Proceedings 6th International Conference on Image Processing and its Applications (IPA 97),
pp. 194 – 198, Dublin, Ireland, July 1997.

MODEL-BASED 3D-MOTION ESTIMATION WITH ILLUMINATION COMPENSATION

P. Eisert and B. Girod

University of Erlangen-Nuremberg, Germany

ABSTRACT

In this paper we present a model-based algorithm for the estimation of three-dimensional motion parameters of an object moving in 3D-space. Photometric effects are taken into account by adding different illumination models to the virtual scene. Using the additional information from three-dimensional geometric models of the scene leads to linear algorithms for the parameter estimation of the illumination models which are all computationally efficient. Experiments show that the Peak Signal Noise Ratio (PSNR) between camera and reconstructed synthetic images can be increased by up to 7 dB compared to global illumination compensation. The average estimation error of the motion parameters is at the same time reduced by 40 %.

1. INTRODUCTION

In recent years, model-based coding techniques for very low bit rate video compression have received growing interest [1, 2, 3]. Motion parameters of objects are estimated from video frames using three-dimensional models of the objects. These models describe shape and texture of the objects. At the decoder the video sequence is synthesized rendering the models at the estimated positions. Typical applications for model-based coding are videotelephony, videoconferencing, multimedia applications or animation for TV productions.

To extract the object's relative motion parameters from two successive video frames the encoder in our system uses a hierarchical gradient-based scheme that is combined with a rigid body motion constraint. The virtual scene is updated by moving the geometric model according to these parameters. However, due to errors in the estimation, the position of the model can differ from the real object's one and a mismatch between camera image and synthetic image can occur in long sequence motion tracking. To avoid this error accumulation a feedback loop is introduced at the encoder [2, 4]. The extracted motion parameters are not only transmitted to the decoder but are also used to render the same synthetic image at the encoder (Figure 1). The motion estimation is then performed between the actual camera frame $I(k)$ and the previous synthetically generated image $\hat{I}(k-1)$. This ensures that the projection of the 3D shape model and the 2D image are consistent. However, the 3D

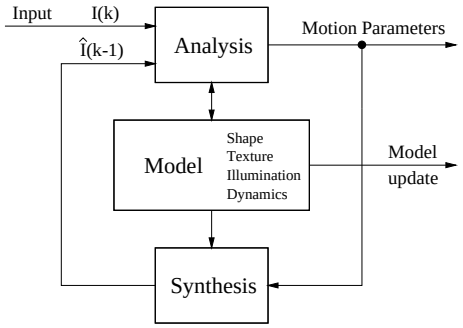

Figure 1: Feedback structure of the coder.

model used for rendering the synthetic images cannot describe the object in the camera images perfectly. Model failures make motion estimation more difficult and reduce the quality of the synthesized images. In [5] Pentland showed that illumination has a large influence on the appearance of the objects. If the illumination in the real scene and during model acquisition differs, significant model failures can arise. For minimization of model failures illumination models are added to the synthetic scene and both motion and illumination parameters are estimated alternately. This allows dealing with varying illumination conditions during the recording of the video sequence. In this paper we show that adding illumination models to the synthetic scene improves the accuracy of the estimated motion parameters as well as the registration of synthetic and real images. In contrast to the work of Stauder [6] and Bozdagi et al. [7] three-dimensional models of the moving objects are used leading to linear illumination estimation algorithms with low computational complexity.

2. BASIC GEOMETRY

The three-dimensional scene used for parameter estimation and rendering of the synthetic images consists of a camera model and a model of the rigid object moving in 3D space. The camera model and its associated coordinate systems are shown in Figure 2.

The 3D coordinates of an object point $[x\ y\ z]^T$ are

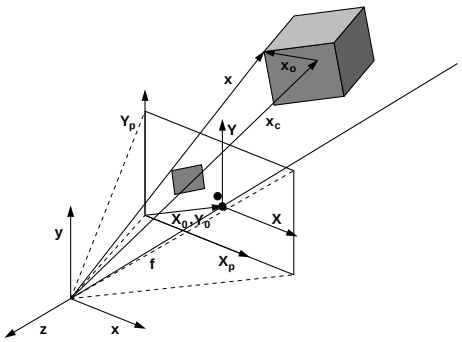

Figure 2: Scene geometry.

projected onto the image plane assuming perspective projection:

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

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

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

The object moving in the scene is assumed to be rigid and therefore the motion can be described by a rotation R around the object center $\vec{x}_c$ and a translation $\vec{t}$. The 3D position of an object point $\vec{x}$ after a rigid body motion is given by

$$\vec{x'} = R(\vec{x} - \vec{x}_c) + \vec{x}_c + \vec{t}. \quad (3)$$

Under the assumption of small rotations between two successive frames the rotation matrix R can be linearized as follows

$$R \approx \begin{bmatrix} 1 & -r_z & r_y \\ r_z & 1 & -r_x \\ -r_y & r_x & 1 \end{bmatrix}. \quad (4)$$

3. MOTION ESTIMATION

The entire image is used for motion estimation and we set up the optical flow constraint equation for all pixels

$$I_{Xp} \cdot u + I_{Yp} \cdot v + I_t = 0 \quad (5)$$

where $[I_{Xp}\ I_{Yp}]$ 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 rotation and translation from (5) together with the three-dimensional motion equation of the object

$$\begin{aligned} x' &\approx x\left(1 + r_y\frac{z - z_c}{x} - r_z\frac{y - y_c}{x} + \frac{t_x}{x}\right) \\ y' &\approx y\left(1 + r_z\frac{x - x_c}{y} - r_x\frac{z - z_c}{y} + \frac{t_y}{y}\right) \\ z' &\approx z\left(1 + r_x\frac{y - y_c}{z} - r_y\frac{x - x_c}{z} + \frac{t_z}{z}\right) \end{aligned} \quad (6)$$

in a similar way to the method described in [8]. We now combine (1) and (6) to obtain the projected image points. Assuming only small object motion between successive frames and using a first order approximation of the resulting equation leads to the following relation for the 2D pixel displacements:

$$\begin{aligned} u = X'_p - X_p \approx \\ -f_x\left(r_y(1 - \frac{z_c}{z}) + r_z(Y_n + \frac{y_c}{z}) + \frac{t_x}{z} - \right. \\ \left. X_n\left(r_x(Y_n + \frac{y_c}{z}) - r_y(X_n + \frac{x_c}{z}) - \frac{t_z}{z}\right)\right) \\ v = Y'_p - Y_p \approx \\ f_y\left(r_z(X_n + \frac{x_c}{z}) + r_x(1 - \frac{z_c}{z}) - \frac{t_y}{z} + \right. \\ \left. Y_n\left(r_x(Y_n + \frac{y_c}{z}) - r_y(X_n + \frac{x_c}{z}) - \frac{t_z}{z}\right)\right). \end{aligned} \quad (7)$$

Except for the motion parameters t_x, t_y, t_z, r_x, r_y and r_z all other parameters are known. The object center has already been calculated in the previous frame and the distance z of the object points from the camera origin can be determined from the 3D model of the object. Combining (5) and (7) results in a linear equation for the 6 unknown motion parameters that can be set up at each pixel location that is part of the object

$$a_0 r_x + a_1 r_y + a_2 r_z + a_3\frac{t_x}{z_c} + a_4\frac{t_y}{z_c} + a_5\frac{t_z}{z_c} = -I_t. \quad (8)$$

The parameters a_0 to a_5 are given by

$$\begin{aligned} a_0 &= I_{Xp} f_x X_n Y_{nc} + I_{Yp} f_y\left(1 - \frac{z_c}{z} + Y_n Y_{nc}\right) \\ a_1 &= -I_{Xp} f_x\left(1 - \frac{z_c}{z} + X_n X_{nc}\right) - I_{Yp} f_y Y_n X_{nc} \\ a_2 &= -I_{Xp} f_x Y_{nc} + I_{Yp} f_y X_{nc} \\ a_3 &= -I_{Xp} f_x \frac{z_c}{z} \\ a_4 &= -I_{Yp} f_y \frac{z_c}{z} \\ a_5 &= -I_{Xp} f_x X_n \frac{z_c}{z} - I_{Yp} f_y Y_n \frac{z_c}{z} \end{aligned} \quad (9)$$

with the abbreviations

$$X_{nc} = X_n + \frac{x_c}{z} \ , \ Y_{nc} = Y_n + \frac{y_c}{z}. \qquad (10)$$

Because the object usually covers more than 6 pixels, we obtain an overdetermined system that can be solved in a least-squares sense. The high number of equations makes it also possible to discard some potential outliers that can be estimated from the gradient values before solving the linear system.

The optical flow constraint equation assumes the luminance being locally linear and is therefore only able to handle very small displacements. To overcome this requirement a hierarchical coarse-to-fine approach with subsampled images is used to increase the range of possible motions. First, an initial estimate for the motion parameters is computed from highly subsampled images. With these parameters a motion compensated synthetic image is generated that is now much closer to the camera image. This step is repeated at higher resolutions to decrease the residual error. With four different levels of resolution starting from 44 by 36 pixels and ending with CIF resolution the algorithm converges for translations up to 30 pixels and rotations up to 15 degrees between two successive frames.

4. ILLUMINATION ESTIMATION

To improve the accuracy of the motion estimation and the realism of the synthesized images the illumination condition of the scene is taken into account by incorporation of illumination models. The parameters of the models describing the reflectance distribution are estimated from the video sequence.

The first illumination model presented here is a Lambertian model [9, 10] where the assumed illumination consists of ambient and directional light. In [6] Stauder uses this model for object-based coding and estimates the illumination parameters from two successive real video frames with a nonlinear iterative scheme. Bozdagi et al. [7] determine illumination direction and surface albedo with the method proposed by Zheng et al. [11] that is based on a surface approximation by spherical patches. In our scheme the illumination differences between a camera image and the corresponding synthetic image are estimated. The additional information provided by the object model simplifies this task and leads to a linear algorithm for the parameter estimation. The 3D-model is assumed to be homogeneously illuminated with ambient light during the acquisition of the model. For each pixel i belonging to the object we obtain one equation

$$I_{cam,i} = I_{syn,i} \cdot (k_{amb} + k_{dir} \cdot \max(-\vec{l} \cdot \vec{n}_i, 0)) \qquad (11)$$

describing the illumination differences between the synthetic and the camera images. I_{cam} and I_{syn} denote the pixel intensities of camera and synthetic image, k_{amb} and k_{dir} the reflection coefficients, $\vec{l}$ the direction of the direct light and $\vec{n} = [n_x \ n_y \ n_z]^T$ the surface normal of unit length corresponding to that pixel. Both I_{syn} and $\vec{n}$ are given by the object model. We therefore have to estimate four parameters: two specifying the normalized direction of the point light source $\vec{l}$ and the two reflection coefficients k_{amb} and k_{dir}. Only those equations are considered where the maximum function is probably different from zero. Those pixels can be determined from the illumination direction of the previous frame and we obtain an overdetermined linear system of equations (12) that can be solved in a least-squares sense with small computational effort.

$$I_{cam} = I_{syn} \begin{bmatrix} 1 & -n_x & -n_y & -n_z \end{bmatrix} \begin{bmatrix} k_{amb} \\ k_{dir} l_x \\ k_{dir} l_y \\ k_{dir} l_z \end{bmatrix} \qquad (12)$$

Once the parameters for the illumination model are obtained the differences between synthetic and camera images can be compensated applying equation (11) on each pixel of the synthetically rendered frame which results in an estimate for the camera image.

The Lambertian approach can easily be extended to handle also non Lambertian reflection functions. For that purpose higher orders of the surface normal components are added to equation (12). Using a second order approximation leads to

$$I_{cam} = I_{syn} \begin{bmatrix} 1 \ n_x \ n_y \ n_z \ n_x^2 \ n_y^2 \ \ n_x n_y \ \ n_x n_z \ \ n_y n_z \end{bmatrix} \vec{k} \qquad (13)$$

with nine unknowns in $\vec{k}$. The illumination compensation is performed similar to the previous method.

A third and more general approach for the illumination estimation is the use of a reflectance map [9]. The reflection which is assumed to be a function of the surface normal $\vec{n}$ is not defined by an explicit model, but described by a number of sampling points for discrete values of the normal vectors represented by a table. The relation between the pixel intensities for pixel i of the synthetic image I_{syn} and the illuminated camera image I_{cam} is given by

$$I_{cam,i} = I_{syn,i} \cdot r(\vec{n}_i). \qquad (14)$$

The discrete entries of the table, approximating the reflection $r(\vec{n})$, can be estimated from the quotients of real and synthetic pixel intensities $\frac{I_{cam,i}}{I_{syn,i}}$ belonging to the corresponding normal directions. After acquisition of the reflectance map the synthetic image is adjusted to the camera image multiplying each pixel by the reflectance map entry that is specified by the surface normal at that point.

Below, the estimated reflection function of the head object in Figure 3 is shown for the three approaches.

The reflection map is depicted in 3 and the approximations of the Lambertian and the second order approach can be seen in Figure 4.

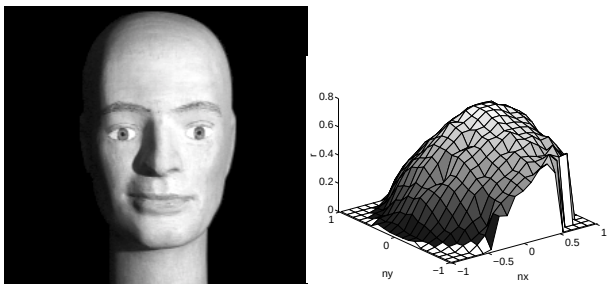

Figure 3: Camera image of a head object that is mainly illuminated from the right (left) and its estimated reflectance map (right).

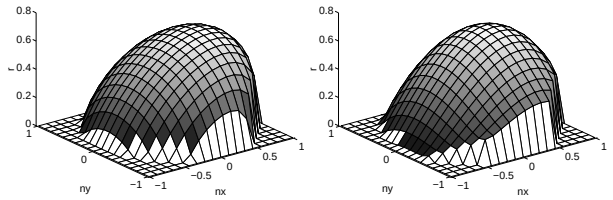

Figure 4: Estimated reflection function with Lambertian (left) and second order approach (right).

5. EXPERIMENTAL RESULTS

The algorithms are tested for both synthetic and real image data. To obtain the structure and the texture for the 3D-model an artificial human head (Figure 3) is scanned with a 3D laser scanner. With this model a synthetic video sequence (CIF resolution) is generated and the mean absolute estimation error is determined. and the results of the estimated motion parameters are compared to the correct values. For the synthetic frames the algorithm is very accurate as can be seen from the first line of Table 1. Additionally, a video sequence with CIF resolution is recorded with a calibrated camera. Camera calibration is employed to get the absolute 3D-position of the object in the camera coordinate system. This enable us to determine the alignment of the moving object and the 3D model of the laser scanner. The motion estimation is performed both with and without illumination compensation. The resulting errors averaged over 14 different motion parameter sets are shown in Table 1. The average error magnitude is reduced in all cases when performing an illumination estimation and compensation. For all three proposed illumination models similar results are obtained. Even more evident is the improvement of the similarity between

	$\Delta\Theta$	Δt_x	Δt_z
Synthetic sequence	0.007^o	0.01 mm	0.06 mm
Video sequence without illumination compensation	0.45^o	0.61 mm	3.9 mm
Video sequence with illumination compensation	0.38^o	0.15 mm	1.5 mm

Table 1: Average error magnitude of the estimated motion parameters ($\Delta\Theta$: mean error of the angle of rotation, Δt_x, Δt_z: error of the translation in x and z direction).

the camera and the synthetic image after motion compensation and illumination adaption as illustrated in Figure 5. The PSNR values for the different methods of illumination compensation are shown in Table 2. In spite of the quite homogeneous illumination condition during the recording of the video sequence, with the proposed algorithms increases of up to 7 dB are achieved compared to a global illumination compensation where only the ambient part of the light is estimated. The reflectance map and the second order method showed a slightly better performance compared to the Lambertian approach.

Figure 5: Frame difference after motion and global illumination compensation (left) and frame difference with Lambertian approach for illumination compensation (right).

	$PSNR$	$\Delta PSNR$
Ambient	29.5	0
Lambert	35.5	6.0
Second order approach	36.5	7.0
Reflectance map	36.5	7.0

Table 2: Average PSNR and increase in PSNR compared to a global illumination compensation

The illumination estimation is also tested under different illumination conditions where the position and

the number of light sources are varied. Motion and illumination compensation is performed on the video sequences and some results of the compensated synthetic sequences are shown in Figure 6.

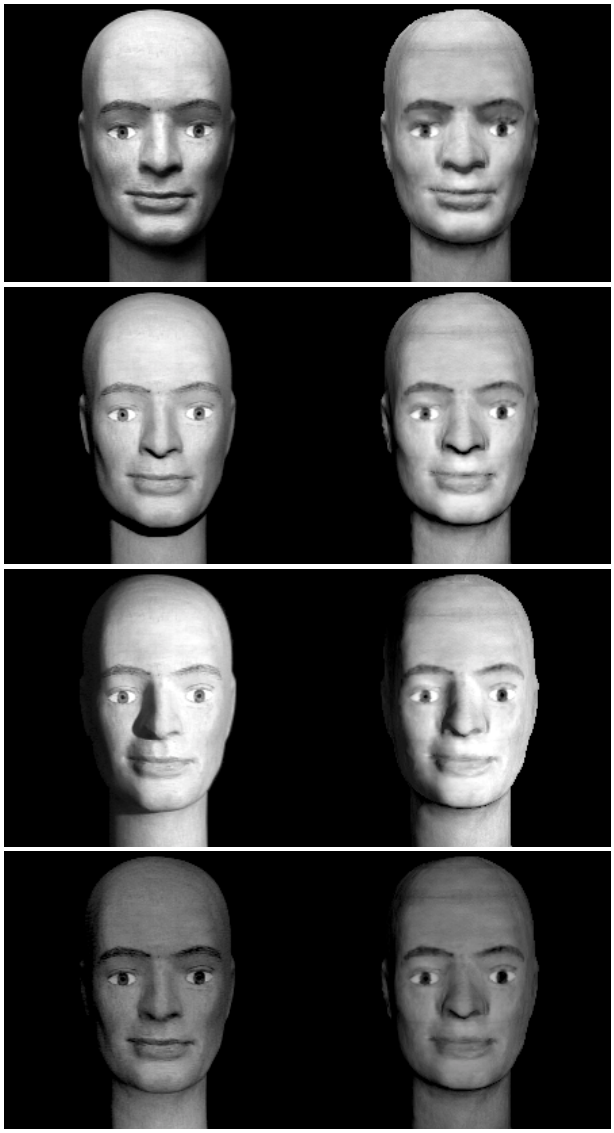

Figure 6: Original video frames recorded under varying illumination conditions (left) and synthetic images after illumination compensation (right).

6. CONCLUSIONS

It has been shown in our paper that illumination has a large influence on the accuracy of the model-based motion estimation. Considering photometric effects increases the performance of the estimation algorithms. For that purpose three illumination models are proposed which describe illumination differences between the 3D-model and the real scene. Due to the fact that surface normals and illumination conditions during the acquisition of the object model are known, the derived algorithms are linear and exhibit low computational complexity. Applying the illumination compensation schemes on real video sequences reduces the error of the estimated motion parameters by about 40 % in average and improves the PSNR between the original and the synthetic image up to 7 dB.

References

[1] B. Girod, "Image sequence coding using 3D scene models", *SPIE Symposium on Visual Communications and Image Processing*, vol. 3, pp. 1576–1591, Sep. 1994.

[2] H. Li, A. Lundmark, and R. Forchheimer, "Image sequence coding at very low bitrates: A review", *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 589–609, Sep. 1994.

[3] MPEG-4, *SNHC Verification Model 3.0, Document N1545*, Feb. 1997.

[4] 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, Jun. 1993.

[5] A. Pentland, "Photometric motion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 879–890, Sep. 1991.

[6] J. Stauder, "Estimation of point light source parameters for object-based coding", *Signal Processing: Image Communication*, pp. 355–379, 1995.

[7] G. Bozdagi, A. M. Tekalp, and L. Onural, "3-D motion estimation and wireframe adaption including photometric effects for model-based coding of facial image sequences", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 3, pp. 246–256, Jun. 1994.

[8] 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, Jun. 1993.

[9] B. K. P. Horn, *Robot Vision*, MIT Press, Cambridge, 1986.

[10] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics, Principles and Practice*, Addison-Wesley, 2 edition, 1990.

[11] Q. Zheng and R. Chellappa, "Estimation of illuminant direction, albedo, and shape from shading", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 680–702, Jul. 1991.