Planar Bones for MPEG-4 Facial Animation

Manuel A. Sánchez Lorenzo
m.sanchez@dcs.shef.ac.uk

Steve C. Maddock
s.maddock@dcs.shef.ac.uk

Computer Graphics Research Group
University of Sheffield
Regent Court
211 Portobello Street
Sheffield, S1 4DP

Abstract

This paper proposes a deformation technique called Planar Bones, derived from Surface-oriented Free Form Deformations, and explores its application to the context of Facial Animation. The warping method proposed uses as the control input a polygonal mesh, which is built as a triangulation of the Feature Points defined by MPEG-4. A strong correspondence between the facial geometry and this control structure is maintained throughout the animation, overcoming the issues of many of the current systems defined using this standard.

1 Introduction

The Motion Picture Experts Group (`MPEG`) has devised a paradigm of Facial Animation (`FA`) encoding and compression that significantly improves the performance in transmission of FA data in comparison with other non content-oriented representations, such as compressed video. Indeed the advantages in terms of bandwidth requirement are enormous, moving from hundreds of kilobits per second, in the case of video capture, down to 2-5 kbit/s for an average MPEG-4 FA stream.

The method described by the standard [6] consists of the application of the usual MPEG compression techniques, like Discrete Cosine Transformation analysis, coefficient quantization and Run-Length Encoding, to a stream sampling the variation in time of 69 Facial Animation Parameters (`FAPs`). These describe, at a high level, face configurations as the weighted combination of a series of basic actions, generally related to a precise facial region, but also comprising global movements such as head tilt and jaw rotation.

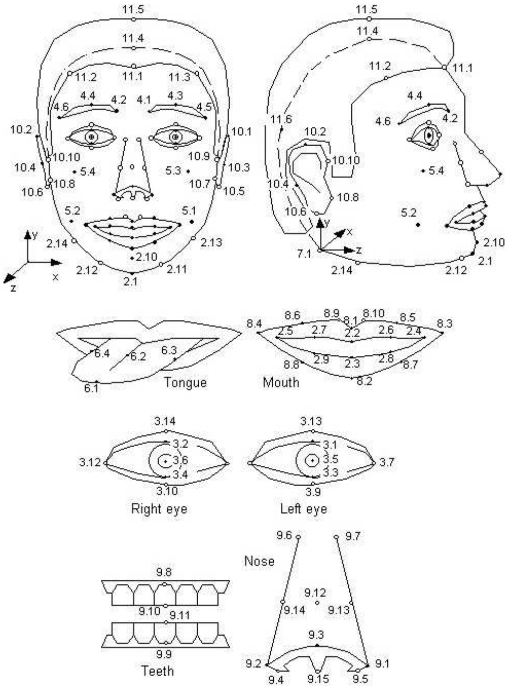


Figure 1. Feature Points distribution on a reference face.

These basic actions are expressed as displacements of a set of 84 Feature Points (`FPs`) mapped on the head, whose layout for a neutral face configuration is shown in Figure 1. The subsequent adaptation to a particular physiognomy is achieved by means of expressing the placement of these FPs in terms of five Facial Animation Parameter Units (`FAPUs`) (see Figure 2); these strive to capture the most distinctive characteristics that set apart several individuals. Not only the distri-

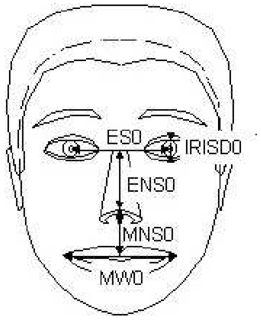


Figure 2. Facial Animation Parameter Units.

bution, but also the magnitude of the displacement of the Feature Points, while representing animation, is dependent on the FAPUs, thus allowing reuse of motion on different face anatomies.

The location of these FPs constitute the lowest level of abstraction that can be extrapolated from an MPEG-4 definition of a facial configuration, according to the standard. Therefore any animation system based on this standard for building a geometric representation of a face should take them as control input. The following subsection introduces the solutions given in the present literature.

1.1 Facial Animation systems based on MPEG-4

Some approaches, such as Wireface [7] or the more popular Candide [1, 16], use a very simplistic face representation built as a triangulation of the Feature Points. This allows direct translation of the output of the MPEG-4 decoding into animation. However the limited quality of the representation makes this method unsuitable for many contexts.

When a more detailed geometric representation of the face is required, its deformation has to be extrapolated from the MPEG-4 definitions. At a high level of abstraction, it can be parameterised directly by the FAPs, disregarding FPs. This is the procedure followed by Kalra and other MiraLab researchers [8, 9] in their reference implementation of an MPEG-4 Facial Animation client, MiraFace [7]. FAPs are used to control Free Form Deformation lattices that adapt the targeted geometry to reflect a given facial configuration. This volumetric warping technique [2, 17] performs a highly continuous deformation that maintains the smoothness of the skin; however it is not that convenient for handling contours and boundaries, such as the eye and mouth openings. Moreover, using a parameterisation based on FAPs requires specific setup of the control structures to fit every physiognomy, and the portability across different faces achieved in this way is not backed by the standard.

In [5], Gachery and Thalmann propose another animation technique also controlled directly through FAPs. In this case the specification of Facial Animation Tables (`FATs`), also provided by MPEG-4, is adapted to provide a customizable control of the interpolation of morph targets associated with every FAP. However this approach means that all the extreme deformations of a given face for, at least, the 66 low-level FAPs have to be modelled. An alternative technique that doesn't suffer from this drawback is the one proposed by Noh et al. [14]. It defines a way of translating deformation between different face geometries by defining a mapping of the displacement function over the domain of a flattened triangulation of the MPEG-4 FPs. It still requires the definition of morph targets for the reference face, but then they can be extrapolated to others.

In the works of Kshirsagar et al. [10] and Pasquariello et al. [15], deformation of the geometry representing the face is achieved by means of a weighted interpolation of FP displacements. While these techniques follow more closely the definitions of the standard, their application to the skin surface is inappropriate, since they ignore its connectivity when performing the deformation; what they actually perform is akin to pinching of circular regions, rather than a surface-consistent warp. This results in visually noticeable artifacts.

Recent contributions [13, 4] propose using Radial Basis Functions (`RBFs`) to build a warping function interpolating the displacements experienced at the FPs. They produce better results than the weighted interpolation described before; however imposing constraints on the regions of affection of certain control points, in order to prevent transmission of deformation across discontinuities, proves quite difficult in practice. The use of methods like edge-based metrics or fixed partitioning alters the properties of the RBFs, making the interpolation they perform unstable.

1.2 Our approach to MPEG-4 Facial Animation

We suggest the application of a deformation algorithm based on Planar Bones. It consists in a reformulation of Surface Oriented Free Form Deformations [19] that solves many of the issues that arise in the practical use of this technique for Facial Animation. The resulting method provides controllable locality of deformation as well as a strong surface to surface correspondence with a polygonal mesh used as a control structure.

A diagram representing the process performed by the proposed Facial Animation framework is depicted by Figure 3. Before discussing their use in Facial An-

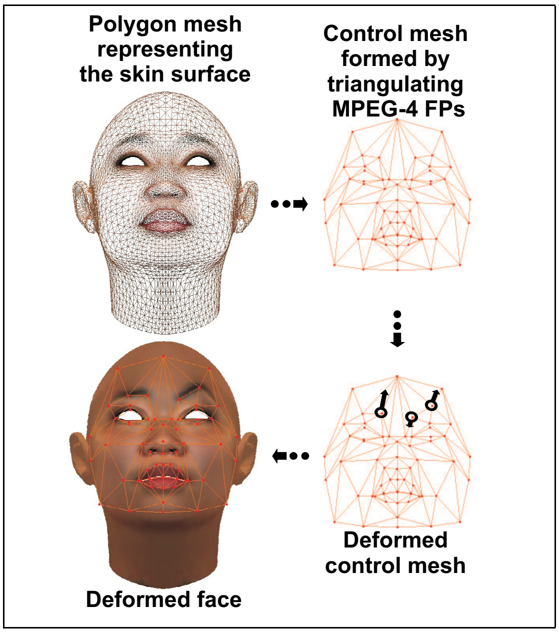

Polygon mesh representing the skin surface
Control mesh formed by triangulating MPEG-4 FPs
Deformed control mesh
Deformed face

Figure 3. Implementation of `MPEG-4` Facial Animation using Planar Bones.

imation, we will introduce Planar Bones as a general warping method.

2 The Planar Bones deformation algorithm

The main purpose of deformation algorithms is to provide a parameterisation of a geometric warp that provides simplicity in deformation control without compromising expressivity in terms of achievable warped configurations. Surface-oriented Free Form Deformations (`SoFFDs`) [19] accomplish this by defining a mesh to mesh mapping between the control surface and the deformation target. Large scale warps are easily performed with a coarse control mesh, while finer ones can be achieved by refining this control structure in precise regions.

SoFFDs have been satisfactorily applied for character animation purposes and implemented in commercial 3D packages. They provide better control than Axial Deformations [11] for skeletal animation [12, 3, 18, 20]. Triangles are used as control elements, instead of segments, and arranged as a wrapping delimiting the extents of the deformed regions. However this method suffers from similar issues to those of its axial counterparts when control elements are stretched or sheared, which is something that inherently happens in Facial Animation.

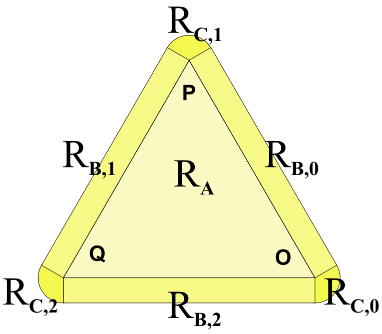


Figure 4. Affection volume regions for Planar Bones.

Correcting these artifacts is the aim of the reformulation of SoFFDs as Planar Bones. First we provide a definition of the original algorithm that is more convenient for the analysis of the problems it presents, and then we rebuild the expression in a way that overcomes them.

2.1 Formulation of Surface-oriented Free Form Deformations

As discussed before, SoFFDs share an identical basis with linear Axial Deformations (more popularly known as *Bones*), but using the plane spanned by a triangular element, instead of a segment, to drive the deformation. We start with a single control element before extending this to combinations.

Consider the triangle defined by vertices O, P, Q, which moves into a configuration spanned by o, p, q. The image of a point S in the deformed geometry will be given by:

$$s = o + (S - O)\left(A^{OPQ}\right)^{-1} A^{opq} \tag{1}$$

where the transformation matrices A^{OPQ} and A^{opq} are defined as follows:

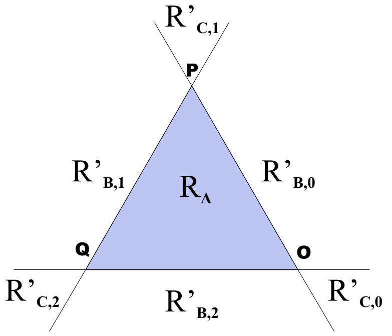
$$A^{\mathrm{LMN}} = \begin{pmatrix} (\mathrm{M} - \mathrm{L}) \\ (\mathrm{N} - \mathrm{L}) \\ n^{\mathrm{LMN}} \end{pmatrix} \tag{2}$$

with the plane normal vector n^{LMN} given by:

$$n^{\mathrm{LMN}} = \frac{(\mathrm{M} - \mathrm{L}) \times (\mathrm{N} - \mathrm{L})}{\|(\mathrm{M} - \mathrm{L}) \times (\mathrm{N} - \mathrm{L})\|} \tag{3}$$

The targeted geometry has to be partitioned in order to achieve locality in the deformation; this is performed by using a function of the distance to the control element to delimit its volume of affection. Computing the distance itself requires particularization for certain regions of this affection volume; the segmentation used is represented in Figure 4.

In order to determine the partition a geometric element S belongs to, a first classification can be made according to the barycentric coordinates (u, v, w) of its projection over the control triangle OPQ:



$$\forall S, (u, v, w) = Bar_{\mathrm{OPQ}}(\textstyle\prod_{\mathrm{OPQ}}(S))$$

$$\begin{cases} (u \geq 0) \wedge (v \geq 0) \wedge (w \leq 1) & \Rightarrow \quad S \in \mathbf{R_A} \\ (u \geq 0) \wedge (v < 0) \wedge (w \leq 1) & \Rightarrow \quad S \in \mathbf{R'_{B,0}} \\ (u \geq 0) \wedge (v \geq 0) \wedge (w > 1) & \Rightarrow \quad S \in \mathbf{R'_{C,1}} \\ (u < 0) \wedge (v \geq 0) \wedge (w \leq 1) & \Rightarrow \quad S \in \mathbf{R'_{B,1}} \\ (v < 0) \wedge (w > 1) & \Rightarrow \quad S \in \mathbf{R'_{C,2}} \\ (u < 0) \wedge (w > 1) & \Rightarrow \quad S \in \mathbf{R'_{B,2}} \\ (u < 0) \wedge (v < 0) & \Rightarrow \quad S \in \mathbf{R'_{C,0}} \end{cases} \tag{4}$$

Candidate regions $\mathbf{R'_{B,i}}$ and $\mathbf{R'_{C,i}}$ are subsequently used to determine the which vertices lie in the apex and edge regions $\mathbf{R_{B,i}}$ and $\mathbf{R_{C,i}}$:

$\forall S \in \mathbf{R'_{B,i}}$

$$\begin{cases} c_i(S) < 0 & \Rightarrow \quad S \in \mathbf{R_{C,i}} \\ c_i(S) \in [0, \|e_i\|] & \Rightarrow \quad S \in \mathbf{R_{B,i}} \\ c_i(S) > \|e_i\| & \Rightarrow \quad S \in \mathbf{R_{C,i+1}} \end{cases} \tag{5}$$

$\forall S \in \mathbf{R'_{C,i}}$

$$\begin{cases} c_{i-1}(S) \leq \|e_{i-1}\| & \Rightarrow \quad S \in \mathbf{R_{B,i-1}} \\ (c_{i-1}(S) > \|e_{i-1}\|) \wedge (c_i(S) < 0) & \Rightarrow \quad S \in \mathbf{R_{C,i}} \\ c_i(S) \geq 0 & \Rightarrow \quad S \in \mathbf{R_{B,i}} \end{cases} \tag{6}$$

where for every triangle LMN:

$$e_j = \{L, M, N\}_{j+1} - \{L, M, N\}_j$$
$$c_j(S) = \frac{(S - \{L, M, N\}_j) \cdot e_j}{\|e_j\|} \tag{7}$$

and all indices must be taken modulus 3. This also applies for the equations in subsection 2.3.

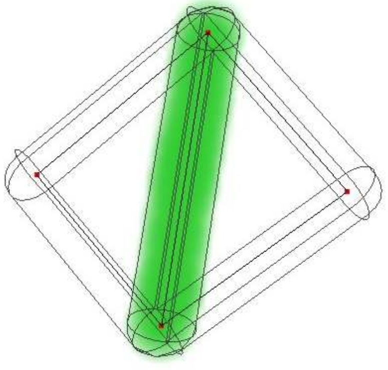


Figure 5. Confluence of the affection volumes of different Planar Bones (shaded).

Once determined the region in Figure 4 that the projection of S belongs to, the distance from the control element to this point is trivially computed from either a plane (R_A), a line $(R_{B,i})$ or a point $(R_{C,i})$.

2.2 Combining the effects of several control elements

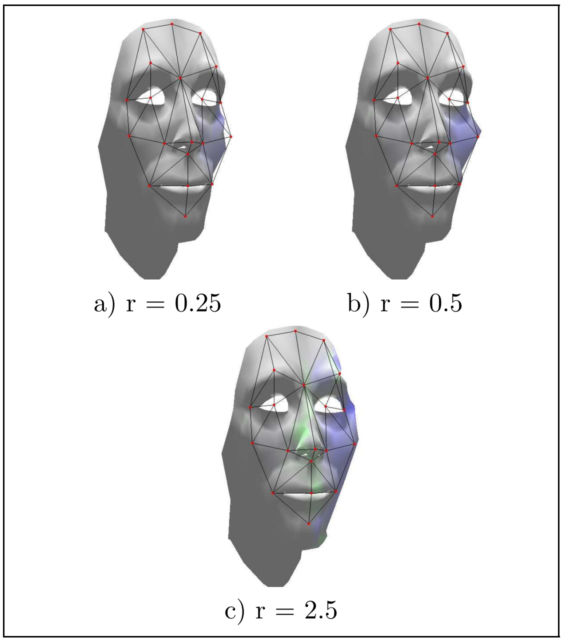
When a triangle mesh of control elements is used to conduct the deformation, individual affection volumes become coincident, as depicted by Figure 5. In order to handle this situation, we can define the deformed image of any vertex S_i as the weighted combination of the contribution of affecting control elements, in the same way that *Bones* do for skeletal animation:

$$s_i = \sum_j w_{ij} s_i^j \tag{8}$$

where s_i^j is the image of S_i according to the j^{th} control element that it is attached to, evaluated using Equation 1.

The weighting w_{ij} is computed as a function of the distance from the vertex to the correspondent control triangle (d_{ij}). Candidate weighting functions must evaluate to 1 when the distance is 0 and vanish when it approaches the radius of the affection volume (r_j), in order to produce continuous deformation. Smoothness in this decay is also desirable, therefore highly differentiable functions (over the interval $[0, r_j)$) are preferred. The function we use is:

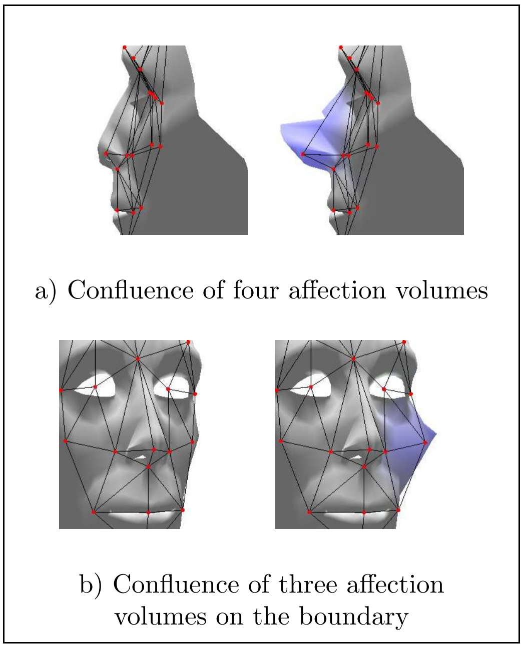
$$w_{ij}^n = \begin{cases} \left(\frac{1}{2}\left(1 + \cos\left(\frac{d_{ij}}{r_j}\pi\right)\right)\right)^n & \text{if } r_j > d_{ij} \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$



a) r = 0.25 b) r = 0.5

c) r = 2.5

Figure 6. Variation of the affection volume radius of SoFFD control elements: a) part of the cheek is not moved; b) correct deformation; c) other regions are also affected.



a) Confluence of four affection volumes

b) Confluence of three affection volumes on the boundary

Figure 7. Distortions experienced in the application of SoFFDs: the tip of the nose (a) and extreme of the cheek (b) are overwarped.

The interpretation of the relation between the control structure of the warping algorithm and the geometry it is applied to is significantly different from that given in skeletal animation. *Bones* are used in contexts where the control segments lie along the axes of the limbs, which are generally cylindrically shaped. This makes the definition of the radius of the affection volume quite intuitive, i. e. that of the cylinder that encompasses the deformed section of the limb. In the case of SoFFDs the control structure is going to be defined in a way that approximates the deformed surface. Therefore determining the radius of the volume of influence of every control element is not trivial for surfaces that alternate convex regions with concavities, such as the skin of the human face. In these cases a careless selection of this parameter can easily result in part of the geometry being left out of the extents of the deformation domain, and not affected by the changes in the control structure, as shown in Figure 6.a. On the other hand overextending this parameter could undermine the locality of the warp, making its control rather difficult. This is illustrated by Figure 6.c.

SoFFDs are extremely dependent on the definition of Euclidean distance from a targeted vertex to the corresponding control element. However, this magnitude is not guaranteed to be preserved through the deformation process, since the involved transformations are anisometric, potentially involving scaling and shearing. This means that the extents of the affection volumes are also scaled and sheared, breaking the correspondence of their contained geometry with the control surface. This fact is especially noticeable at joins between several control elements, where the contribution of their associated distortions becomes constructive, as shown in Figure 7. The Planar Bones deformation algorithm redefines the expression of the SoFFD warp, particularizing it for each of the regions of the affection volume and thus overcoming the described inconsistency.

2.3 The Planar Bones reformulation

As can be deduced from the expressions given in subsection 2.1, only the vertices contained in extremal regions of the affection volume may suffer from variation in their distance to the corresponding triangular element through the deformation. $\mathbf{R_A}$ remains impervious to this effect since its warp is consistent with that of the control triangle.

For each of the regions $\mathbf{R_{B,i}}$ and $\mathbf{R_{C,i}}$, we define reference frames B_i and C_i as follows:

$$B_i^{\text{LMN}} = \begin{pmatrix} e_i \\ \frac{e_i \times n^{\text{LMN}}}{\|e_i \times n^{\text{LMN}}\|} \\ n^{\text{LMN}} \end{pmatrix} \tag{10}$$

$$C_i^{\text{LMN}} = \begin{pmatrix} \frac{e_i \times n^{\text{LMN}}}{\|e_i \times n^{\text{LMN}}\|} \\ \frac{e_{i-1} \times n^{\text{LMN}}}{\|e_{i-1} \times n^{\text{LMN}}\|} \\ n^{\text{LMN}} \end{pmatrix} \tag{11}$$

where n^{LMN} is the normal of the control triangle LMN as defined in (2).

For S belonging to regions $R_{B,i}$ Equation 1 is reformulated using Equation 10:

$\forall S \in \mathbf{R_{B,i}}$,

$$s = \{o,p,q\}_i + (S - \{O,P,Q\}_i)\left(B_i^{OPQ}\right)^{-1} B_i^{opq} \tag{12}$$

It follows from the definition of B_i^{LMN} that, using the previous expression, this region of the control volume is not subject to any anisometric deformation other than the scaling along the associated triangle edge. Since the distance is computed from this segment, it remains constant through the deformation.

A similar approach is not applicable for regions $R_{C,i}$, since the transformation matrices C_i^{LMN} may exhibit shearing, resulting in the whole deformation varying the distance in a non-linear way that can not be corrected by just modifying the transformation matrix itself. A more complex treatment is therefore required.

The deformation for the apices is reformulated by intercalating a non-linear scaling term between the two transformation matrices:

$\forall S \in \mathbf{R_i}$,

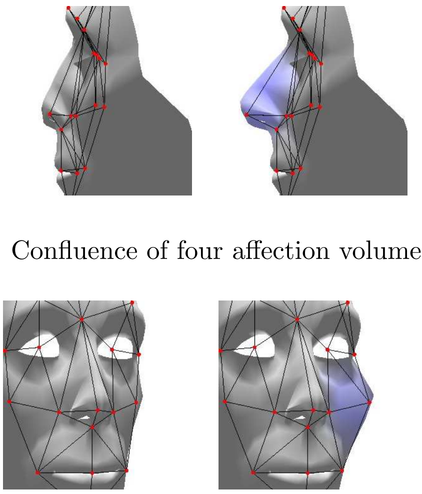
$s = \{o,p,q\}_i +$

$$(S - \{O,P,Q\}_i)\left(C_i^{OPQ}\right)^{-1} Diag\begin{pmatrix} \frac{D_i(S)}{d_i(S)} \\ \frac{D_i(S)}{d_i(S)} \\ 1 \end{pmatrix} C_i^{opq} \tag{13}$$

For a column vector v of dimension N, $Diag(v)$ is defined as the $N \times N$ diagonal matrix with elements v_j. The coefficients $D_i(S)$ and $d_i(S)$ of (13) are formulated as follows:

$$D_i(S) = \|\prod_{\text{OPQ}}(S) - \{O,P,Q\}_i\|$$

$$d_i(S) = \|\left(\prod_{\text{OPQ}}(S) - \{O,P,Q\}_i\right)\left(C_i^{OPQ}\right)^{-1} C_i^{opq}\| \tag{14}$$



a) Confluence of four affection volumes

b) Confluence of three affection volumes on the boundary

Figure 8. Application of the given reformulations to the previous examples, exhibiting no distortions.

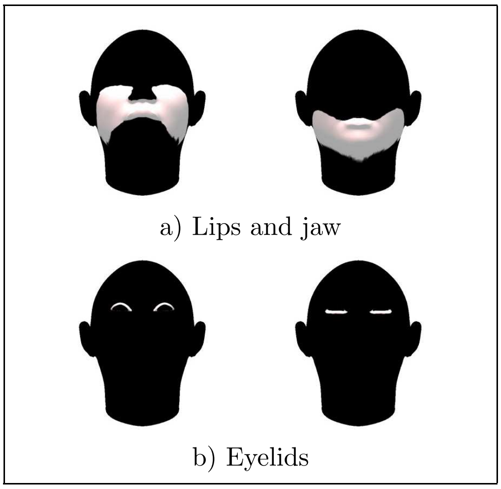
The application of this reformulation corrects the SoFFD issues described in the previous subsection, as demonstrated by Figure 8. This comes at the cost of reducing the degree of continuity in the deformation down to C_0, because of this particular case discrimination; however the blending function used in overlapping affection values attenuates this so that surfaces appear smooth.

3 Application of Planar Bones to MPEG-4 Facial Animation

The MPEG-4 FP set provides a suitable control input for the application of Planar Bones to the deformation of a highly detailed face geometry; it only requires to be triangulated in order to build a proper control mesh. Figure 3 shows the triangulation we use.

Another requirement of Planar Bones is to determine the affection radius for each control element. In order to achieve this, we find the closest feature point to a given triangle (control element) and use a fraction (user defined parameter) of this distance. This is done for each control element such that every vertex of the targeted face geometry is included in at least one affection volume. Locality of deformation is also preserved by this procedure.

Even with these considerations, the Planar Bones formulation doesn't take into account discontinuities along the targeted surface, and in Facial Animation terms that can result in an undesired transmission of



a) Lips and jaw

b) Eyelids

Figure 9. Texture maps used to mask the affection regions.

deformation across the mouth and eye openings. Some MPEG-4 animation systems use static partitioning of the geometry [15] or edge traversals [10, 13] to treat this problem. However these techniques prove to be extremely dependent on a proprietary face representation, and difficult to adapt to a new one. Our implementation tackles this problem by means of masking the affection regions with texture maps, which can be intuitively provided as part of the modelling process. The textures used in our samples are shown in Figure 9.

The control paradigm constructed in this way not only permits automated input coming from an MPEG-4 stream, but also may be suitable for performing artistic gesture editing and animation. Using the approach outlined in Figure 3, Figure 10 shows gestures produced by displacing the MPEG-4 FPs.

4 Conclusions

Planar Bones provide a very convenient deformation paradigm for Facial Animation and, as demonstrated in this paper, the MPEG-4 FA standard can be adapted in order to provide the necessary control input. Our system also attempts to define a simple method for including new face definitions; its only requirements, besides the placement of the MPEG-4 FPs, are to properly estimate the scale factor for the whole affection volume, and to define texture maps that mark the regions deformed by those control elements lying on boundaries of the eye and mouth openings.

One of the principal virtues of the deformation algorithm used is that it benefits from the same efficiency of typical character skinning approaches, such as *Bones*, and it can also be implemented with a programmable hardware transformation pipeline, such as those present in current graphics cards. This permits the integration of highly detailed face representation, like the one used in the samples provided (∼14k polygons), in interactive content.

Reflecting more precise characteristics of the tissue behavior, such as appearance of bulges and creases, is not directly possible since the deformation is conducted without any physical consideration. However they could be extrapolated from a deformation analysis after the application of the warp. Further research is currently being conducted in this direction.

5 Acknowledgements

The authors would like to acknowledge The Pedro Barrié de la Maza Foundation for funding this research. Manuel Sánchez would also like to acknowledge Mark Eastlick, James Edge, Harini Kulatunga and Michael Meredith for their support in producing this paper.

References

[1] J. Ahlberg. Candide-3 - an updated parameterised face (`http://citeseer.nj.nec.com/ahlberg01candide.html`), 2001.

[2] P. Bèzier. General distortion of an ensemble of biparametric surfaces. *Computer-Aided Design*, 10:117–120, 1978.

[3] J. Chadwick, D. R. Haumann, and R. E. Parent. Layered construction for deformable animated characters. *SIGGRAPH'89 Conference Proceedings*, pages 234–243, 1989.

[4] D. Fidaleo, J. Noh, T. Kim, R. Enciso, and U. Neumann. Classification and volume morphing for performance-driven facial animation. *International Workshop on Digital and Computational Video (DCV'99)*, December 1999.

[5] S. Gachery and N. Magnenat Thalmann. Designing mpeg-4 facial animation tables for web applications. *Multimedia Modeling 2001*, pages 39–59, 2001.

[6] Iso/iec 14496-2 - information technology - coding of audio-visual objects - part 2: Visual. second edition, 2001.

[7] Iso/iec 14496-5 - information technology - coding of audio-visual objects - part 2: Reference software. second edition, 2001.

[8] P. Kalra, A. Mangili, N. Magnenat Thalmann, and D. Thalmann. Simulation of facial muscle actions based on rational free form deformations. *Proceedings of Eurographics'92*, pages 59–69.

[9] P. Kalra, A. Mangili, N. Magnenat Thalmann, and D. Thalmann. 3d interactive free form deformations for facial expressions. *Proceedings of Compugraphics'91*, 1992.

[10] S. Kshirsagar, S. Gachery, and N. Magnenat-Thalmann. Feature point based mesh deformation applied to mpeg-4 facial animation. *Proceedings Deform'2000*, pages 23–34, 2000.

[11] F. Lazarus, S. Coquillart, and P. Jancène. Axial deformations: an intuitive deformation technique. *Computer-Aided Design*, 26:607–613, 1994.

[12] N. Magnenat Thalmann, R. Laperriere, and D. Thalmann. Joint-dependent local deformations for hand animation and object grasping. *Proceedings Graphics Interface '88*, pages 26–33.

[13] J. Noh, D. Fidaleo, and U. Neumann. Animated deformations with radial basis functions. *ACM Symposium on Virtual Reality Software and Technology (VRST 2000)*, pages 166–174, October 2000.

[14] J. Noh and U. Neumann. Expression cloning. *SIGGRAPH 2001 Conference Proceedings*, pages 277–288, 2001.

[15] S. Pasquariello and C. Pelachaud. Greta: A simple facial animation engine. preprint (`http://www.dis.uniroma1.it/~pelachau/wsc01.pdf`), 2001.

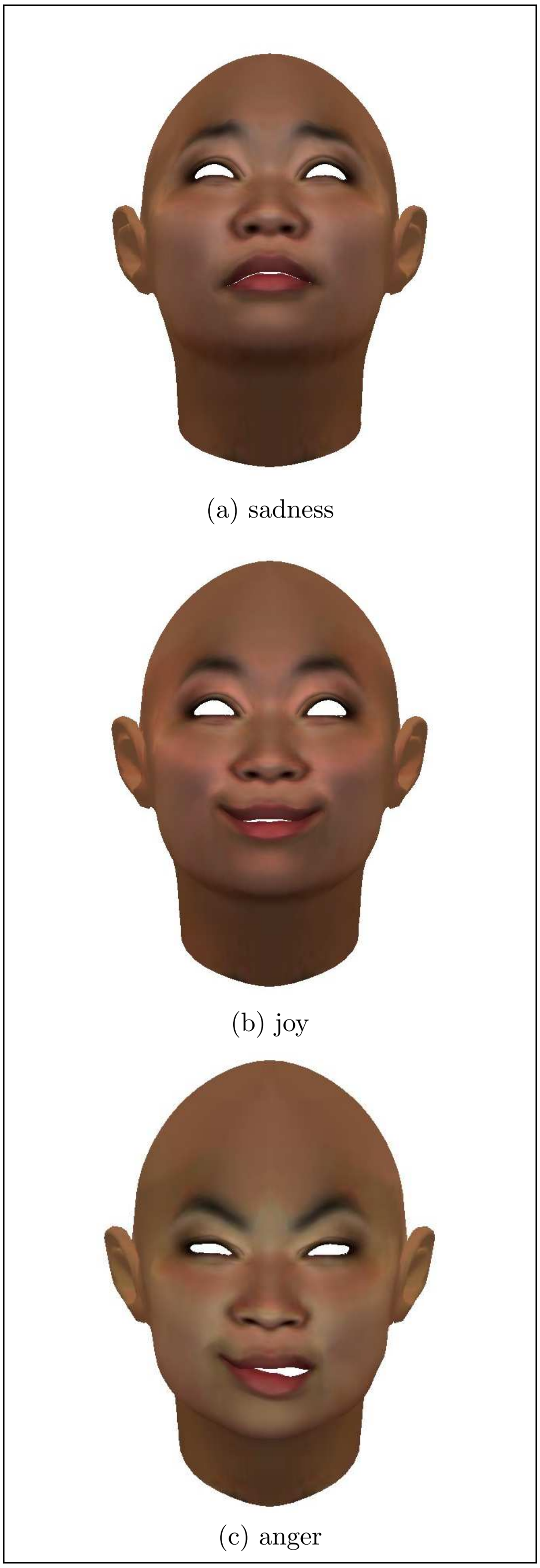
[16] M. Rydfalk. Candide, a paremeterized face. report no. lith-isy-i-866 (`http://www.icg.isy.liu.se/candide/main.html`), 1987.

[17] T. W. Sederberg and S. R. Parry. Free-form deformation of solid geometric models. *SIGGRAPH'86 Conference Proceedings*, 20:151–160, 1996.

[18] H. Seo, F. Cordier, L. Philippon, and N. Magnenat Thalmann. Interactive modelling of mpeg-4 deformable human body models. *Proceedings of Deform'2000*, pages 120–131.

[19] K. Singh and E. Kokkevis. Skinning characters using surface-oriented free-form deformations. *Graphics Interface 2000*, pages 35–42, 2000.

[20] J. Weber. Run-time skin deformation. *Proc. Game Developers Conference 2000 (http://www.intel.com/ial/3dsoftware/doc.htm)*, 2000.



(a) sadness

(b) joy

(c) anger

Figure 10. Sample of facial gestures generated with Planar Bones using the MPEG-4 parameterisation.