International Conference on Image Processing (ICIP 97), vol. 2, pp. 418-421, Santa Barbara, October 1997.

Model-Based Estimation of Facial Expression Parameters from Image Sequences

Peter Eisert and Bernd Girod
Telecommunications Institute
University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
{eisert,girod}@nt.e-technik.uni-erlangen.de

Abstract

In this paper we present a model-based algorithm for the estimation of 3D motion and the analysis of facial expressions of a speaking person. A set of facial animation parameters based on the MPEG-4 standard is determined from two successive video frames using a hierarchical optical flow based method. The motion in the image plane is constrained by a 3D triangular B-spline model that defines shape, texture and facial expressions of an individual person. The computational requirement for this solution is low due to the linear structure of the algorithm.

1 Introduction

Model-based coding is a promising approach for very low bit-rate video compression. Motion parameters of objects are estimated from video frames using three-dimensional models of the objects. These models describe the shape and texture of the objects. At the decoder, the video sequence is synthesized by rendering the models at the estimated positions.

The head of a speaking person cannot be modeled as a rigid body. Local deformations due to facial expressions must be taken into consideration when analyzing the 3D facial motion [1, 2]. We assume that facial expressions can be represented by a linear combination of small elementary local movements. These movements are described by facial animation parameters (FAPs). Examples of face parameterization are the Facial Action Coding System [3] and the system of the MPEG-4 SNHC group [4] that is used in this work.

In our coder all these parameters are estimated simultaneously using a hierarchical optical flow based method. The optical flow constraint is combined with the parameterized 3D motion equations for each object point. The determination of the motion as a function of the FAPs is simplified due to the use of triangular B-splines for the head surface construction. In this way we only have to model the motion of a small number of control points.

2 Camera Model

The three-dimensional scene used for parameter estimation and rendering of the synthetic images consists of a camera model and a head model. For the camera model we use perspective projection where the 3D coordinates of an object point $[x\ y\ z]^T$ are projected into the image plane according to

$$\begin{aligned} X_p &= X_0 - f_x \frac{x}{z} \\ Y_p &= Y_0 - f_y \frac{y}{z}. \end{aligned} \tag{1}$$

Here, f_x and f_y denote the focal length multiplied by scaling factors in x- and y-direction, respectively. These scaling factors transform the image coordinates into pixel coordinates X_p and Y_p. In addition, they allow the use of non-square pixel geometries. The two parameters X_0 and Y_0 describe the image center and its translation from the optical axis due to inaccurate placement of the CCD-sensor in the camera. For simplicity, normalized pixel coordinates X_n and Y_n are introduced

$$X_n = \frac{X_p - X_0}{f_x}\ ,\ Y_n = \frac{Y_p - Y_0}{f_y}. \tag{2}$$

3 Head Model

The estimated motion parameters are constrained by a 3D model of the speaking person specifying 3D shape and facial expressions. The shape of the generic model is adapted to an individual person using a 3D laser scan. Like other well-known facial models [5, 6], our head model also consists of a number of triangles onto which texture is mapped to obtain a photorealistic appearance [7]. The shape of the surface is, however, modeled by second order triangular B-splines [8] that allow to locally control the shape with a small

number of control points $\mathbf{c}_i$. The coordinates of the vertices $\mathbf{v}_j$ in the mesh is then defined by the positions of the control points $\mathbf{c}_i$ according to

$$\mathbf{v}_j = \sum_{i \in I} N_{ji} \mathbf{c}_i, \text{ such that } \sum_{i \in I} N_{ji} = 1, \qquad (3)$$

with N_{ji} being the precalculated basis functions.

B-splines are well-suited for the modeling of facial skin [9] and constrain the motion of neighboring vertices which simplifies modeling of facial expressions. For the parameterization of facial expressions the proposal of the MPEG-4 SNHC group [4] was chosen. By changing the facial animation parameters, the control points of the spline surface are moved which results in a new shape for the 3D model. Two examples of expressions rendered with this head model can be seen in Figure 1.

Figure 1: Illustration of different synthesized facial expressions.

4 Facial Parameter Estimation

The motion estimation algorithm presented in this paper estimates the facial animation parameters from two successive frames. To avoid error accumulation in the long-term parameter estimation, a feedback loop is used [10, 11] as depicted in Figure 2. The model

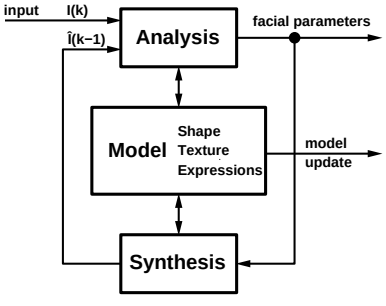

Figure 2: Feedback structure of the coder.

of the head is moved according to the parameters estimated from frame I(k) and I(k-1) and a synthetic image is generated by rendering the model with modified shape and position. The estimation is then performed between the actual camera frame and the synthetic image of the previous frame which assures that no large misalignment of the model occurs. The feedback loop is also used in our hierarchical approach to compensate the facial motion before starting the next iteration on a higher resolution level. To reduce errors that are caused by linearizations in the algorithm, we first estimate the parameters with subsampled images. This rough estimate is then used to compensate the motion in the synthetic image leading to a new frame that is closer to I(k). These steps are repeated on higher resolution images resulting in more and more accurate facial parameters.

For the motion estimation the whole image is used by setting up the optical flow constraint equation

$$I_X \cdot u + I_Y \cdot v + I_t = 0 \qquad (4)$$

where $[I_X \; I_Y]$ is the gradient of the intensity at point $[X_p \; Y_p]$, u and v the velocity in x- and y-direction and I_t the intensity gradient in temporal direction.

Instead of computing the optical flow field by using additional smoothness constraints and then extracting the motion parameter set from this flow field, we estimate the facial animation parameters from (4) together with the 3D motion equations of the head's object points [1]. For a rigid body motion the motion equation can be easily written in terms of a translation and a rotation. When allowing the body to be flexible, this is no longer possible and the motion equation must be set up at each object point independently. The trajectory of an object point is, however, not independent of its neighbors but is constrained by the head model that describes motion of surface points as a function of facial animation parameters.

Local deformations caused by facial expressions are taken into consideration by changing the facial animation parameters. These FAPs determine the position of all object points in the synthetic image by applying different transformations as shown in Figure 3. First

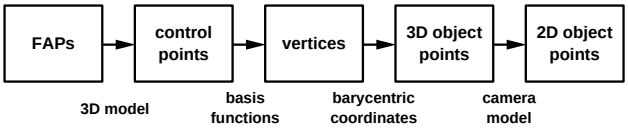

Figure 3: Transformation from FAPs to image points.

the control points of the surface are moved according to the given FAPs. Using the basis functions of the splines the position of the vertices can be calculated

from the control points. Three vertices form a triangle and the 3D motion of all object points inside this triangle specified by their barycentric coordinates is determined. The 2D image point is finally obtained by projecting the 3D point into the image plane. These transformations, that are all linear except for the projection, must be incorporated into our algorithm to obtain a second constraint that can be used together with (4).

The new control point position $\mathbf{c}'$ can be determined from the position $\mathbf{c}$ in the previous frame by

$$\mathbf{c}' = \mathbf{c} + \sum_k a_k \mathbf{d}_k \tag{5}$$

where a_k are the changes of the facial animation parameters between the two frames that are estimated by the algorithm and $\mathbf{d}_k$ the 3D direction vectors of the corresponding movement.

Strictly speaking, equation (5) is just valid for translations. If a number of control points are rotated around given axes by some action units, the description for the motion of control points becomes much more complicated due to the combination of rotation (defined by rotation matrices) and translation. The order of these operations can no longer be changed and the use of matrix multiplication results in a set of equations that is nonlinear in the parameters that have to be estimated. However, we can use the linear description (5) also for rotation, if we assume that the rotation angles between two successive frames are small. Then, the trajectory of an object point k that is rotating around the center O can be approximated by its tangent $\mathbf{d}_k$ as shown in Figure 4. This tangent differs for all object points, but we have to set up equation (5) for all points anyhow because of local deformations in the surface. For a rotation with the

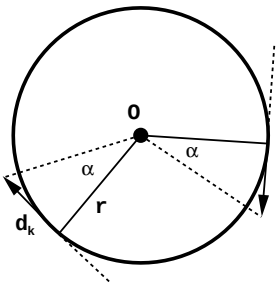

Figure 4: Approximation of rotations.

angle α

$$\alpha = a_k \cdot s_k, \tag{6}$$

that is defined by the facial animation parameter a_k and the corresponding scaling factor s_k the length of the translation vector $\mathbf{d}_k$ can be calculated by

$$|\mathbf{d_k}| = a_k \cdot r \cdot s_k. \tag{7}$$

Here, r is the distance between the object point and the given rotation axis. With this assumption (the direction of $\mathbf{d}_k$ is specified by the direction of the tangent), equation (5) can also be used for rotation leading to a simple linear description for the FAPs that can be estimated very efficiently. The small error caused by the approximation vanishes after some iterations in the feedback structure shown in Figure 2.

Having modeled the shift in control points, the motion of the vertices of the triangular mesh can be determined using (3) and the local motion of an object point $\mathbf{x}$ is calculated from that using

$$\mathbf{x} = \sum_{m=0}^{2} \lambda_m \mathbf{v}_m = \sum_{j \in J} (\sum_{m=0}^{2} \lambda_m N_{mj}) \mathbf{c}_j, \tag{8}$$

where λ_m are the barycentric coordinates of the object point in the triangle that encloses that point. The motion equation for a surface point can be represented as

$$\mathbf{x}' = \mathbf{x} + \sum_k a_k \mathbf{t}_k = \mathbf{x} + T \cdot \mathbf{a}, \tag{9}$$

where $\mathbf{t}_k$'s are the new direction vectors to the corresponding facial animation parameter calculated from $\mathbf{d}_k$ by applying the linear transforms (3) and (8). T combines all the vectors in a single matrix and $\mathbf{a}$ is the vector of all FAPs. Let $\mathbf{t}_x$, $\mathbf{t}_y$ and $\mathbf{t}_z$ be the row vectors of this matrix. The components of equation (9) are given by:

$$x' = x(1 + \frac{1}{x}\mathbf{t}_x \cdot \mathbf{a}) \tag{10}$$

$$y' = y(1 + \frac{1}{y}\mathbf{t}_y \cdot \mathbf{a}) \tag{11}$$

$$z' = z(1 + \frac{1}{z}\mathbf{t}_z \cdot \mathbf{a}). \tag{12}$$

Dividing (10) and (11) by (12), inserting the camera model (1) and using a first order approximation leads to

$$\begin{aligned} X' - X &\approx -\frac{f_x}{z}(\mathbf{t}_x + X_n \mathbf{t}_z)\mathbf{a} \\ Y' - Y &\approx -\frac{f_y}{z}(\mathbf{t}_y + Y_n \mathbf{t}_z)\mathbf{a}. \end{aligned} \tag{13}$$

Together with (4) a linear equation at each pixel position can be set up

$$\frac{1}{z}\left[I_X f_x \mathbf{t}_x + I_Y f_y \mathbf{t}_y + (I_X f_x X_n + I_Y f_y Y_n)\mathbf{t}_z\right]\mathbf{a} = I_t \tag{14}$$

with z being the depth information coming from the model. We obtain an overdetermined system that can be solved in a least-squares sense with low computational complexity. The size of the system depends directly on the number of implemented FAPs.

5 Experimental Results

To analyze the accuracy of the proposed method a number of synthetic images with well defined facial animation parameters are rendered. For every pair of images the estimated facial parameters are compared with the correct values. The algorithm is run with three different levels of resolution with a final resolution of 352 x 288 pixels (CIF). The viewing angle of the camera is about 25^o. The absolute error of the estimated facial parameters averaged over several frames is 0.06% of the maximum value leading to an average PSNR in the facial area of 80 dB. This shows that the original and the estimated images are basically identical.

In a second experiment a video sequence of a talking person is recorded in CIF resolution. The facial animation parameters are estimated for all 230 frames and the corresponding synthetic sequence is rendered. Two frames of the camera sequence and the synthesized frames from the estimated parameters are shown in Figure 6. The PSNR between original and synthetic images is measured in the facial area leading to an average PSNR of 32.0 dB coded with 0.58 kbit/s at 12.5 Hz. The changes of PSNR during time are depicted in Figure 5.

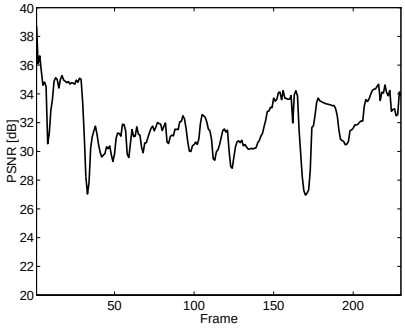

Figure 5: PSNR of a coded talking head sequence.

6 Conclusions

In this paper we have presented an algorithm for the estimation of facial animation parameters from monocular video sequences. The approach is model-based and uses a 3D model specifying shape and texture of a head. This model constrains the 3D motion of the object points and together with the optical flow constraint the parameters are estimated by our linear hierarchical approach. Experimental results show that the estimation works well for both synthetic and real video sequences.

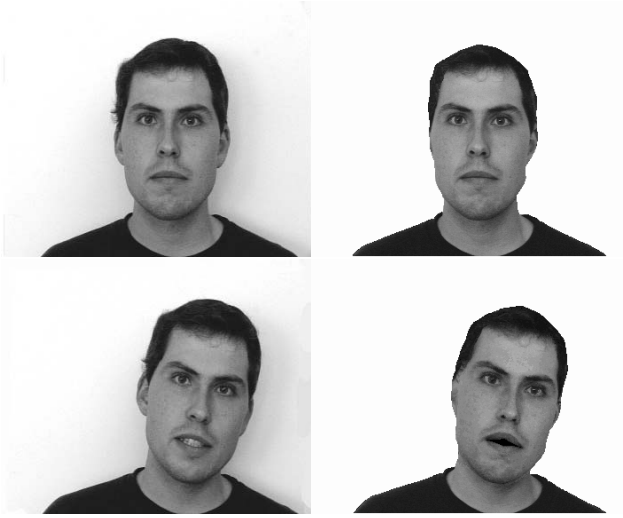

Figure 6: two camera frames of a talking head sequence (left) and synthetic images (right).

References

[1] H. Li, P. Roivainen, and R. Forchheimer, "3-D motion estimation in model-based facial image coding", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 545–555, June 1993.

[2] J. Ostermann, *Analyse-Synthese-Codierung basierend auf dem Modell bewegter, dreidimensionaler Objekte*, VDI Reihe 10, Nr. 391, VDI-Verlag, 1995.

[3] P. Ekman and W. V. Friesen, *Facial Action Coding System*, Consulting Psychologists Press, Inc., Palo Alto, 1978.

[4] MPEG-4, *SNHC Verification Model 4.0, Document N1666*, Apr. 1997.

[5] F. I. Parke, "Parameterized models for facial animation", *IEEE Computer Graphics and Applications*, vol. 2, no. 9, pp. 61–68, Nov. 1982.

[6] M. Rydfalk, *CANDIDE: A Parameterized Face*, PhD thesis, Linköping University, 1978, LiTH-ISY-I-0866.

[7] P. Eisert and B. Girod, "Facial expression analysis for model-based coding of video sequences", *Proc. Picture Coding Symposium (PCS)*, pp. 33–38, Sep. 1997.

[8] G. Greiner and H. P. Seidel, "Modeling with triangular B-splines", *ACM/IEEE Solid Modeling Symposium*, pp. 211–220, 1993.

[9] M. Hoch, G. Fleischmann, and B. Girod, "Modeling and animation of facial expressions based on B-splines", *Visual Computer*, vol. 11, pp. 87–95, 1994.

[10] R. Koch, "Dynamic 3-D scene analysis through synthesis feedback control", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 556–568, June 1993.

[11] P. Eisert and B. Girod, "Model-based 3D motion estimation with illumination compensation", in *Proc. International Conference on Image Processing and its Applications*, Dublin, Ireland, Jul. 1997, vol. 1, pp. 194–198.