Influence of compression on 3D face recognition

Lorenzo Granai, J. Rafael Tena *, Miroslav Hamouz, Josef Kittler

Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, United Kingdom

Abstract

This paper studies how the performance of a 3D face recognition system is affected by compression. A novel lossy compression technique tailored for registered 3D data along with a scheme for 3D face registration and recognition are presented and the results discussed. The proposed scheme achieves a significant compression ratio (factor of 35) without the loss of recognition performance.

1. Introduction

The field of automatic face recognition has existed for over 30 years (Zhao et al., 2003) and it has recently received renewed attention especially with the arrival of biometric IDs. In particular 3D shape methods have attracted a lot of attention due to their invariance to illumination and pose, which are the two main factors limiting the performance of current 2D face recognition technologies (Bowyer et al., 2006). 3D face recognition is a very active research topic, in which new approaches are continuously proposed with encouraging results (Bronstein et al., 2005; Samir et al., 2006; Kakadiaris et al., 2007). However, many issues influencing large-scale use of 3D face recognition technology are yet to be resolved. An important problem related to biometric IDs is the compression of the data that facilitates person identification. Efficient compression of personal information is crucial to enable transmission over low-bandwidth networks as well as storage in low-memory devices or in large national databases. This is especially true in cases where the amount of data is high (e.g., 3D) or when the identification is done via combining different biometric characteristics.

This paper presents results the importance of which is three-fold. First, we present a novel recognition technique for 3D faces based on wavelets and LDA. Second, we introduce a compression scheme tailored for registered 3D face data which could in principle be also used for purposes other than recognition (e.g., generic 3D video coding). Third, the impact of compression on recognition performance is analysed. Although there exist general 3D mesh compression algorithms, the direct effect of the compression on the recognition performance has not been studied in the literature. Moreover, a compression method tailored for 3D faces promises higher compression yields than a generic technique. Fig. 1 shows a simplified representation of the most important components of the proposed recognition algorithm. 3D data describing a human face (3D client data) is first registered using the 3D deformable model algorithm introduced in (Tena et al., 2006) and explained in Section 2. The output of this operation is a set of 3D points which sample the input face at the locations specified by a generic model. The registered data that characterises a face is then compressed in a lossy manner as discussed in Section 3. This information can be transmitted over a network or stored in a low-memory device. When a client needs to be identified its data is uncompressed and then used for the recognition procedure, where it is compared with the client’s own data for verification or with several other clients for identification. The output is a binary decision which states whether the two samples of 3D data correspond to the same person or not.

2. Registration

Deformable surface fitting methods have been widely used to establish dense correspondences across different 3D objects of the same class. The technique we use here to register each face is the iterative dense registration algorithm proposed in (Tena et al., 2006),
The location of these landmark points were provided with the nose, and the tip of the chin, were used for global mapping. Marks corresponding to the outer canthi of the eyes, the tip of necessary to manually identify them for each face scan (Bookstein, 1989). In the experiments here reported, four land-
copies. The iterative algorithm has three distinct stages: (i) global is to be registered. The original scans are then discarded and the model has a negligible influence on the accuracy of the registration algorithm. In our experiments we use a model with higher vertex
concentration on facial regions, such as eyes, nose, and mouth,
where the rate of surface change is accentuated. The generic face model \( MG \) is deformed to the shape of each face scan \( MD \) that is to be registered. The original scans are then discarded and the deformed versions of the generic face model are kept as registered copies. The iterative algorithm has three distinct stages: (i) global mapping, (ii) local matching and (iii) energy minimisation; where the last two are repeated at different levels of detail of the generic face model.

During the global mapping stage, a set of landmarks are identified on \( MC \) and \( MD \). The two sets of landmarks are brought into exact alignment using the thin plate spline interpolation technique, which smoothly deforms \( MC \) minimising the bending energy (Bookstein, 1989). In the experiments here reported, four landmarks corresponding to the outer canthi of the eyes, the tip of the nose, and the tip of the chin, were used for global mapping. The location of these landmark points were provided with the database used in the experiments (see Section 5), so it was not necessary to manually identify them for each face scan \( MD \), but only once for \( MC \). Automatic landmark localisation is currently an eminent research topic (e.g., Azouz et al. (2006); Zhang and Wang (2007); Haker et al. (2007)). In the future, automatic landmarking could eliminate the need for manual annotation and allow for a fully automated registration process. In the local matching stage, for each vertex \( v_i \) on \( MC \), the closest vertex on \( MD \), within a variable search radius of \( r \), is found. This gives a set of matches between \( MC \) and \( MD \). If a match within \( r \) is not found for \( v_i \), then the mirror im-
age of the match of the corresponding vertex on the other side of the face is taken, given the existence of such a match. The assumption of human face symmetry is only enforced during the first iteration of the algorithm, so that if the assumption is invalid for a certain face, the algorithm can recover in the following iterations. For any vertex \( v_i \) in \( MC \) still without a match in \( MD \) a match is interpolated by averaging the matches of its neighbours. In the final stage of the algorithm, \( MC \) is conformed to \( MD \) by minimising a weighted sum of internal and external energy which is defined as
\[
E = E_{\text{ext}} + \epsilon E_{\text{int}},
\]
where the parameter \( \epsilon \) balances the trade-off between adherence to \( MD \) and maintaining the smoothness of \( MC \). The external energy attracts the vertices of \( MC \) to their matched points in \( MD \):
\[
E_{\text{ext}} = \sum_{i=1}^{N} |v_i - v_i'|^2,
\]
where \( v_i \) is the \( i \)th vertex of \( MC \) with \( 1 \leq i \leq N \) and \( v_i' \) is its most similar point in \( MD \). The internal energy constrains the deformation of \( MC \) thus maintaining the original local mesh structure:
\[
E_{\text{int}} = \sum_{i=1}^{N} \sum_{j=1}^{K} |v_i - v_j'|^2 - |v_i - v_j|^2,
\]
where \( K \) is the number of neighbour vertices of the \( i \)th vertex, \( v_i' \) and \( v_j \) are neighbouring vertices and \( v_i \) and \( v_j \) are their original positions in \( MC \). The fitting error is further reduced by iterating the local matching and energy minimisation stages a total of four times. The last two iterations are done at a higher scale of detail, which is achieved by subdividing the polygons of \( MC \) to create a finer mesh – see Fig. 2. Further details and evaluation results on the registration algorithm can be found in Tena et al. (2006).

It should be noted that the registration algorithm has been extended from its original version in Tena et al. (2006) to simultaneously register the texture image of the input 3D face scan, if available, to a generic image template (Tena, 2007). As this template is mapped to \( MC \), the texture maps of all input faces are also registered. The registered texture is not used in this work, but in future it could be exploited to improve recognition performance using fusion strategies.

3. Transform and compression

After registration, the client’s 3D data is represented by a set of \( N \) ordered points in 3D space, \( N \) being the number of vertices of the generic model \( MC \). In this work, \( N \) is equal to 2762. However, in Section 5 we also subsample the registered data to reduce \( N \) to 706, allowing us to compare our compression technique against subsampling. The structure of \( MC \) defines how its vertices are ordered. In general, \( MC \) can be seen as a list of vertices ordered in a random fashion (see Fig. 3). This means that a point located in the forehead region may be followed by a point located in the chin region. Although this ordering is irrelevant for 3D data representation, it has a direct impact on compression. A permutation in this order can be introduced if this is fixed and a priori known in order...
to enhance data predictability, we determine a new order for the points of the model $M_G$ used for registration, so that the global length of a line connecting all the points is minimised. This fact makes our compression algorithm tailored and exclusive to model-based 3D registration methods, because significant compression gains are made on modifying the ordering of the vertices of the generic model to which all 3D data is registered. Finding the best point ordering is equivalent to solving in 3D an open travelling salesman problem (TSP, see Applegate et al. (2006)), i.e., a TSP without coming back to the starting point. The complexity of solving such a problem can be very high and yet no effective solution method is known for the general case. However, there exist fast suboptimal methods that are able to provide the shortest path in the majority of cases the complexity or dimensionality of which is not too high. The particular order adopted in this work has been computed by the algorithm “concorde” (see Applegate et al., 2006).

Given $N$, the resulting permutation $\phi$ is fixed and dependent on the 3D model only and therefore it can be assumed to be known to the decoder, together with the model. Moreover, the permutation needs to be computed only once. Of course this does not guarantee that the actual points of any registered face are sorted according to the shortest path. However, since their positions are not far from the points of the model, it is likely that their order will not be far from optimal and hence be beneficial for compression purposes. With respect to the results in Section 5, we can observe that such a permutation has improved the compression ratio by about 50%, given the same quantisation parameters. For instance, when quantisation parameters $Q_l = 1, Q_w = 1$ were used, a bit load of almost 4 kbytes was achieved; this was further reduced to 2.1 kbytes when reordering was introduced (see Fig. 4). Fig. 3 shows a line connecting all the 2762 vertices of our generic face model, following the original order and the new order determined by solving the open 3D TSP. The improved data predictability is evident.

Once the points have been reordered, we perform a zero order smooth padding, which slightly increases the data dimensionality from $N$ to $M = 2768$. This is necessary in order to be able to apply a dyadic wavelet transform. Formally, we call $(w_i : w_i = (x_{w_i}^i, y_{w_i}^i, z_{w_i}^i))_{1 \leq i \leq M}$ the vector of 3D points after the padding. The point coordinates are then transformed using the Daubeches 9,7-Biorthogonal Discrete Wavelet Transform (DWT, see Cohen et al. (1992)). Four decomposition levels are used, motivated by empirical results. The DWT is implemented via a lifting scheme, which has a complexity of $O(M)$ (Sweldens, 1996). Each axis in the 3D space is treated independently, yielding three 1D sets of coefficients. The vectors containing the wavelet coefficients have size $M$ and are:

$$X = \text{DWT}(x^{w}),$$
$$Y = \text{DWT}(y^{w}),$$
$$Z = \text{DWT}(z^{w}).$$

Fig. 4. Left: average size (or Bit-Rate) versus Equal Error Rate. Note that the lower bound is given by the EER of the uncompressed data which is 9.3%. Right: average size (or Bit-Rate) versus mean geometric distortion (see Eq. (5)).
3.1. Compression

The wavelet coefficients are compressed following a scheme similar to the one presented in Granai et al. (2006) for a single frame of a 3D video. The three low-pass subbands are coded in a differential way (DPCM) and uniformly quantised with step $Q_l$. Every other subband of every axis is quantised using a uniform dead-zone quantisation, with step $Q_w$. The amplitude of the dead-zone is twice the quantisation step $Q_w$, and the ratio between $Q_l$ and $Q_w$ is variable (see also Section 5 for the actual values used in the experiments). Adaptive Arithmetic Coding (AAC) is finally used to reduce the entropy of the quantised wavelet coefficients: an adaptive model is computed for every subband, without making a distinction among $X$, $Y$, and $Z$. Note that due to the use of a generic 3D face model there is no need to compress the 3D connectivity, because this is fixed and known at the decoder side. This saves a significant amount of information, as explained in next section.

3.2. Decoding

The decoding procedure, necessary to reconstruct the 3D shape information from the binary compressed data is simple and very fast. We suppose that the 3D model $M_n$ and the permutation $\pi$ are known to the decoder. Indeed the permutation only depends on $M_n$ and thus it is fixed for all 3D faces. All the other information necessary for the decompression is in the header-file (quantisation steps, wavelet decomposition levels, AAC parameters). Now one has only to apply an inverse quantisation in order to find the wavelet coefficients that are used for recognition. These are approximations of the vectors $X$, $Y$, and $Z$ in Eq. (4). If one additionally wants to reconstruct the 3D information, based on the model, an inverse DWT has to be performed, followed by an inverse permutation.

4. Recognition

The registered face surface data can be processed by the recognition module which computes a similarity score between two 3D feature vectors. In our current implementation, the feature vector $\gamma$ is created by concatenating the unquantised wavelet coefficients of the 3D coordinates of each vertex, computed following the method previously illustrated: $\gamma = [X \ Y \ Z]$. Its dimension is $3M$. The recognition algorithm is based on Linear Discriminant Analysis (LDA) and Normalised Correlation (NC). In principle any metric classifier could be used in this proposed compression scheme. The LDA + NC classifier is an appropriate choice for this study because it is a representative of mainstream face recognition methods and is currently used in industrial grade biometric applications.

4.1. LDA + NC

LDA is a feature extraction technique which was introduced into the field of face recognition by Belhumeur et al. (1997). It aims at finding a small number of features which maximise the discriminative capabilities within the feature space. A linear transformation of the original data space which maximises the ratio of the mean between-class scatter to the mean within-class scatter is computed from the training data. In the context of 2D face recognition with sparse training sets, it has been previously observed that the Euclidean metric is not the optimal metric to use with the LDA representation (Kittler et al., 2000). Greater classification accuracy can be obtained by using the angle between two vectors as a measure of similarity. This is the basis of the normalised correlation technique in which the cosine of the angle between a probe vector and class mean vector is used as a similarity measure. The combination of LDA + NC is a very fast classifier as only simple mathematical operations are required to obtain a similarity score given the input vectors. Such complexity is desirable in large-scale identification scenarios where millions of tests must be performed to find the closest match.

5. Experimental results

In this section we provide and discuss the results obtained by performing verification tests on the Face Recognition Grand Challenge database (FRGC, see Phillips et al. (2005)). The FRGC is currently the largest available 3D face database with a rigorous evaluation protocol attached. It consists of 4950 3D face scans. The protocol denoted as "V2,Exp3" defines a verification experiment resulting in more than 16 million verification comparisons (or verification – client or impostor – accesses). A decision threshold$^1$ is used on the similarity scores produced by the recognition module and a binary decision (Accept or Reject) is given. Finally, standard Equal Error Rate (EER) is computed for this experiment from the similarity scores given by the recognition module. EER is obtained from the receiver operator characteristic curve as the point which gives the same false acceptance and false rejection rate. The high number of tests makes the EER estimate reliable and we believe that the results reported here are significant. This database has been widely used in many evaluation experiments. For example in Kakadiaris et al. (2006) and Passalis et al. (2007) the authors present a wavelet-based approach which gives very high recognition rates. However, compression related issues have not been addressed in these studies. For training the LDA classifier, 912 densely registered 3D scans from the Face Recognition Grand Challenge training partition have been used. More details on the Face Recognition Grand Challenge, the database and the protocol can be found in Phillips et al. (2005).

5.1. Without losses

In order to verify a personal identity, one needs to have the wavelet coefficients $X$, $Y$, and $Z$, i.e., 3M single-precision floating-point numbers, with a total size of around 32 kbytes. Of course, this data can be losslessly compressed with standard techniques (for example Huffman coding, Run-length encoding or LZW, see Sayood (2000)). Experimentally we have observed a small reduction of the size down to around 30 kbytes if these techniques are employed. As already explained, the recognition is based on the wavelet coefficients, following the technique described in Section 4. The EER obtained without compression (or with lossless compression), i.e., with unquantised wavelet coefficients is 9.3%. This constitutes the baseline reference to measure the effects of our compression method. Performing the classification in the wavelet domain therefore brings a small performance gain with respect to the same recognition method working directly with the spatial coordinates of the 3D points, for which the EER was 9.4% (Granai et al., 2007).

5.2. Introducing compression

Here we analyse how the lossy compression scheme proposed in Section 3 affects recognition performance. Fig. 4 shows how the EER increases as compression gets higher, using coarser quantisation steps for the wavelet coefficients. The size of the data representing a 3D face goes from 2.12 kbytes ($Q_l = 1$ and $Q_w = 1$ at the extreme right-hand side of the figure) down to 169 bytes ($Q_l = 32$ and $Q_w = 40$). The shape distortion introduced on the

---

$^1$ Working on the validation set of the database, the threshold is varied in order to obtain a receiver operator characteristic curve that describes the performance of the system.
3D data, by the minimum and maximum compression levels used in our study, is shown in Fig. 5. Considering the last two points on the Size-EER curve it is interesting to notice that the error remains stable at 9.3% and it is not affected by compression as long as the quantisation is fine enough ($Q_l \leq 4$, $Q_w \leq 8$). This means that we can reduce the file size of the 3D data down to 930 bytes (a compression ratio higher than 35) without losing any accuracy in recognition. Although compression causes loss (data cannot be reconstructed exactly), from a recognition point of view this loss is negligible since it does not affect the results of the verification experiment. From left-hand side of Fig. 4, we can observe that the EER starts to increase sharply when the average file size decreases below 500 bytes. In general one can observe that the Size-EER behaviour is strongly non linear. It can be approximated by two straight lines, one with a very high slope (i.e., interpolating the first points with higher error), the other one almost horizontal, interpolating the high quality data points (on the right). The breakpoint identifies the most interesting area, where the compression ratio is high and its impact on the EER still negligible. The second and third point from the right in the figure are obtained with $Q_l = 4$ and $Q_w = 1$ and 8, respectively. These are the quantisation parameters that give the most interesting results.

The high non-linearity of the error can be confirmed by a comparison with the right-hand side of Fig. 4 which illustrates the Size-Distortion behaviour of the proposed compression algorithm. With $\{v_i\}_{1 \leq i \leq N}$ being the list of vertices ordered according to the 3D model, such that $v_i = (x_i, y_i, z_i)$, the distortion is measured by considering the mean Euclidean distance $\Delta$ between original points $v_i$ and the reconstructed points $\hat{v}_i = (\hat{x}_i, \hat{y}_i, \hat{z}_i)$:

$$\Delta = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2}. \tag{5}$$

$\Delta$ can be expressed in millimetres. The same points are used to plot the two curves and it is easy to observe how the small distortions introduced by compression do not affect recognition performance. We can thus argue that the strong non-linearity introduced by classification/recognition increases robustness against the distortions caused by compression.

Since the recognition error increases when quantisation is too coarse, we tested the possibility of reducing the number of points in order to further compress the data. In particular, while the registration of the 3D data was carried out using all 2762 points, the recognition was performed using only a subset of 706 points, i.e., the subsampled points shown on the left-hand side of Fig. 2. However, we found out that compression performance does not improve at all. As shown by the Size-EER curve in Fig. 4, the best EER rate obtained using the subset of 706 points was 9.6% at 896 bytes, while using the full set of 2762 points we could achieve a similar EER at only 600 bytes. This reduction in recognition performance caused by subsampling confirms the observations already made in Granai et al. (2007). Our explanation is that the information lost by quantising the wavelet coefficients is less important than the one lost when fewer points are used. Adopting a coarse grid does not take into account the fact that not all the points in the face contain the same amount of information, while the wavelet-based approach can, at least partly, address this problem. It must be said that there exist general algorithms for 3D mesh compression (see Smolic et al. (2007)). A major advantage of our scheme is that the connectivity is based on the generic face model and therefore does not need to be transmitted. Basically, the 3D registration allows us to perform compression of a list of vertices only, since the connectivity is known at the decoder together with the generic model $M_c$. A typical 3D compression algorithm is Edgebreaker, which however focuses mainly on 3D topology compression (Rossignac, 1999). Rossignac states that the best lossy vertex list compression algorithms use approximately 12 bits per vertex. The equivalent rate of our proposed algorithm is approximately 3 bits per vertex, without introducing any degradation of the recognition performance. This important gain that the proposed approach achieves compared to the state-of-the-art can be explained by the fact that the structure of the data is taken into account. We know that the 3D vertices are the output of the registration process discussed in Section 2. This fact not only relieves us from considering the connectivity, which is included in the model, but also gives us information about the data structure, information which is exploited in the reordering procedure.

6. Conclusion

A novel wavelet-based 3D face compression technique was introduced and its effects on a state-of-the-art 3D recognition algorithm investigated. The proposed method yields extremely high compression ratios. It was proved experimentally that the size of registered 3D data can be reduced by a factor of up to 35 without affecting the recognition performance. Moreover, the compression can still go even higher, at the cost of degradation of the recognition accuracy. As a drastic application example, such high
compression ratios facilitate storing high-resolution 3D biometric recognition data on a 2D barcode as shown in Fig. 6 (ISO/IEC16022, 2006). In the framework proposed, the influence of the compression on the identification performance (see Fig. 4) is studied rather than investigating compression vs. distortion, as is usually done within the compression community. However, for the sake of completeness, distortion results are reported here as well. The current application needs emphasise the importance of a joint study of 3D compression and recognition algorithms.

Acknowledgment

Special thanks to Theo Vlachos.

References


