Tracking of Deformable Surfaces in Multiview Image Sequence Coding

Ioannis Kompatsiaris, Dimitrios Tzovaras and Michael G. Strintzis

Information Processing Laboratory
Electrical and Computer Engineering Department
Aristotle University of Thessaloniki, Thessaloniki 54006, Greece
phone: (+30-31) 996-359, FAX: (+30-31) 996-398
e-mail : ikomp@diion.ece.auth.gr

Abstract - This paper describes a procedure for 3D model-based flexible motion estimation using information from all channels of a multiview image sequence. The 3D model is initialized by accurate adaptation of a 2D wireframe model to the foreground object of one of the views. The rigid 3D motion is estimated for each triangle and spatial homogeneity neighbourhood constraints are used to improve the reliability of the estimation efficiency and to smooth the motion field produced. A novel technique is used to estimate flexible motion of the nodes of the wireframe from the rigid 3D motion vectors of the wireframe triangles containing each node. Kalman filtering is used to track both rigid 3D motion of each triangle and flexible deformation of each node of the wireframe. The performance of the resulting 3D flexible motion estimation method is evaluated experimentally.

1. INTRODUCTION

Multiview video processing has been the focus of considerable attention in recent literature \cite{1}, \cite{2}, \cite{3}. In a multiview image sequence, each different view is recorded with a difference in the observation angle, creating an enhanced 3D feeling to the observer, and increased "telepresence" in teleconferencing and several other (medical, entertainment, etc.) applications.

In both monoscopic and multivision, the ability of model-based techniques to describe a scene in a structural way has opened up new areas of applications. Video production, realistic computer graphics, multimedia interfaces and medical visualisation are some of the applications that may benefit by exploiting the potential of model-based schemes. The derivation of 3D models directly from images, usually requires estimation of dense disparity fields, postprocessing to remove erroneous estimates and fixing of a surface model to the calculated depth map. In \cite{4} an algorithm was presented which optimally models each scene using an hierarchical structure derived directly from intensity images. The wire-frame model consists of adjacent triangles that may be split into smaller ones over areas that need to be represented in higher detail. The motion of the model surface using both rigid and non-rigid body assumptions is estimated concurrently with depth parameters.

In the present paper, a procedure for 3D model-based flexible motion estimation using information from all channels of a 3-view image sequence, is proposed. The 3D model is initialized by adapting a 2D wireframe to the foreground object. The adaptation of the wireframe model is a novel procedure resulting in a very accurate representation of the foreground object. Using depth and multiview camera geometry the 2D wireframe is reprojected in the 3D space, forming a consistent wireframe for all views. The rigid 3D motion of each triangle is estimated next, taking into account the temporal correlation between subsequent frames. For each Group of Pictures (GOP) the rigid 3D motion of the first frame is estimated using a least median of squares technique. In order to improve the efficiency and stability of this procedure, homogeneity neighbourhood constraints on the motion of each triangle are imposed. The rigid 3D motion of each triangle for subsequent frames is estimated using a Kalman estimation algorithm. For the flexible motion estimation of each node of the 3D model, the rigid 3D motion vectors of each neighbouring triangle are used in a novel flexible motion estimation procedure. Since more than one observations are available for each node, Principal Component Analysis (PCA) may be used in order to find the best estimate. This procedure is applied only at the beginning of each GOP and the motion of the nodes of the rest of the frames is tracked using Kalman filtering. With the flexible 3D motion of each node available, the model may be updated at the next time instance and all views can be reconstructed using only the 3D flexible motion, 3D model and the previous frame information.

The paper is organised as follows. In the following section, the three-camera geometry is described, and in Section 3 the procedure for the 3D model initialisation is given. Section 4 describes the rigid 3D motion estimation of each triangle for the first frame of each GOP and tracking of the motion at subsequent frames using a Kalman filter approach. The PCA used to find the best estimate among neighbouring triangles is described in Section 5. In Section 6 experimental results are given.

2. CAMERA MODEL

A camera model describes the projection of 3-D points onto a camera target. The model used here is the CAHV model introduced in \cite{5}. In our multiview camera geometry, three cameras c are used : \( c = left, top, right \). For each camera c the model contains the following parameters shown in Fig. 1 : a) position of the camera \( C_c \), b) optical axis \( A_c \), c) horizontal camera target vector \( H \), d) vertical camera target vector \( V_\gamma \) and \( sx, sy \) pixel size.

In our camera model we shall assume that the radial distortion is compensated. In this case projection
of a 3D point \( P \), with coordinates relative to world coordinate system, onto the image plane \((X', Y')\) is
\[
X' = \frac{(P - C_e) \cdot H_e}{(P - C_e) \cdot A_e}, \quad Y' = \frac{(P - C_e) \cdot V_e}{(P - C_e) \cdot A_e}.
\]

Conversely, given its position \((X_e, Y_e)\) on the camera plane, the 3D position of a point can be determined by:
\[
P = C_e + \tau_e \cdot S_e(X_e, Y_e),
\]
where \(S_e(X_e, Y_e)\) is the unit vector pointing from the camera to the point in the direction of the optical axis and \(\tau_e\) is the distance between the 3D point and the center of camera e.

3. 3D MODEL INITIALIZATION

The generation of the 3D model object is based on an initial adaptation of a 2D wireframe model to the foreground object of one of the projected images (left, top or right).

In the present paper very few assumptions are made regarding the content of the scene and the motion of its objects. The a priori information needed consists only of the depth map and a background/foreground segmentation mask; all other information is extracted using only the projected images of all views. For the depth map estimation the algorithms presented in [1] or in [6] can be used, while the foreground object can be separated using motion information or combined motion and depth information [7]. Without loss of generality, to simplify the subsequent discussion, we shall assume that only one object exists in the foreground. However, it is emphasized that the methodology of the paper remains valid and applicable even in the presence of more than one such objects.

The adaptation of the 2D wireframe to the foreground object is a coarse to fine procedure consisting of the following steps:

- A regular grid is first created, covering the full size of the 2D image.
- Only triangles overlapping with the foreground object or with the boundary of the foreground object are retained. The remaining triangles are discarded.
- Nodes of the triangles lying outside of the boundary of the foreground object are forced to meet this boundary.
- The "force" applied for the movement of the outer nodes is propagated to the remaining nodes assuming a Gaussian distribution, as in [8], so as not to drastically alter the regularity of the wireframe.

The result of the above procedure is a very accurately adapted wireframe over the foreground object in one of the views (Fig. 2). By applying Eq. (2) to all nodes \( p = (X, Y) \), the 3D points \( P = (x, y, z) \) are found and the 3D model is formed (Fig. 3). Note that the 2D wireframe is created only as an intermediate step in order to form the 3D model. In the following only the 3D model is used.

4. RIGID 3D MOTION ESTIMATION AND TRACKING OF EACH TRIANGLE

The estimation of rigid 3D motion vectors for each triangle of the 3D model is needed before the evaluation of flexible motion estimation is attempted. In order to increase the efficiency and stability of the triangle motion estimation algorithm, neighbourhood triangles are taken into account. This also results to a smoother 3D motion field, since erroneous large local deformations are suppressed.

4.1 Rigid 3D Motion Estimation

In the first frame of each Group of Pictures (GOP) the rigid motion of each triangle \( T_k \), \( k = 1, \ldots, K \), where \( K \) is the number of triangles in the foreground object, is modeled using a linear 3D model, with three rotation and three translation parameters [9]:
\[
P_{t+1} = R^{(t)} P_t + T^{(t)},
\]
with \( R^{(t)} \) and \( T^{(t)} \) being of the form:
\[
R^{(t)} = \begin{bmatrix}
1 & 0 & 0 \\
-w_x^{(t)} & 1 & 0 \\
-w_y^{(t)} & -w_y^{(t)} & 1
\end{bmatrix}, \quad T^{(t)} = \begin{bmatrix}
\xi_t^{(t)} \\
\eta_t^{(t)} \\
\zeta_t^{(t)}
\end{bmatrix},
\]
where \( P_t = (x_t, y_t, z_t) \) is a 3D point on the plane defined by the coordinates of the vertices of triangle \( T_k \).

In order to guarantee the production of a homogeneous estimated triangle motion vector field, smoothness neighbourhood constraints are imposed for the estimation of the model parameter vector \( \mathbf{u}^{(t)} = (u_x^{(t)}, u_y^{(t)}, u_z^{(t)}, t_x^{(t)}, t_y^{(t)}, t_z^{(t)}) \). We define as neighbourhood \( TS_k \) of each triangle \( T_k \) the ensemble of the triangles neighbouring the triangle \( T_k \), i.e. those sharing at least one common node with \( T_k \), along with \( T_k \).

For the estimation of the model parameter vector \( \mathbf{u}^{(t)} \) the MLS iterative algorithm [10] was used. At time \( t \), each point \( P_i \) in \( TS_k \) is projected to points \((X_{e_i}, Y_{e_i})\), \( i = l, r \) on the planes of the three cameras. Using equations (1) and (3), the projected 2D motion vector, \( d_e(X_{e_i}, Y_{e_i}) \) is determined by
\[
d_e(X_{e_i}, Y_{e_i}) = X_{e_{c+1}}(X_{c+1}) - X_{c+1} = \frac{(R^{(t)} P_t + T^{(t)} - C_e) \cdot H_e}{(P_t - C_e) \cdot A_e} - \frac{(P_t - C_e) \cdot H_e}{(P_t - C_e) \cdot A_e},
\]

DSP 97 – 418
\[ d_p(X_{t+1}, Y_{t+1}) = Y_{t+1} - Y_{t+1} = \left( R^{(k)} P_t + Y^{(k)} - C_{y^k} \right)^T \cdot V_c \quad \left( P_t - C_{c^k} \right)^T \cdot V_c \]

where \( d_p(X_{t+1}, Y_{t+1}) = d_{c^k}(X_{t+1}, Y_{t+1}), d_{y^k}(X_{t+1}, Y_{t+1}) \).

Using the initial 2D motion vectors, estimated by applying a block matching algorithm to the images corresponding to the left, top and right cameras and also using equations (5) and (6), a linear system for the global motion parameter vector \( a^{(k)} \) for triangle \( T_k \) is formed:

\[ u^{(k)} = D^{(k)} \cdot a^{(k)} . \]

This is a system of \( 2 \times 3 \times L \) equations with six unknowns, where \( L \) is the number of 3D points contained in \( T S_k \), since for each 3D point \( P_i \), two equations are formed for the \( X \) and \( Y \) coordinates for each one of the three cameras. Since \( 2 \times 3 \times 6 \) for all triangles (in the worst case \( T S_k \) is composed of a single triangle with \( L = 3 \) and \( 2 \times 3 \times 3 = 18 \)) this is overdetermined and can be solved by the robust least median of squares motion estimation algorithm described in detail in [10]. Erroneous initial 2D estimates, produced by the block-matching algorithm, will be discarded by the least median of squares motion estimation algorithm.

4.2 3D Motion Tracking Using Kalman Filtering

In order to exploit the temporal correlation between consequent frames, a Kalman filter is applied for the calculation of the 3D rigid motion parameters at every time instant. In this way, the estimation of the motion parameters is honed by additional observations as additional frames arrive. Omitting for the sake of notational simplicity, the explicit dependence of the motion parameters to the triangle \( T_k \), thus writing \( a_i, b_i, C_i \) instead of \( a_i^{(k)}, b_i^{(k)}, C_i^{(k)} \), the dynamics of the system are described as follows:

\[ a_{t+1} = a_t + w \cdot e_{t+1} \]

\[ b_{t+1} = D_{t+1} \cdot a_{t+1} + v_{t+1} \]

where \( a \) is the rigid 3D motion vector of each triangle and \( e_t \) is a unit-variance white random sequence. The term \( w \cdot e_{t+1} \) describes the changes from frame to frame and a high value of \( w \) implies small correlation between subsequent frames and can be used to describe fast-changing scenes, whereas a low value of \( w \) may be used when the motion is relatively slow and the temporal correlation is high. The noise term \( v_{t+1} \) represents the random error of the formation of the system (7), modeled as white zero mean Gaussian noise, where \( E[e_n \cdot v_{m}] = F_n \theta \delta(n - m), \) \( n \) being the \( n \)-th element of \( v \).

The equation giving the estimated value of \( \hat{a}_{t+1} \) according to \( \hat{a}_t \) is [11]:

\[ \hat{a}_{t+1} = \hat{a}_t + K_{t+1} \cdot (b_{t+1} - D_{t+1} \cdot \hat{a}_t) \]

where \( \hat{a}_{t+1} \) and \( \hat{a}_t \) are the new and old prediction of the unknown motion parameters corresponding to the \( t+1 \)-th and \( t \)-th frame, respectively, and \( K_{t+1} \) represents the correction matrix.

The initial value \( \hat{a}_0 \) of the filter (beginning of each GOP) is given by solving Eq. (7). In the above, \( w \) and \( v \) are assumed to be the same for the whole mesh, hence independent of the triangle \( T_k \). Notice that Eq. (7) is solved only once in order to provide the initial values for the Kalman filtering. During the next frames \( D \) and \( b \) are only formed and used at the Kalman filter procedure.

5. ESTIMATION OF FLEXIBLE SURFACE DEFORMATION USING PCA

The 3D rigid motion parameters \( a^{(k)} \) estimated from the previously described procedure are input to the 3D flexible motion estimation procedure. The aim of the 3D flexible motion estimation is to assign a motion vector to every node of the 3D model.

The flexible motion of each node of the wireframe \( P_i \), \( i = 1, \ldots, P \), is affected by the 3D rigid motion \( a^{(k)} \) of every triangle connected to \( P_i \), whose 3D motion was estimated by the procedure described in the preceding sections. We propose the use of Principal Component Analysis (PCA) in order to determine the best representative motion vector for node \( P_i \).

More specifically, for each node \( P_i \), \( a_i^{(l)} \), \( l = 1, \ldots, N_i \) observations are available, where \( N_i \) is the number of triangles containing this node. The covariance matrix of \( a_i \) is:

\[ C_i = \frac{1}{N_i} \sum_{l=1}^{N_i} (a_i - \bar{a}_i) (a_i - \bar{a}_i)^T \]

where \( \bar{a}_i \) is the mean value of \( a_i^{(l)} \).

Let \( u_{i,k} \) be the eigenvector of the \( 6 \times 6 \) matrix \( C_i \) corresponding to its \( k \)-th highest eigenvalue. The mean value of the projections of all observations to \( u_{i,k} \) is:

\[ q_{i,k} = \frac{1}{N_i} \sum_{l=1}^{N_i} u_{i,k}^T \cdot (a_i^{(l)} - \bar{a}_i^{(l)}) \]

The best estimated value of the motion parameters for node \( i \) based on \( M_i \) eigenvectors is:

\[ a_i = \frac{1}{M_i} \sum_{m=1}^{M_i} q_{i,m} \cdot u_{i,m} + \bar{a}_i \]

where \( M_i \leq 6 \). The number of \( M_i \) used for each node depends on the corresponding number of dominant eigenvalues.

6. EXPERIMENTAL RESULTS

The proposed model-based coding was evaluated for the coding of left, top, right channels of a real multiview image sequence. The interlaced multiview videoconference sequence "Ludo" of size 720 x 576 \(^1\) was used. All experiments were performed at the top field

\(^1\)This sequence were prepared by the OCEPT for use in the PANORAMA ACTS project.
of the interlaced sequence, thus using images of size 720 × 388.

The 3D motion of each triangle can be estimated using the algorithm in Section 4 and the geometry of the 3D model produced by the techniques described in Section 3. The 3D motion of each triangle was then used as an input to the flexible motion estimation algorithm described in Section 5. Since a 3D motion vector is assigned to each node of the wireframe, succeeding frames can be reconstructed using only the 3D model, the 3D motion and the frames of the previous time instant. The flexible motion estimation was performed between frames 0 and 3 (since differences between frames 0 and 1 was negligible). The original top view of frame 0 can be seen in Fig. 2. The reconstructed top view are shown in Fig. 4. The frame differences between the original frame 0 and 3 are shown in Fig. 5 for all views while the frame differences between the original frame 3 and reconstructed frame 3 are shown in Fig. 6. The performance of the algorithm was also tested in terms of PSNR, giving an improvement of approximately 4 dB for all views, as compared to the PSNR between frame 0 and 3.

7. ACKNOWLEDGEMENTS

This work was supported by the EU CEC Project PANORAMA (ACTS project 092) and the YPER project of the Greek Secretariat of Science and Technology.

REFERENCES


