ROBUST RIGID HEAD MOTION ESTIMATION BASED ON DIFFERENTIAL EVOLUTION

Axel Weissenfeld, Onay Urfalioglu, Kang Liu and Joern Ostermann

Institut für Informationsverarbeitung
University of Hannover
Appelstr. 9A, 30167 Hannover, Germany
aweissen@tnt.uni-hannover.de

ABSTRACT

In this paper we present a system to robustly estimate the 3D position of a human head. Before the face model is positioned in the initial frame, it is adapted to the 3D scan of the tracked human head. Head tracking is achieved by minimizing a robust cost function with a stochastic optimization algorithm called Differential Evolution. This approach enables the estimation of large motions between consecutive frames. Furthermore, the algorithm can even handle a large number of outliers e.g. caused by occlusion and still estimate the precise position.

1. INTRODUCTION

A head tracking system estimates the rigid motion of the human face throughout an image sequence. Head tracking systems are important for many applications in computer vision like expression analysis, face identification, and 3D facial animation systems. Head motion can be used to recognize simple gestures, like head shaking or nodding, or for capturing a person's focus of attention, providing a natural clue for human machine interfaces.

Existing approaches can be divided in motion-based and model-based systems. In the first approach, distinct facial features, such as eye corners or nostrils, are tracked throughout the image sequence [1]. The displacements between corresponding feature points can be estimated using optical flow or block-based motion estimation methods. In this way, a 2D motion field is estimated in order to calculate the motion of the object model. The object model is only used to transform 2D motion vectors into object model motion vectors. One problem with methods based on this approach, as shown by Li et al. [2], is the accumulation of motion estimation errors.

A model-based tracker stores texture information of the object and tries to adapt the object model's position to fit the new frame. Therefore, the motion estimation is dependent on the texture information of the initial and current frame and on the object model. Model-based motion estimation can be accomplished by optical flow [3] or image registration in texture space [4]. The latter can be accomplished by a stochastic optimization algorithm. Many variations of motion estimation algorithms have been proposed in the literature. Differences can be noticed in the boundary conditions, like the use of calibrated or uncalibrated image sequences or the used motion model.

In this paper we estimate the rigid motion of fast moving human faces and of a human face in temporally downsampled sequences, sequences in which a larger number of consecutive images is lost, e.g. during transmission. Therefore head motion may appear jerky instead of continuous. Hence, optical flow presuming a linear signal model is not suitable for tracking. However, image registration in texture space with a stochastic optimization algorithm is suitable. While being very robust, stochastic optimization algorithms require a higher computational effort than optical flow.

In [5] a stochastic algorithm based on the particle filtering approach is proposed for estimating the 3D pose of the head. This algorithm utilizes the Condensation algorithm [6], which assumes smooth state transitions related to consecutive frames, in order to effectively estimate motion. However, the head motion in the sequences we are analyzing is not expected to be continuous. This would lead to an enlargement of the sampling intervals, defining the bounds of the probability density functions and therefore disable a reasonable realization of the Condensation algorithm. To overcome this problem, we replace the Condensation algorithm with a different stochastic optimization algorithm called Differential Evolution (DE), which was initially proposed in [7]. The DE algorithm performs a global search in parameter space leading to the global minimum of a cost function without the knowledge of the former motion trajectory. Being a very robust algorithm, DE enables the estimation of large motions between consecutive frames. Furthermore, a robust cost function [8] is introduced which makes the optimization invariant against outliers that are generally caused by occlusions or noise.

In the remainder of this paper, we describe the motion estimation (Section 2) and Differential Evolution algorithm

This work is supported by the EC within FP6 under Grant 511568 with the acronym 3DTV.

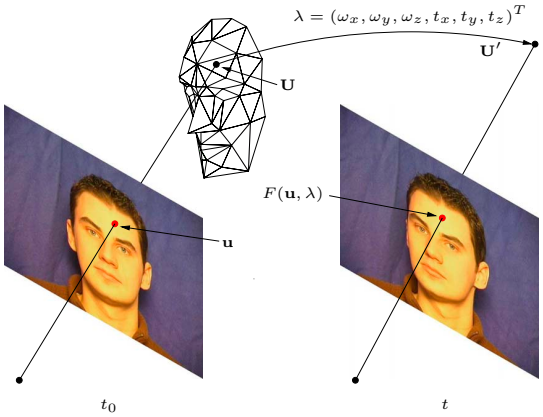

Fig. 1. Motion of a 2D feature point from $F(\mathbf{u}, \mathbf{0})$ in the reference image $I(t_0)$ to $F(\mathbf{u}, \lambda)$ in image $I(t)$, while the corresponding 3D feature point moves from $\mathbf{U}$ to $\mathbf{U}'$.

(Section 3). In Section 4 the robust cost function is described and in Section 5 experimental results of the motion estimation algorithm are presented.

2. MOTION ESTIMATION

Let $I(\mathbf{u}, t)$ be the brightness at the location $\mathbf{u} = (x, y)^T$ in the image I recorded at time t. The initial frame to which the rigid face model is adapted is denoted as $I(t_0)$ and referred to as reference image. The area in the reference image marked by the face model is called the reference template and it is used for the motion estimation. In this area, a number of feature points containing the texture information are defined by $\mathbf{u} \in \Omega$. These points must have distinct visual characteristics such as a high gradient. We use the Harris detector for feature point detection. The feature points are tracked throughout the image sequence. The 3D points corresponding to $\mathbf{u}$ are denoted as $\mathbf{U} = (X, Y, Z)^T$. The rigid motion of 3D points throughout the image sequence is described by a parametric motion model defined as $F(\mathbf{u}, \lambda)$, parameterized by $\lambda = (w_x, w_y, w_z, t_x, t_y, t_z)$ with $F(\mathbf{u}, \mathbf{0}) = \mathbf{u}$.

The problem of motion estimation of a rigid face model can be stated as (Fig. 1): In an image $I(t)$ a 3D point $\mathbf{U}$ is moved from its original position defined by the reference template to a new position $\mathbf{U}'$. Similarly, the point $\mathbf{u}$ on the camera target with the luminance value $I(\mathbf{u}, t_0)$ in the reference template, is moved from $F(\mathbf{u}, \mathbf{0})$ to the position $F(\mathbf{u}, \lambda)$ in image $I(t)$. Assuming diffuse illumination and diffuse reflecting surfaces,

$$I(\mathbf{u}, t_0) = I(F(\mathbf{u}, \lambda), t) \qquad \text{for all } \mathbf{u} \in \Omega \tag{1}$$

holds. For motion estimation we minimize

$$C(\lambda) = \sum_{\mathbf{u} \in \Omega} [I(F(\mathbf{u}, \lambda), t) - I(\mathbf{u}, t_0)]^2 \tag{2}$$

Utilizing a population of solution candidates, the DE-algorithm calculates the cost C(λ) for each candidate and provides the best solution after the convergence of the population.

2.1. Parametric motion model

The parametric motion model $\mathbf{F}$ describes the motion of a 2D feature point $\mathbf{u}$ from $F(\mathbf{u}, \mathbf{0})$ to $F(\mathbf{u}, \lambda)$ by first moving $\mathbf{U}$ to $\mathbf{U}'$ and then projecting the 3D point onto the camera target (Fig. 1). Motion in 3D consists of rotation $\mathbf{R}$ and translation $\mathbf{T}$ with

$$\mathbf{U}' = \mathbf{R}\mathbf{U} + \mathbf{T} \tag{3}$$

2.2. Face model

The geometric shape of the subject's head is approximated by a three-dimensional face model, in order to be able to estimate spatial movements. Here we use the standard face model Candide [9]. The mesh of the face model is defined by its 3D vertices and connectivity describing the head's surface. Initially the Candide mask is precisely adapted to a 3D scan of the human head [10].

3. DIFFERENTIAL EVOLUTION OPTIMIZATION

We use the evolutionary optimizer called Differential Evolution (DE) [7, 11], which is an efficient global optimization technique for continuous problem spaces used in many applications. The optimization is based on a population of $n = 1, ..., N$ solution candidates $s_{n,i} \triangleq \lambda$ at iteration i where each candidate has a position in the 6-dimensional search space. The population improves by generating new positions iteratively for each candidate. The new positions for the iteration step $i + 1$ are determined by

$$d_{n,i+1} = s_{k,i} + F \cdot (s_{l,i} - s_{m,i}) \tag{4}$$

$$s_{n,i+1} = C\left(s_{n,i},\ d_{n,i+1}\right), \tag{5}$$

where k, l, m are random integers from interval $[1, N]$, F is a constant weighting scalar, $d_{n,i+1}$ a displaced $s_{k,i}$ by a weighted difference vector and $C()$ is a crossover operator. The crossover operator intermixes the coordinates of $s_{n,i}$ and $d_{n,i+1}$. It is parameterized by a probability value to decide whether to take the coordinate from $s_{n,i}$ or $d_{n,i+1}$. For this case the resulting $s_{n,i+1}$ owns a lower cost and it replaces $s_{n,i}$, or otherwise it is discarded.

DE includes an adaptive range scaling for the generation of solution candidates through the difference term in (4). This enables global search in the case where the solution candidate vectors are spread in the search space and so the mean

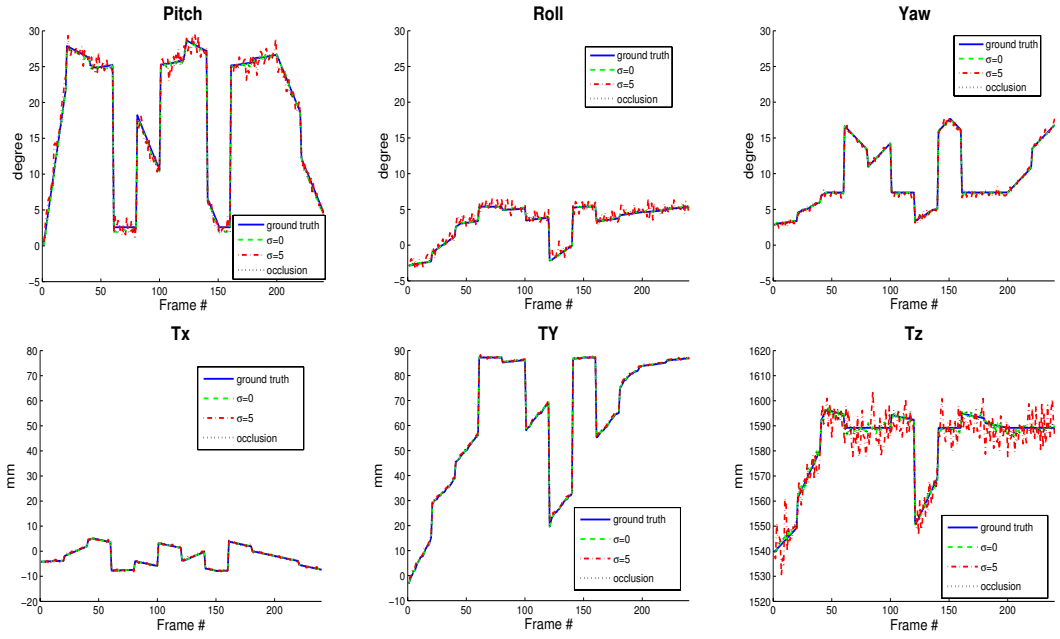

Fig. 2. Comparison between estimated poses and ground truth in a synthetic sequence. In each graph the blue solid curve depicts the ground truth and the dashed green (σ=0), dashed-dotted red curve (σ=5) and dotted black curve (occlusion) depicts the estimated via the described motion estimation algorithm.

difference vector is relatively large. On the other hand, in the case of a converging population the mean difference vector becomes small relatively enabling efficient fine tuning.

4. ROBUST COST FUNCTION

As described in the introduction, a rigid head motion with diffuse illumination without local deformations and occlusions, is assumed for the initial cost function (2). However, in real sequences these assumptions are usually not valid. Therefore, a robust cost function is required, enabling motion estimation in presence of outliers. We use the following robust cost function [8]:

$$C(\lambda) = \sum_{\mathbf{u}\in\Omega} -e^{\left(\frac{-1}{2\kappa}[I(F(\mathbf{u},\lambda),t)-I(\mathbf{u},t_0)]^2\right)} \qquad (6)$$

The parameter κ is related to the outlier-rate and the expectation value σ of the inlier-error. The optimal value of κ decreases with a decreasing inlier-rate. As a rule of thumb, the setting

$$\kappa \approx 10\sigma^2 \qquad (7)$$

enables motion estimation from even highly contaminated observables.

5. EXPERIMENTAL RESULTS

We evaluated the described technique using synthetic and real sequences. Different synthetic sequences are generated by mapping a texture to the face model and then rendering the face model in different positions. Since this paper focuses on estimating the motion in temporally downsampled sequences, a synthetic sequence of 240 frames is generated in which after each 20th image the position of the head is randomly moved. Hence, the position of the head strongly varies between these jumps. This synthetic sequence is used to demonstrate the robustness of the proposed motion estimation algorithm. Camera noise is simulated by adding white noise with zero mean and variance σ^2, which is given in image intensity, to the image sequence. The image intensity ranges from 0 to 255. Furthermore, occlusion consisting of a moving object in front of the head is added to the synthetic sequence. In these test sequences we can compare the true and estimated motion in each frame. In the real sequences, the face model is adapted to fit the geometric shape of the human head and is positioned in the initial frame before the motion estimation is performed.

5.1. Synthetic sequences

In Fig. 2 the estimated and true motion parameters are presented. The estimated pitches, yaws, rolls and translations for σ=0 , σ=5 and occlusion by the system are compared with the ground truth. Their colors are green, red, black and blue respectively. The horizontal axis describes the frame number and the vertical axis means the pitch, yaw or roll in degrees. The translation is described in millimeters. The motion parameters are estimated well throughout the noiseless synthetic sequence, even the jumps do not significantly increase the er-

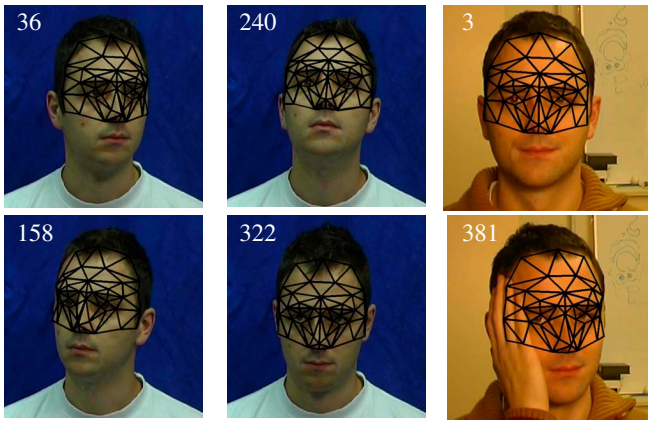

Fig. 3. In the left and middle row the frames 36, 158, 240 and 322 of the first sequence are displayed. The frames between 36 to 158 and 240 to 322 were dropped before the estimation. In the right row occlusion is present.

ror. The error standard deviation of the estimated pitches and z-translations vary from ground truth by less than 0.33 degrees and 1.2mm, respectively. If noise is added to the synthetic sequence, then the accuracy decreases. Especially the error standard deviation, of the estimated z-translation, increases to 5.6mm ($\sigma = 5$) and 1.4mm (occlusion).

5.2. Real sequences

In order to show the performance of the described algorithm, real sequences from a recorded human subject are analyzed. The intrinsic camera parameters are determined by calibration. In Fig. 3 the motion estimation results, from two sequences, with over 800 images, are presented. In the first sequence several frames are dropped in order to simulate fast motion. The presented algorithm has the capability of estimating large motions, so that the algorithm still tracks the human head. In the second sequence occlusion is present. Nevertheless the face model seems to be glued to the head due to precisely estimated motion parameters.

6. CONCLUSIONS

We developed a model-based motion estimation algorithm for full head motion recovery. The algorithm is based on minimizing a robust cost function with the stochastic optimization algorithm called Differential Evolution. This algorithm is very robust with respect to outliers and even large motions between consecutive frames are precisely estimated. Furthermore, we tested the algorithm on synthetic and real sequences and estimated precise 3D motion parameters.

7. REFERENCES

[1] Zicheng Liu and Zhengyou Zhang, "Robust head motion computation by taking advantage of physical properties," in *Workshop on Human Motion*, 2000, pp. 73–77.

[2] H. Li, P. Roivainen, and R. Forcheimer, "3-d motion estimation in model-based facial image coding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 6, pp. 545–555, 1993.

[3] James R. Bergen, P. Anandan, Keith J. Hanna, and Rajesh Hingorani, "Hierarchical model-based motion estimation," in *ECCV '92: Proceedings of the Second European Conference on Computer Vision*. 1992, pp. 237–252, Springer-Verlag.

[4] M. La Cascia, S. Sclaroff, and V. Athitsos, "Fast, reliable head tracking under varying illumination: An approach based on robust registration of texture-mapped 3D models," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, April 2000, vol. 22(4).

[5] F. Dornaika and F. Davoine, "Head and facial animation tracking using appearance-adaptive models and particle filters," in *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 10*, Washington, DC, USA, 2004, p. 153, IEEE Computer Society.

[6] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," in *International Journal of Computer Vision*, 1998, pp. 5–28.

[7] R. Storn and K. Price, "Minimizing the real functions of the icec'96 contest by differential evolution," in *IEEE International Conference on Evolutionary Computation*, Nagoya, May 1996, pp. 842–844.

[8] O. Urfalioglu, "Robust estimation with non linear particle swarm optimization," in *Proceedings of Mirage 2005: Computer Vision / Computer Graphics Collaboration Techniques and Applications*, INRIA Rocquencourt, France, Mar. 2005, pp. 842–844.

[9] Joergen Ahlberg, "Candide-3 - an updated parameterised face," Linkoeping University, 2001.

[10] Axel Weissenfeld, Nikolace Stefanoski, Shen Qiuqiong, and Joern Ostermann, "Adaptation of a generic face model to a 3d scan," in *Proc. 2nd Workshop on Immersive Communication and Broadcast Systems, Berlin, Germany*, Oct. 2005.

[11] K. Price, "Differential evolution: a fast and simple numerical optimizer," in *Biennial Conference of the North American Fuzzy Information Processing Society, NAFIPS*. jun 1996, pp. 524–527, IEEE Press, New York. ISBN: 0-7803-3225-3.