Compact 3D Model Generation based on 2D Views of Human Faces: Application to Face Recognition

Kostas Karpouzis, George Votsis, Nicolas Tsapatsoulis and Stefanos Kollias
Department of Electrical and Computer Engineering
National Technical University of Athens
Heroon Polytechniou 9, 157 73 Zographou, Greece

Abstract. A face recognition and synthetic-natural hybrid coding tool involving frontal and profile views is presented in this paper. The tool utilizes the above mentioned 2D images -upon which certain protuberant points are automatically detected- and adapts a generic 3D head model (polygon mesh) according to the predetermined information gained by the available views. This mesh provides shape information which -combined with texture information- is important for the robustness of a recognition system. The main advantage of the proposed approach is its ability to overcome constraints raised from arbitrary variations in scale, rotation and orientation, which are dominant deterioration factors in the identification task. Besides the face recognition aspect, applications which involve geometry transmission, such as teleconferencing, can take advantage of the proposed algorithm's ability to parametrically describe a complex organic model, such as a human head.
During the texture map creation process, issues related to the luminance differences and rotation variance between the available views, are successfully dealt with. Regarding the adaptation of the polygon topology of the 3D model, we apply a set of localized transformations, in order to preserve the continuity of the human head surface. Besides this, the problem of the minimum organic model representation required is addressed. The aspect under which this issue is verged upon, is that of the solution of a trade-off problem between low computational complexity and high approximation quality.

Key words: 3D Face Modeling, Automated Feature Extraction, Geometry Compression

1. Introduction

Computer recognition of human faces has been an active research area for more than two decades. Humans can easily detect and identify faces in a scene with very little or no effort. However, designing an automated system to perform this task is very difficult. Several techniques have been proposed in order to improve the efficiency of these kinds of systems, most of which are based on single, frontal view facial images, with constraints posed on the lighting conditions as well as on the scale, rotation and orientation of the face in the image. Approaches to face recognition from frontal view images involve structural or statistical features of the human face as well as template matching and global transformations [2, 9]. Alternative techniques utilize profile views for the recognition task. Most of these approaches depend exclusively on prominent

Fig. 1. Benchmark points in profile and frontal views.

Derived features in frontal view	Description	Derived features in profile view	Description
$d(14,15)/d(4,9)$	meas. of face width	Angle $1-2-3$	measure of chin
$d(12,13)/d(14,15)$	dist. between eyes	Angle $7-8-9$	measure of nose
$d(5a,5b)/d(14,15)$	mouth length	$d(8,10)/d(2,8)$	high/low slung
$d(4,6)/d(14,15)$	mouth width	Angle $5-6-7$	upper lip measure
$d(12a,12b)/d(14,15)$	eye aperture length	Angle $3-4-5$	lower lip measure
$d(12c,12d)/d(14,15)$	eye aperture width	Angle $4-5-6$	angle between lips
$d(8,9)/d(14,15)$	nose length		
$d(8a,8b)/d(14,15)$	nose width		

Tab. 1. Derived features for an holistic head mesh description (see also Figure 1). Note: $d(i,j)$ is the Euclidean distance between points i and j.

visible features of the profile view such as nose and chin [4]. However, recognition based only on these features ignores texture information and is therefore insufficient.

In our approach, frontal and profile images are combined to create a texture map which is cylindrically projected onto an suitably adjusted 3D head mesh. Initially, several protuberant points (let us call them benchmark points) are automatically detected in both views. These points are then used in two different ways; on the one hand to produce geometric measures (see Table 1, Figure 1), based on which prominent facial features on the head model are modified, and on the other hand to find the correct mapping between the pixels of the available views and the vertices of the head mesh.
Although the proposed modeling tool was initially applied to a face recognition scheme,

it may also be used in many other forthcoming technology applications. A characteristic example may be the use of the proposed technique in shape recovery of a face that participates in a telepresence application, where the facial views are acquired from -at least- two different cameras. While the reconstruction of a human head can also be achieved with the utilization of other algorithms, the presented approach results in a parametrical description of the head mesh; thus, a client computer can reproduce the original head by just using the extracted set of parameters. Furthermore, it may also prove to be a significant assistance in 3D video generation [10] and in enhanced object manipulation and in virtual navigation [13].

2. Detection of benchmark points

The benchmark points in frontal and profile views are shown in Figure 1. The automated detection of these points is our primary goal. It crucially affects the accurate alteration of the 3D model. Thus, the reliability of the point detection procedure is a critical issue. In frontal views, where the detection is more elaborate, a dual approach can be considered to bestow the certainty needed; a hybrid method using template matching [2] and Gabor filtering [8] is fortified by the use of the eigenfeature theory [9]. In profile views, a geometrical method is adopted.

2.1. Detection in frontal view

The fundamental concept upon which the automated localization of the predetermined points on the frontal images is based, consists of two steps: the hierarchical and reliable selection of specific blocks of the image and the subsequent use of a standardized procedure for the detection of the required benchmark points.
The detection of blocks describing facial features relies on the effective detection of the most characteristic feature. By adopting this reasoning, the choice of the most significant feature has to be made. The outcome of surveys [4] proved the eyes to be the most crucial and easy to be located facial feature.
The basic search for the desired feature blocks is performed by a simple template matching procedure. The comparison criterion used in practice is the maximum correlation coefficient between the prototype and the repeatedly audited blocks of an appropriately selected area of the face. In order to restrict the search area as much as possible, we use the knowledge of the human face physiology. The algorithm has a satisfactory performance, even in cases of small violations of the initial limitations.
Detection using eigenvectors is an alternative to template matching technique which allows greater range of distortions (lighting, rotation and scaling) in the input image. In this method, each pattern in the input image is associated with a reconstruction error. This error, referred to as "distance-from-feature-space", is computed by using the projection of the pattern on a limited number of eigenvectors and then using the projection

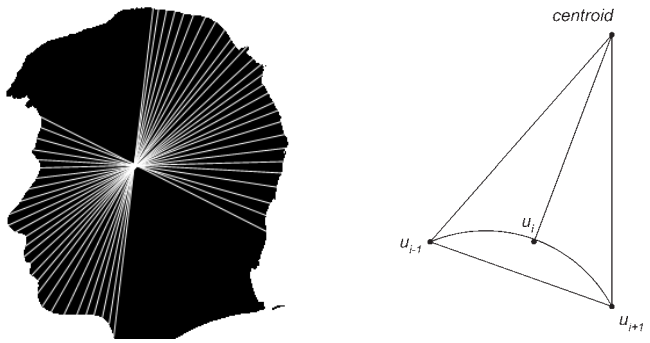


Fig. 2. (a) Segmented binary profile (b) Local curvature estimation.

coefficients to reconstruct it. Actually projection and reconstruction using only the first two eigenvectors leads the reconstruction error to be minimized in the position of the best matching image pattern [9].
The final block selection by the mere use of the above methods has not always been crowned with success. For that reason, the use of Gabor filtering was deemed to be one suitable measure of reliability. As it can be mathematically deduced from the filter's form [8], it ensures simultaneous optimum localization in the natural space as well as in frequency space. The filter is applied to both the selected area and the template in four different spatial frequencies. Its response is regarded as valid, only in the case that its amplitude exceeds a saliency threshold. The area with minimum phase distance from its template is considered to be the most reliably traced block.
After having isolated the regions of interest from the frontal image, the localization of the predetermined points ensues. This task is achieved by the use of integral projections of those regions towards the vertical and horizontal direction. These projections give such wholistic information about each feature in both directions, that they enable an intuitive detection of the benchmark points. It is obvious that the whole search procedure has been attempted to be as close to human perception as possible.

2.2. Detection in profile view

Feature points extracted from profiles were used in the earliest approaches of face recognition. In our approach the benchmark points lie exclusively in the profile outline. Thus the first step is to convert the profile view into a binary image from which the profile line can be easily extracted. The profile line in the binary image can be considered as a one-dimensional signal and the benchmark points on it represent the local extremes of this signal. The local curvatures are then estimated in a geometrical way. The basic

concept, upon which our method is based, uses the centroid of the binary profile image as a point of reference. Straight lines, passing by the centroid, segment the binary profile, as it can be seen in Figure 2(a). Each pair of the created triangular slices may easily define an approximation of the local curvature, by using a simple computation of the Euclidean distance between the intermediate point u_i and the line segment joining u_{i-1} and u_{i+1} (Figure 2(b)):

$$D = \left\| u_i - \frac{1}{2}\left(u_{i-1} + u_{i+1}\right) \right\| \tag{1}$$

The decision about the convexity or concavity of the profile points can be estimated by comparing the distance rl_i, between the segment l_i and the centroid, and the distance r_i, between point u_i and the centroid. By performing a serial search, one may track the local extrema of the whole curve.

3. Texture map creation

The next step involves combining the two views into a single texture map. Three main factors are affecting this procedure:
First, the two views should be perpendicular which is not always true, since in many cases the available frontal or profile view or both of them are rotated. An affine coordinated based reprojection algorithm, presented in [11], is performed to appropriately rotate the available views.
Illumination conditions in frontal and profile view are generally different and therefore directly combining them leads to an inadequate texture map. Histogram equalization is performed in both views to smooth the illumination differences.
Cylindrical projection of an image introduces an amount of distortion and to compensate for it, a warping transformation is executed. This transformation is a main part of the texture map creation procedure which is described in the following paragraph.

The texture map is created by placing regions of the frontal and profile views side by side. Columns from the frontal view are arranged in the center region of the texture map while the ones of the profile in the left and the right side of the texture map. At the center column of the texture map we place the column of the frontal view which corresponds to the *'nosepoint'* (benchmark point 8). The remaining columns are distorted according to the following transformation [12]:

$$\mathbf{r} = radius \cdot \arctan\left(\frac{\mathbf{x}}{radius}\right) \tag{2}$$

where $\mathbf{r}$ is the distance, in pixels, from the center column of the texture map, $\mathbf{x}$ is the distance from the column of *'nosepoint'* in the frontal image and, *radius* is the radius of the cylinder. The last columns copied from the frontal view are the ones which correspond to the outer corners of the eyes.

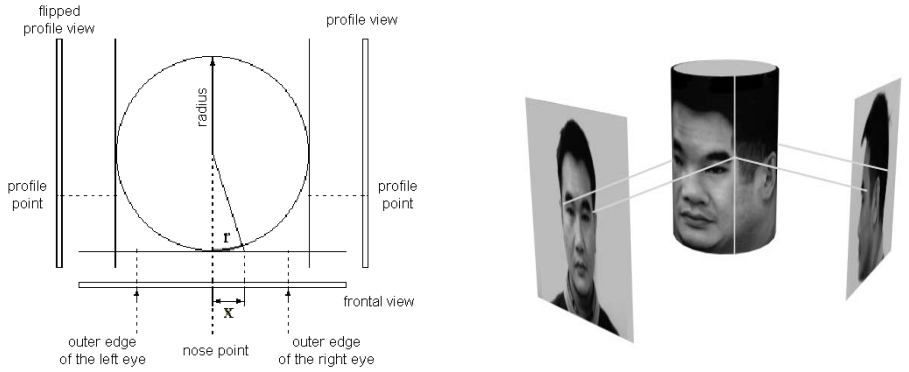


Fig. 3. (a) Texture map creation principles (b) Combination of the two views into a unified texture map.

Fig. 4. An example of a unified texture map.

In a similar way we arrange the columns of the profile in the left and right side of the texture map (see Figures 3(a) and 3(b)). Columns of the texture map which are not assigned values are created using interpolation. Interpolation also balances the local illumination differences between boundary regions of the frontal and profile views. Finally columns of the texture map which correspond to the back of the head are created by distorting the rightmost column of the profile to the right. A unified texture map, distorted using the previously described algorithm, is depicted in Figure 4.

4. Adaptation of the head mesh

The next task is to map the texture, the benchmark points and the extracted features to the generic head mesh; this is performed in three different steps. In the first step we apply mapping coordinates [1] to each vertex that the mesh consists of. Mapping coordinates are parameters in texture space that map every vertex of the head mesh to a relevant point in the texture map, i.e., the extended image and thus providing the renderer with the correct color for it. We choose to apply cylindrical mapping coordinates to the model because only a simple mathematical transformation is required to calculate them, given the Cartesian coordinates of each vertex and because approximating the human head with a cylinder seems to work better than other usual mapping modes. Cylindrical mapping does not cover areas of the head that are perpendicular to its axis, e.g. top of the head, but these can be rendered with other techniques that do not require photographs (e.g. particle systems). Correct positioning of the texture map is achieved through matching the benchmark points of the texture to those of a template that is created by unfolding the polygon mesh onto the imaginary mapping cylinder. After cylindrical coordinates (r, ϕ, z) are calculated, the ϕ coordinate is used to position a vertex in the template horizontally and the z coordinate for the vertical positioning. After all edges are drawn, the result is an image in which the benchmark points are known (from the mesh) and can be pre-calculated. Generally it is not possible to match all benchmark points of the texture image to those of the template without any modification, because the generic head mesh that we use cannot efficiently replicate every possible human head. In order to correct such differences we compare the derived features of the texture to those of the template. The derived features of the texture are generally not equal to those calculated from the original pictures, due to the distortion we introduced in order to combine them in the cylindrical texture map, but can easily be recalculated given the distortion transformation (see Equation 2). The derived features are used in a feedback manner to scale the relevant parts of the head mesh in order to match those of the human head. This creates a mesh that is sufficient for our purposes after a relatively small number of changes; minor differences that may occur usually can be obscured with the use of texturing [3, 5] and do not usually pose a problem in situations where extreme realism is not required.

The adaptation of the vertex and face topology of the model is not a straightforward task, since it is necessary to preserve the smoothness of the human skin and facial features. This means that, instead of applying required transformations in a uniform manner, we calculate vectors of weights for each vertex and each case of adaptation to a specific feature. The vertices that are precalculated to correspond to the benchmark points are scaled at full weight, which is necessary in order to position them at the correct point in space and match the derived features. All other vertices in the same neighborhood are scaled at gradually descending weights, in order to secure the continuity of the region.

In addition to that, the mesh is divided to specific regions that are individually scaled to match a specific measure. This ensures us that the influence of each scaling transformation is local to the neighborhood of each benchmark point and does not alter the position of all vertices of the model, except when this is implied by the derived feature, e.g. face width.
Rendering the mesh is a straightforward procedure that can be achieved using either a commercial or a proprietary program. In order to test these algorithms, we developed a small modeling and rendering front-end, based on the HOOPS cross platform library [6].This environment can be used to either automatically or interactively, modify the mesh by selecting sets of polygons and repositioning or scaling them. Global transformations and positioning of the mesh in an arbitrary angle can be easily accomplished, and results can be rendered into an image or a Postscript printer.

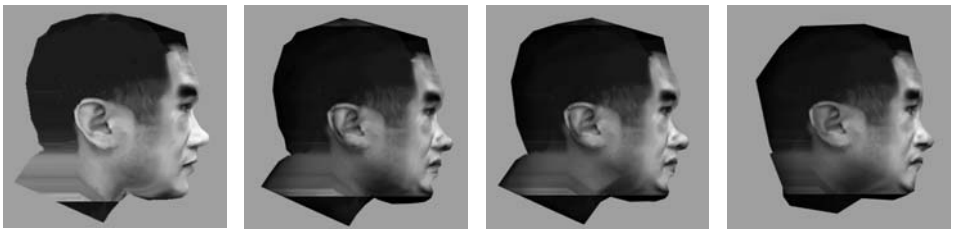
5. Scalability

In most cases, organic objects must be modeled with a large number of vertices and faces, in order to efficiently approximate the curvature of their surface. Because the time required to render or transmit a model through a network is at least linearly proportional to the number of faces in the model, a compression scheme can be adopted in order to speed up both transmission and rendering, in our case with a trade-off with the accuracy of the representation.
Geometry compression techniques can be either lossy or lossless, with wavelet encoding and compression being an example for the latter [7]. The lossy algorithm utilized can be used to create several versions of the original model, each with different number of faces and vertices. These versions are useful in virtual environments, where different levels of detail are required, or in a 3D model thumbnail scheme, where the user can be provided with a preview or scaled down mock-up of the model. Compression of the model is achieved with the reduction of adjacent faces and relevant edges, that are or can be considered coplanar. If we establish a threshold angle beneath which two faces are considered coplanar, then reduction can be expanded to include almost-coplanar faces. When this threshold increases, more faces and edges are reduced and, as a result, the quality of the representation deteriorates and the model can be rendered unrecognizable. For lower values of the threshold angle, we can achieve compression up to 68% with the resulting model still being easily identifiable.

In order to examine the influence of the model compression to the quality of the final rendering, we perform the following test: we render the model in the profile view for several compression rates (threshold angles) and investigate the variation of the measures defined in this view. In particular, consider the feature vector $\mathbf{f_{100}} = [f_1, f_2, ..., f_6]$, where f_1 is the angle defined by profile points 1,2 and 3 and f_6 is the angle defined by profile points 4,5 and 6, calculated from a profile image rendered from the highest resolution

Retained Vertices (%)	Faces	Vertices	Distortion Index (%)
100	2141	1180	0
93	1989	1101	2
91	1929	1065	3
80	1711	902	7
61	1307	763	26
32	680	415	57

Tab. 2. The influence of compressed models to the profile features



(a) (b) (c) (d)

Fig. 5. Synthetic profile views rendered from (a) 2141 faces (original), (b) 1929 faces, (c) 1307 faces, (d) 680 faces

model (see Table 1).
If $\mathbf{f_p}$ is the feature vector of a rendered from coarser model profile image, we compute the distortion index:

$$a = \frac{\|\mathbf{f_{100}} - \mathbf{f_p}\|}{\mathbf{f_{100}}} \tag{3}$$

The results presented in Table 2 depict that the compressed models are rather efficient while the threshold angle remains low, but significantly deteriorate the measures in profile views for larger threshold angles.

6. Applications

The applications in which the proposed natural-synthetic modeling technique may be used lie in several areas. Computer identification of human faces is one of them. The identification task can be summarized as follows: given a pair of frontal and profile images, search is performed in a database containing face images in different views, to find the one which best matches the input pair.
The matching scheme is a significant issue in a recognition system that crucially affects

(a) (b)

Fig. 6. (a) Synthetic frontal view of a member of the face database, (b) Synthetic view of a member of the database, rotated by 15^o

the confidence of identification [10, 11]. However, the specific problem is not faced, since our algorithm mainly aims to construct a sufficient 3D model, based on the input pair, rather than build the best recognition scheme. However, a general identification procedure could be performed in the following manner: during such a procedure, a rotated rendered view (see Figure 6(b)) is compared to the respective real views included in the face database, using the correlation coefficient as the matching criterion. The nearest neighbor is identified to be the target face. The used face database has been created at the University of Bern by Mr. Bernard Achermann, to whom we owe special thanks.

The proposed modeling tool may also be used in many other forthcoming technology applications. A characteristic example may be the use of the proposed technique in shape recovery of a face that participates in a telepresence application. This could mean the ability to introduce virtual reality systems, with contents based on real environments. It may also prove to be a significant assistance in 3D video generation [10].
Furthermore, other digital video applications seem to be promising as far as the use of this model is concerned. For example, application in enhanced object manipulation -which is definitely an area of interest for innovative digital video technology- offers new means for editing and compression, while virtual navigation experiences are made possible [13].

7. Conclusion

A generic 3D head model, which is modified using information from profile and frontal views, can be incorporated efficiently into a face recognition scheme, as well as in applications of 3D video generation and new digital video technology. Specifically in the former consideration, human head anatomy and specific geometric features are combined

to eliminate shortcomings inherent to recognition techniques which involve 2D information provided by a single view. Texture information is also included by mapping the views onto the modified head model. Limitations raised from inaccurate detection of benchmark points are crucial, therefore robust localization of the facial features is necessary. Besides this, the problem of the minimum resolution needed is addressed. Further work includes utilization of wavelet coding and lossless compression, instead of the lossy algorithm currently in use.

References

1992
[1] Watt A., Watt M.: Advanced animation and rendering techniques -Theory and practice. Addison - Wesley.

1993
[2] Brunelli R., Poggio T.: Face Recognition: Features versus Templates. IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(10), 1042-1052.

1995
[3] Bajaj C., Bernardini F., Hu G.: Automatic Reconstruction of Surfaces and Scalar Fields from 3D Scans. SIGGRAPH 95, Los Angeles, USA.

1995
[4] Chellapa P., Wilson C., Sirohey S.: Human and Machine Recognition of Faces: A Survey. Proceedings of IEEE, 83(5), 705-740.

1995
[5] Lee Y., Terzopoulos D., Waters K.: Realistic Modeling of Facial Animation. SIGGRAPH 95, Los Angeles, USA.

1996
[6] Leler W., Merry J.: 3D with HOOPS: building interactive 3D graphics into your C++ application. Addison - Wesley.

1996
[7] Stollnitz E., DeRose T., Salesin D.: Wavelets for Computer Graphics: Theory and Applications. Morgan-Kaufmann.

1997
[8] McKenna S.J., Gong S., Wurtz R.P., Tanner J., Banin D.: Tracking Facial Feature Points with Gabor Wavelets and Shape Models. Lecture Notes in Computer Science, Springer Verlag.

1997
[9] Moghaddam B., Pentland A.: Probabilistic Visual Learning for Object Representation. IEEE Trans.on PAMI, 19, 696-710.

1997
[10] Tai L.C., Jain R.: 3D Video Generation with multiple Perspective Camera Views. ICIP'97, Santa Barbara, USA.

1997
[11] Sengupta K., Ohya J.: An affine coordinate based algorithm for reprojecting the human face for identification tasks. ICIP'97, Santa Barbara, USA.

1997
[12] Tsapatsoulis N., Karpouzis K., Votsis G., Kollias S.: Analysis by Synthesis of Facial Images Based on Frontal and Profile Views. IWSNHC3DI'97, Rhodes, Greece.

1997
[13] Yeo B.L., Yeung M.,: Analysis and Synthesis for New Digital Video Applications. ICIP'97, Santa Barbara, USA.