3D Image Analysis and Synthesis, pp. 61–66,Erlangen, November 1996.

Illumination Compensated Motion Estimation for Analysis Synthesis Coding

Peter Eisert and Bernd Girod

Telecommunications Institute, University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
Email: {eisert,girod}@nt.e-technik.uni-erlangen.de

Abstract

In this paper we present two methods to improve the accuracy of three-dimensional motion estimation in analysis synthesis coding by using illumination models. A Lambertian and a reflectance map approach are given which are both computationally efficient. Experiments on real images show that the mean squared error between synthetic and camera images compared to global illumination compensation can be reduced by a factor of 4 and 5, respectively.

1 Introduction

Model-based video coding is a promising approach for very low bit rate compression. In a model-based video coding system three-dimensional motion and scene structure are analyzed using models of the objects. At the decoder the scene is synthesized with respect to the same models using parameters received from the coder. An important application for model-based coding is video telephony where the scene contents is usually restricted to head and shoulders. The coder has to extract information of facial motion and mimics. The estimated parameters (e.g. action units [3]) in combination with a 3D model of the head are sufficient to reconstruct a realistic appearance of the speaking person at the decoder. Data rates of less than 1 kbit/s are achievable by transmitting just a small number of parameters and a small amount of side information for model updates [4]. The structure of such a system is shown in Figure 1.

The coder in our system uses a hierarchical gradient-based scheme to extract the relative motion parameters between two successive frames in head-and-shoulder scenes. In order to obtain the absolute 3D position of the head model, the sum of all previously estimated motion vectors must be taken into consideration. This can lead to an error accumulation after several video frames due to small errors in the motion estimation process. To avoid a mismatch between synthetic and camera images a feedback loop is introduced at the coder as shown in Figure 2 [1],[2]. The extracted motion parameters are not only transmitted to the decoder but are also used to render the same synthetic image at the coder. The motion estimation is then performed between the new camera frame $I(k)$ and the previous synthetically generated image $\hat{I}(k-1)$ which ensures that the 3D shape model and the 2D image are consistent and no error accumulation occurs.

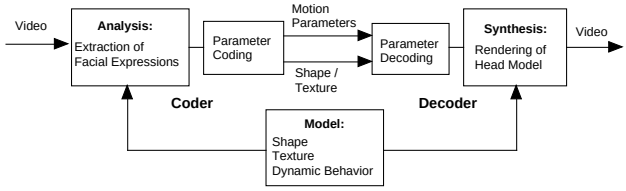

Figure 1: Structure of a model-based system

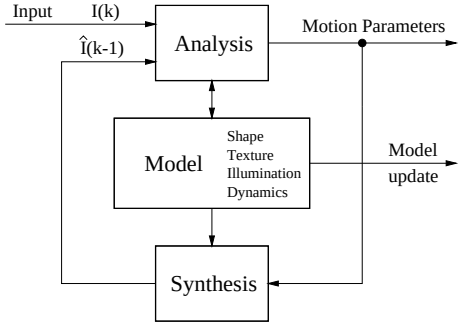

Figure 2: Feedback structure of the coder

However, the 3D 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. Different illumination conditions are one important cause of model failures.

For minimization of model failures illumination models are added to the 3D scene and both motion and illumination are estimated. The synthetic images can then be adapted to the actual illumination conditions. In this paper we show that the accuracy of the estimated motion parameters as well as the registration of synthetic and real image are significantly improved. Due to the use of linear illumination models the additional computational effort remains small.

This paper is organized as follows. First, the basic geometry and the camera model used for the motion estimation are shown. We then briefly discuss the estimation of motion parameters for a rigid body moving in 3D space. After that we describe two different approaches for the estimation of illumination conditions. Finally, results for the proposed methods are presented.

2 Camera Model

The three-dimensional scene used for the parameter estimation and the rendering of the synthetic images consists of a camera model, a head model and an illumination model. The camera model and its associated coordinate systems are shown in Figure 3.

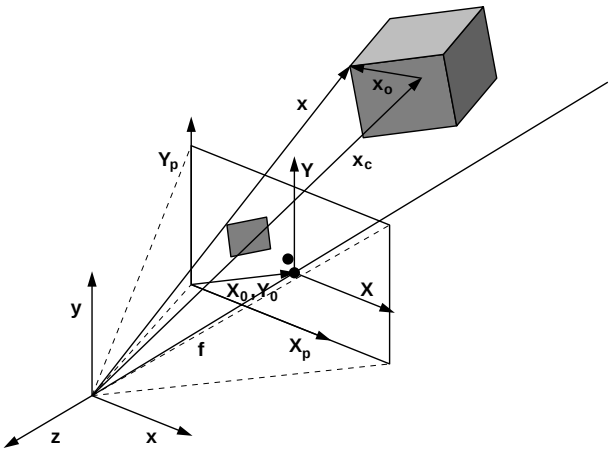

Figure 3: Scene geometry

The 3D coordinates of an object point $[x\ y\ z]^T$ are projected into the image plane assuming perspective projection.

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

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 X and Y into pixel coordinates X_p and Y_p according to equation (2). In addition, they allow the use of non-square pixel geometries.

$$f_x = f \cdot s_x \quad f_y = f \cdot s_y \tag{2}$$

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. All four parameters f_x, f_y, X_0 and Y_0 are obtained from an initial camera calibration and remain constant during a video sequence as long as no change in the focal length (zoom) is applied.

For simplicity, normalized pixel coordinates X_n and Y_n are introduced.

$$X_n = \frac{X_p - X_0}{f_x}\ ,\ Y_n = \frac{Y_p - Y_0}{f_y} \tag{3}$$

3 Rigid body motion

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

$$\vec{x'} = R(\vec{x} - \vec{x}_c) + \vec{x}_c + \vec{t}. \tag{4}$$

The rotation is described by a normalized rotation axis and an angle Θ specifying the amount of rotation around this axis. Under the assumption of a small rotation between two successive frames a linearized variant of the rotation matrix R can be derived.

$$\begin{aligned} R &= \begin{bmatrix} 1 & -n_z\Theta & n_y\Theta \\ n_z\Theta & 1 & -n_x\Theta \\ -n_y\Theta & n_x\Theta & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & -r_z & r_y \\ r_z & 1 & -r_x \\ -r_y & r_x & 1 \end{bmatrix}. \end{aligned} \tag{5}$$

Motion estimation requires the point correspondences in the two-dimensional image plane. The three-dimensional constraint

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

is projected in the image plane according to (1). Assuming only small object motion between successive frames leads to the following equation for the 2D pixel displacements.

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

This equation is a linearized description of the point correspondences in the image plane with the unknown motion parameters t_x, t_y, t_z, r_x, r_y and r_z. The object center has already been calculated in the previous frame and the distance of the object points from the camera origin z can be determined from the 3D model of the object.

4 Motion Estimation

For the motion estimation we use the optical flow constraint equation

$$I_{Xp} \cdot u + I_{Yp} \cdot v + I_t = 0 \quad (8)$$

where $[I_{Xp}\ I_{Yp}]$ is the gradient of the intensity at point $[X_p\ Y_p]$, u and v the displacement in x- and y-direction and I_t the intensity gradient in temporal direction. We do not compute the optical flow field by using additional smoothness constraints and then extracting the motion parameters from this flow field, but estimate the translation and rotation of the rigid body by means of the constraint given in (7). This constraint directly characterizes the displacement $[u\ v]$ and together with (8) a linear equation system for the six unknowns can be set up. For each pixel of the object we obtain one equation

$$a_0 r_x + a_1 r_y + a_2 r_z + a_3\frac{t_x}{z_c} + a_4\frac{t_y}{z_c} + a_5\frac{t_z}{z_c} = -I_t, \quad (9)$$

where a_0 to a_5 are given by

$$\begin{aligned} a_0 &= I_{Xp} f_x X_n Y_{nc} + I_{Yp} f_y\left(1 - \frac{z_c}{z} + Y_n Y_{nc}\right) \\ a_1 &= -I_{Xp} f_x\left(1 - \frac{z_c}{z} + X_n X_{nc}\right) - I_{Yp} f_y Y_n X_{nc} \\ a_2 &= -I_{Xp} f_x Y_{nc} + I_{Yp} f_y X_{nc} \\ a_3 &= -I_{Xp} f_x \frac{z_c}{z} \\ a_4 &= -I_{Yp} f_y \frac{z_c}{z} \\ a_5 &= -I_{Xp} f_x X_n \frac{z_c}{z} - I_{Yp} f_y Y_n \frac{z_c}{z} \quad (10) \end{aligned}$$

with the abbreviations

$$X_{nc} = X_n + \frac{x_c}{z}\ ,\ Y_{nc} = Y_n + \frac{y_c}{z}. \quad (11)$$

Because the object usually covers more than 6 pixels we obtain a highly overdetermined system that can be solved in a least-squares sense

$$[r_x\ r_y\ r_z\ \frac{t_x}{z_c}\ \frac{t_y}{z_c}\ \frac{t_z}{z_c}]^T = -(A^T A)^{-1} A^T I_t. \quad (12)$$

Matrix A consists of the coefficients a_0 to a_5 computed for each pixel of the object. 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 very small 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 the algorithm converges for translations up to 30 pixels and rotations up to 15 degrees between two frames (CIF resolution).

5 Lambertian Illumination

In order to increase the accuracy of the motion estimation an illumination model is added to the 3D scene. The first model presented here is a Lambertian model [6], [7] where the assumed illumination consists of ambient light and directional light. The ambient component is diffuse and non-directional whereas the direct light can be generated by a point light source that is sufficiently far from the object being shaded. For this model four parameters must be estimated: two specifying the direction of the point light source and two for the intensity of the direct and the ambient light, respectively. In contrast to the work of Stauder [5] who estimates the illumination parameters from two real video frames, the illumination differences here are estimated between real and synthetic images. The synthetic image, however, is generated from the 3D-model which is homogeneously illuminated with ambient light. This homogeneous illumination can be achieved by the knowledge of the illumination conditions during the creation of the model. The main advantage of this approach is that the illumination can now very easily be estimated by minimizing the error in a linear least-squares sense. The relation between corresponding pixel intensities in the real and synthetic image is given by

$$I_{real} = I_{syn} \cdot (k_{amb} + k_{dir} \cdot max(-\vec{l} \cdot \vec{n}, 0)) \quad (13)$$

where I_{real} and I_{syn} are the pixel intensities of the images, 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 the pixel. The only nonlinear term in this equation is the maximum function which can be eliminated by taking into account only those areas where the dot product between surface normal and light direction is less than zero. For this purpose an initial light direction must be assumed. However, the estimate is robust over a wide range of initial directions. If this is not sufficient, the calculated estimate can be used as the initial light direction for a second iteration step.

For each pixel of the object we obtain one linearized equation

$$[I_{syn}\ -I_{syn}n_x\ -I_{syn}n_y\ -I_{syn}n_z] \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = I_{real} \quad (14)$$

leading to an overdetermined linear system with four unknowns that is solved with a least square estimator. The illumination parameters are extracted from the solution using the fact that the direction vector $\vec{l}$ is normalized to unity.

$$\begin{aligned} k_{amb} &= x_0 \\ k_{dir} &= \sqrt{x_1^2 + x_2^2 + x_3^2} \\ l_x &= \frac{x_1}{k_{dir}} \\ l_y &= \frac{x_2}{k_{dir}} \\ l_z &= \frac{x_3}{k_{dir}} \end{aligned} \quad (15)$$

Once the parameters for the illumination model are obtained the synthesized image can be illumination compensated using equation (13) for each pixel of the object.

6 Reflectance Map

A second and more general approach for the illumination estimation is the use of a reflectance map [6]. This method is not restricted to Lambertian reflection and can handle multiple light sources. The reflection which is a function of the surface normal $\vec{n}$ is not defined by an explicit model (e.g. equation (13) for the Lambertian approach), but described by a number of sampling points for discrete values of the normal. The surface normal has two degrees of freedom leading to a function that can be represented by a table. The relation between the pixel intensities for pixel i between the synthetic image I_{syn} and the illuminated camera image I_{real} is given by

$$I_{real,i} = I_{syn,i} \cdot r(\vec{n}_i). \quad (16)$$

With the assumption of a homogeneously illuminated object model, the discrete entries of the table, approximating the reflection $r(\vec{n})$, can be estimated from the quotients of real and synthetic pixel intensities $\frac{I_{real,i}}{I_{syn,i}}$ belonging to the corresponding normal directions.

Due to the discrete character of the table the reflectance coefficient of the surface can be approximated more accurately by increasing the size N of the table. Figure 4 shows the dependency between table size and mean squared error (MSE) of synthetic and camera images after motion and illumination compensation.

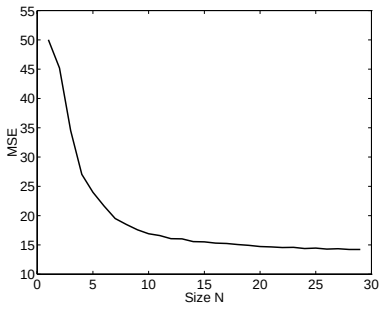

Figure 4: MSE after motion and illumination compensation for different table sizes of the reflectance map

After acquisition of the reflectance map the synthetic image is adjusted to the camera image by multiplying each pixel by the reflectance map entry that is specified by the surface normal at that point. For normal directions not directly corresponding to a table entry a bilinear interpolation of the four table neighbors is performed.

Figure 5 shows the estimated reflectance as a function of the surface normal (n_x and n_y) for an object that is mainly illuminated from the right. The left image is computed with the Lambertian model with an ambient and a directional light component. The image on the right side is obtained from the reflectance map which can better approximate the behavior of the reflectance. On the other hand the surface is not as smooth as the one for the Lambertian model that has fewer degrees of freedom.

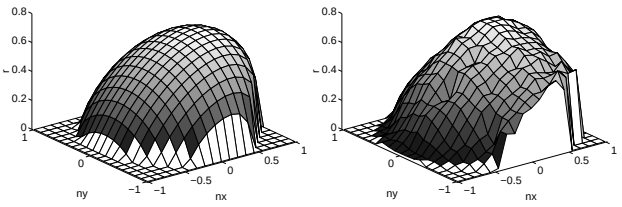

Figure 5: Estimated reflectance for Lambertian approach (left) and reflectance map (right)

7 Results and Experiments

The algorithm was tested for both synthetic and real image data. A video sequence of a rigid head (Figure 7) is recorded with a calibrated camera and well-defined motion parameters. In addition, the head is scanned with a 3D laser scanner [8] to obtain the three-dimensional structure and the texture of the head. The texture quality is further improved by extracting texture elements from multiple views. The motion parameters of the video sequence are estimated both with and without illumination compensation and are compared to the correct values. Motion and illumination estimation are alternately applied.

Table 1 shows the average error of the motion parameters for camera images in CIF resolution (352 x 288 pixels). The head object is rotated around the y-axis and translated along coordinate directions. The amount of motion is varied from 3 to 12 degrees for the rotation and 3 to 48 mm for the translation. As shown by the table the average error magnitude calculated from 14 frames is reduced in all cases when performing an illumination estimation and compensation. For the given geometries the translational error of 0.15 mm corresponds to a displacement of 0.08 pixels for CIF resolution, a rotational error of 0.38^o leads to a displacement of about 0.5 pixels at the tip of the nose. Larger variations of t_z are caused by the relatively small viewing angle of 30^o.

	$\Delta\Theta$	Δt_x	Δt_z
Without illumination compensation	0.45^o	0.61 mm	3.9 mm
With illumination compensation	0.38^o	0.15 mm	1.5 mm

Table 1: Average error magnitude of the estimated motion parameters from 14 frames

Even more evident is the improvement of the difference image between the camera and the synthetic image after the illumination adaption. Figure 6 shows the difference between both images after motion compensation. On the left hand side only the ambient part of the light is estimated which corresponds to a global illumination compensation for the whole object. Especially at the

sides of the head and at the nose large differences are visible due to normal variation of these surface parts. However, when using the proposed illumination compensation these errors vanish as it can be seen on the right hand side of Figure 6.

Figure 6: Differential images after global compensation (left) and with Lambertian approach (right)

Table 2 shows the mean squared error after motion compensation for the different methods of illumination compensation averaged over 14 frames. This error is defined as

$$MSE = \frac{1}{WH} \sum_{x,y}^{W,H} (I_{real}(x,y) - I_{syn}(x,y))^2 \quad (17)$$

with W, H being the width and height of the images and I_{real} and I_{syn} the intensities of the camera and the synthetic image, respectively. The use of the Lambertian model decreases the MSE to 25.2 % compared to the global light compensation with an ambient light model. With the reflectance map a reduction to 20 % can be achieved. Figure 7 illustrates the similarity of camera and synthetic images after the use of illumination models.

	MSE	$MSE/MSE_{Ambient}$
Ambient	72.29	100 %
Lambert	18.16	25.2 %
Reflectance map	14.52	20.0 %

Table 2: Average MSE and reduction of MSE compared to a global light compensation

8 Conclusions

In this paper we have presented an algorithm for the estimation of 3D motion parameters in analysis synthesis coding. It has been shown that

Figure 7: Similarity of camera image (left) and synthetic image (right)

the accuracy of the estimation can be increased by adding illumination models. Two methods, a Lambertian model and a reflectance map approach, were proposed. Experiments showed that the MSE between camera and synthetic images can be reduced up to a factor of 5 by using illumination compensation.

References

[1] H. Li, A. Lundmark, and R. Forchheimer, "Image Sequence Coding at Very Low Bitrates: A Review", *IEEE Trans. on Image Processing*, 3(5), pp. 589–609, September, 1995.

[2] R. Koch, "Dynamic 3-D Scene Analysis through Synthesis Feedback Control", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(6), pp. 556–568, June, 1993.

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

[4] B. Girod, "Image Sequence Coding using 3D Scene Models", *SPIE Symposium on Visual Communications and Image Processing*, September, 1994.

[5] J. Stauder, "Estimation of Point Light Source Parameters for Object-Based Coding", *Signal Processing: Image Communication 7*, pp. 355–379, 1995.

[6] B. K. P. Horn, "Robot vision", *MIT Press, Cambridge*, 1986.

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

[8] Cyberware Laboratory Inc., Monterey, California, "Cyberware MODEL 3030 Digitizer Operating Manual, Revision 10-93",1993.