The full results of each feature distribution modelling system examined in this article are presented in Table 3. In this table the results of each system for each *Test* condition are provided, and it can be seen that a consistent result is obtained regardless of the enrollment condition that is used. Also presented in this table are the results for the previous baseline systems using the same database and protocol, it can be seen that both HMM systems provide an improvement in performance and that the HMM 2D system provides a significant improvement in performance.

	PCA Baseline	GMM Baseline	GMM	HMM 1D	HMM 2D
All	1.99%	0.88%	0.88%	0.72%	0.36%
Spring2003	2.01%	1.34%	0.89%	0.67%	0.42%
Fall2003	1.82%	1.08%	0.45%	0.51%	0.18%
Spring2004	2.43%	0.92%	0.92%	0.80%	0.40%

Table 3: Table with the Equal Error Rates of the GMM, HMM 1D and HMM 2D systems for the all of the conditions for the *Test* set.

13

5. Conclusions and Future Work

The use of state-of-the-art HMM systems has been shown to be directly applicable to 3D face recognition. This work has examined the applicability of two HMM topologies, the: HMM 1D and HMM 2D. Experiments on the FRGC 3D face database have shown that the HMM 2D system can more than halve the equal ERR of the equivalent GMM system with the EER reducing from 0.88% to 0.36%.

Analysis of alternate feature vectors was conducted by examining several DCT variants. A family of DCT feature vectors was explored, with and without mean and standard normalisation, and two things were found. First, the use of delta coefficients yields improved performance with the DCTmoddelta feature vector providing the best performance. Secondly, the use of feature vector normalisation also provides a significant and consistent improvement in performance. Furthermore, combining all of these factors together the initial GMM system reduces its EER from 1.80% to 0.88% and the HMM 2D system reduces its EER from 0.88% to 0.36%.

A consistent result from all of the experiments has been that incorporating spatial information improves the performance of the system. For instance at the feature level the DCTdelta and DCTmoddelta features perform the best for the GMM system (and also for the HMM system). While at the model level using HMMs to model the spatial information also improves the performance. And combining all of these facts yields a significant improvement in performance. For instance modelling both the horizontal and vertical transitions with the HMM 2D results in a halving of the EER, from 0.88% for the GMM system to 0.36% for the HMM 2D system.

Future work will examine the effect of using different normalisation procedures on the 3D face data. Also under consideration will be methods to more directly add spatial information within the feature vector.

6. Acknowledgements

References

Achermann, B., Jiang, X., Bunke, H., 1997. Face recognition using range images. International Conference on Virtual Systems and MultiMedia, 129–136.

Bishop, C. M., 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Bledsoe, W. W., August 1966. The model method in facial recognition. Technical report for Panoramic Research Inc.

Cardinaux, F., Sanderson, C., Bengio, B., 2004. Adapted generative models for face verification. In: International Conference on Automatic Face and Gesture Recognition.

Cartoux, J. Y., Lapreste, J. T., Richetin, M., 1989. Face authentication or recognition by profile extraction from range images. in Workshop on Interpretation of 3D Scenes, 194–199.

Chang, K., Bowyer, K., Flynn, P., 2005. An evaluation of multimodal 2d+3d face biometrics. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (4), 619–624.

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

G. Heusch, Y. R., Marcel, S., 2006. Local binary patterns as an image preprocessing for face authentication. International Conference on Automatic Face and Gesture Recognition, 9–14.

Lee, C., Gauvain, J., 1996. Bayesian adaptive learning and MAP estimation of HMM. Kluwer Academic Publishers, Boston, Massachusetts, USA, pp. 83–107.

McCool, C., Chandran, V., Sridharan, S., Fookes, C., 2008. 3d face verification using a free-parts approach. Pattern Recognition Letters, 1190–1196.

Phillips, J., Flynn, P., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W., 2005. Overview of the face recognition grand challenge. Proceedings of IEEE Conference of Computer Vision and Pattern Recognition 1, 947–954.

Reynolds, D. A., Quatieri, T. F., Dunn, R. B., 2000. Speaker verification using adapted gaussian mixture models. Digital Signal Processing 10, 19–41.

Samaria, F., Fallside, F., June 1993. Face identification and feature extraction using hidden markov models. Image Processing: Theory and Applications, 295–298.

Samaria, F., Young, S., 1994. Hmm-based architecture for face identification. Image and Vision Computing 12 (8), 537–543.

Sanderson, C., Paliwal, K. K., December 2002a. Fast feature extraction method for robust face verification. Electronic Letters 38 (25), 1648–1650.

Sanderson, C., Paliwal, K. K., 2002b. Polynomial features for robust face authentication. In: ICIP02. pp. 997–1000.