OBJECT BASED 3-D MOTION AND STRUCTURE ESTIMATION

A. Aydın Alatan and Levent Onural

Electrical-Electronics Engineering Department
Bilkent University
TR-06533 Bilkent Ankara TURKEY
e-mail: alatan@ee.bilkent.edu.tr

ABSTRACT

Motion analysis is the most crucial part of object-based coding. A motion in 3-D environment can be analyzed better by using a 3-D motion model compared to its 2-D counterpart and hence may improve coding efficiency. Gibbs formulated joint segmentation and estimation of 2-D motion not only improves performance, but also generates robust point correspondences which are necessary for linear 3-D motion estimation algorithms. Estimated 3-D motion parameters are used to find the structure of the previously segmented objects by minimizing another Gibbs energy. Such an approach achieves error immunity compared to linear algorithms. Experimental results are promising and hence the proposed motion and structure analysis method is a candidate to be used in object-based (or even knowledge-based) video coding schemes.

1. INTRODUCTION

In order to apply object based coding for compression of video signals, a powerful segmentation and a generic motion model is needed. Since moving objects are usually rigid in 3-D world, it might be more advantageous to perform the analysis of motion in 3-D, instead of the projected 2-D optic flow.

According to Newtonian mechanics, *rigid body* motion can be represented only by *rotation* and *translation*, and in 3-D world object motions can be assumed to be rigid or articulated (connected rigid portions) in most of the cases. Hence the analysis of 3-D motion is a good candidate for effective motion compensation and video encoding.

In this research, 3-D motion and structure estimation and segmentation of a complex scene, consisting of multiple rigid objects is achieved by using a Bayesian formulation and a linear algorithm [1]. Using the estimated 3-D motion parameters, the unknown depth field is found by minimizing another Gibbs energy, which is defined using *a priori* knowledge about the depth field.

2. 2-D MOTION ESTIMATION AND OBJECT SEGMENTATION

Motion based segmentation is a necessary step for object-based motion analysis. Since both motion estimation and object segmentation are necessary for estimating each other, the methods which perform these two steps individually have limited performance. These two steps can be combined by using Gibbs formulation.

The Gibbs energy function $\mathcal{U}$ which is the exponent of the exponential joint distribution of 2-D motion $\mathcal{D}$, *segmentation* $\mathcal{R}$ and *model failure* $\mathcal{S}$ fields can be written as

$$\mathcal{U}(\mathcal{D},\mathcal{R},\mathcal{S} \mid \mathcal{I}_t,\mathcal{I}_{t+1}) = \mathcal{U}_n + \lambda_m \mathcal{U}_m + \lambda_R \mathcal{U}_R + \lambda_s \mathcal{U}_s \quad (1)$$

in which

$$\mathcal{U}_n = \sum_{\mathbf{x}\in\Lambda} (I_t(\mathbf{x}) - I_{t+1}(\mathbf{x}-\mathbf{D}(\mathbf{x})))^2 (1-S(\mathbf{x})) + S(\mathbf{x})T_s$$

$$\mathcal{U}_m = \sum_{\mathbf{x}\in\Lambda} \sum_{\mathbf{x}_c\in\Lambda} ||\mathbf{D}(\mathbf{x}) - \mathbf{D}(\mathbf{x}_c)||^2 [\delta(R(\mathbf{x}) - R(\mathbf{x}_c))]$$

$$\mathcal{U}_R = \sum_{\mathbf{x}\in\Lambda} \sum_{\mathbf{x}_c\in\Lambda} [1 - \delta(R(\mathbf{x}) - R(\mathbf{x}_c))] + \sum_{\mathbf{x}\in\Lambda} \sum_{\mathbf{x}_c\in\Lambda} \lambda_t \frac{[1-\delta(R(\mathbf{x})-R(\mathbf{x}_c))]}{1+(I_t(\mathbf{x})-I_t(\mathbf{x}_c))^2} + \theta(R(\mathbf{x}))$$

$$\mathcal{U}_s = \sum_{\mathbf{x}\in\Lambda} \sum_{\mathbf{x}_c\in\Lambda} [1 - \delta(S(\mathbf{x}) - S(\mathbf{x}_c))]$$

In the equation above, $I_t(\mathbf{x})$ is the intensity value of the frame I at time t for the location $\mathbf{x}$, where $\mathbf{x}$ and $\mathbf{x}_c$ are neighboring elements on a 2-D grid, Λ, on which the image intensities are defined. $\mathcal{D}$ is the unknown 2-D motion field, which is also defined on Λ. The true 2-D motion vectors are expected to match intensities between I_t and I_{t+1} ($\mathcal{U}_n$ term in $\mathcal{U}$) and have similar values between neighbors except object boundaries ($\mathcal{U}_m$ term in $\mathcal{U}$). $\mathcal{R}$ field is used to segment objects in the scene and prevents $\mathcal{U}_m$ getting a high penalty at motion boundaries. $\mathcal{U}_R$ term supports objects, which

0-8186-7310-9/95 $4.00

have projected broad regions on 2-D image plane with textural coherence. Textural coherence is supported by giving a penalty to different intensity values of neighboring pixels if they do not belong to the same region. Additionally some *taboo patterns*, such as single-point or cross-shaped patterns which are defined on an 8-neighborhood system are rejected by giving a high penalty, using $\theta(R(\mathbf{x}))$ term. $\mathcal{S}$ is a binary field and shows the *model failure regions*, in which the motion compensation error is expected to be greater than a threshold, T_s. Lastly, $\mathcal{U}_s$ term supports $\mathcal{S}$ field to consist of regions, instead of individual points.

By minimizing the energy function, $\mathcal{U}$, a *MAP* estimate of the unknown 2-D motion, segmentation fields and model failure regions can be estimated at the same time.

3. 3-D MOTION ESTIMATION

There are many different approaches to 3-D motion and structure estimation problem [2]. It is shown that the linear algorithms, which are simple, are susceptible to noise, whereas non-linear approaches are computationally costly [3]. Among all, the linear **E**-*matrix* approach [1] has given good results for estimating the global motion of a camera and depth of the stationary environment [3]. However, the application of this approach to object based coding is still unexamined.

Let $\mathbf{X}(t) = (X(t), Y(t), Z(t))$ be the 3-D coordinates of a rotating and translating object. The corresponding coordinates at time instant $t+1$ can be given by the equation

$$\mathbf{X}(t+1) = \mathbf{R}\mathbf{X}(t) + \mathbf{T} \tag{2}$$

where $\mathbf{R}$ is a 3x3 rotation matrix, $\mathbf{T}$ is a 3x1 translation vector. After *perspective projection* of the 3-D object points into 2-D image plane and by some manipulations, the relation below is obtained for the *Essential* (**E**) matrix.

$$\mathbf{U'EU} = 0 \tag{3}$$

where $\mathbf{U} = [x(t) \quad y(t) \quad 1]^T$, $\mathbf{U'} = [x(t+1) \quad y(t+1) \quad 1]^T$. $\mathbf{x}(t) = (x(t), y(t))$ are the 2-D coordinates of the projected object points at time t and $x(t+1)$ and $y(t+1)$ are the corresponding points at time $t+1$. The unknown $\mathbf{E}$ matrix is equal to

$$\mathbf{E} = \begin{bmatrix} 0 & T_z & -T_y \\ -T_z & 0 & T_x \\ T_y & -T_x & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{4}$$

where $T_{x,y,z}$'s are the elements of translation matrix $\mathbf{T}$ and r_{ij}'s are the elements of rotation matrix, $\mathbf{R}$.

By finding some (at least 8) correspondences (or using the estimated $\mathcal{D}$ field), Equation 3 can be solved in the least squares sense and afterwards **E**-matrix can be decomposed into $\mathbf{R}$ and $\mathbf{T}$ analytically [1].

In order to improve the performance of this noise susceptible algorithm, instead of using all the estimates of $\mathcal{D}$ field, "trusted" estimates are chosen by simply thresholding the local energy and image gradient.

4. ESTIMATION OF DEPTH

In order to motion compensate a pixel on 2-D image, the 3-D motion parameters are not sufficient. The depth value of an object point is also necessary to determine the projected 2-D motion of the corresponding pixel on the image plane. Hence for video coding purposes, 3-D structure of the object must be determined as well.

The depth values of each point can be solved linearly, using the estimated 2-D and 3-D motion parameters, and Equation 2 [1]. This gives the depth value, $Z(\mathbf{x})$ at location $\mathbf{x}(t) = (x(t), y(t))$ as

$$Z(\mathbf{x}) = \frac{T_x - x(t+1)T_y}{x(t+1)(r_{31}x(t) + r_{32}y(t) + r_{33}) - (r_{11}x(t) + r_{12}y(t) + r_{13})} \tag{5}$$

However such an approach may give erroneous results, since the linear solution is susceptible to the estimation errors of both 2-D and 3-D motion estimates. Therefore, another Gibbs energy function, which takes into account neighboring relations is written. A similar energy function without any segmentation of the scene, was also proposed to estimate dense depth fields [4].

$$\mathcal{U}(\mathcal{Z} \mid \mathcal{M}, \mathcal{R}, \mathcal{I}_t, \mathcal{I}_{t+1}) = \mathcal{U}_n + \lambda_Z \mathcal{U}_Z \tag{6}$$

where

$$\mathcal{U}_n = \sum_{\mathbf{x} \in \Lambda} (I_t(\mathbf{x}(t)) - I_{t+1}(\mathbf{x}(t+1)))^2$$

$$\mathcal{U}_Z = \sum_{\mathbf{x} \in \Lambda} \sum_{\mathbf{x}_c \in \Lambda} (Z(\mathbf{x}) - Z(\mathbf{x}_c))^2 [\delta(R(\mathbf{x}) - R(\mathbf{x}_c))]$$

In the equation above, $\mathcal{U}_n$ term tries to match the intensity of the moving point on consecutive frames using given 3-D motion parameters and the unknown Z values. $\mathcal{U}_Z$ is the regularization term, which makes the algorithm immune and gives penalty to the differences between neighboring pixel depths of each object, which were segmented by the previous algorithm.

By minimizing the energy function $\mathcal{U}$, the unknown depth field $\mathcal{Z}$ can be estimated robustly using 3-D motion parameters, $\mathcal{M}$. It is important to note that the nonlinear formulation directly takes into account the compensation of intensity values between consecutive

frames and hence the estimated 3-D motion parameters and depth values can be used for video coding purposes without any problems.

5. PROPOSED ALGORITHM

The following algorithm is proposed to estimate 3-D motion and depth values in a complex scene :
1. Obtain segmentation and dense 2-D motion estimates by minimizing Equation 1.
2. Using model failure regions and simple thresholding, eliminate bad 2-D motion estimates (outliers).
3. Using 2-D motion and segmentation estimates, for each object calculate the 3-D motion parameters
4. Using the 3-D motion parameters and segmentation field estimate dense depth field.

6. EXPERIMENTAL WORK

An artificial sequence is tested during the first stage of experiments (Figure 1 (a,b)). Using these two frames, the given energy function (Equation 1) is minimized using a *multiscale constrained minimization* algorithm [5]. A similar minimization method, which is called *hierarchical rigidity* is also proposed independently [6] in order to jointly estimate 3-D motion and structure of a rigid moving body in a hierarchical manner. The results of 2-D motion estimation and segmentation are given in Figure 2. Table 1 shows the mean (among motion vectors on the image) and the variance of the error between the estimated and true 2-D motion vectors.

	μ	σ^2
horizontal comp. of 2D vect	0.18	6.2
vertical comp. of 2D vect	0.66	9.6

Table 1. Mean,μ, ($pixel$) and variance,σ^2, ($pixel^2$) of error for 2-D motion components

By using these estimates, the results of the linear 3-D motion estimation method is given in Table 2 in comparison to correct values.

	θ_x	θ_y	θ_z	T_x	T_y	T_z
Correct	6.00	6.00	6.00	6.00	6.00	6.00
Estimate	5.77	5.86	6.40	7.05	6.63	3.78

Table 2. Estimated 3-D rotation, $\theta_{x,y,z}$ (deg.) and translation $T_{x,y,z}$(pix.) parameters

Using the estimated 3-D values, the depth field is found by using two different methods, one of which is linear (calculate Equation 5) and the other is nonlinear (minimize Equation 6). The results are shown in Figure 3(b) and (c) respectively and Gibbs formulation has better performance in terms of error susceptibility when it is compared to the correct depth value in Figure 3(a).

The second stage of experiments are performed on standard sequences, such as *Salesman* (frames 210 and 220). Predetermined windows (64x64) from both frames (Figure 4(a,b)) are used in order to estimate the 3-D motion and structure in the scene. The results which are obtained using the proposed algorithm in Section 5, are shown in Figure 5. The reconstructed image (Figure 5(f)), which is obtained using the estimated 3-D motion parameters, depth field and previous frame has SNR_{peak} value, which is equal to $32.1dB$.

7. CONCLUSIONS

Bayesian approach successfully segments moving bodies in a scene into some regions and gives good estimates for 2-D projected motion. For each segmented object, 3-D motion estimation is achieved successfully by the available point correspondences. Hence the motion of each object is represented with a few parameters, which is vital for motion compensated coding. However, fitting some parametric surface or wireframe to the estimated dense depth field is necessary from coding point of view and remains to be a future work. It should be also noted that the utilized algorithms are computationally less demanding with respect to similar counterparts and the overall algorithm is a good candidate for real-time applications.

8. REFERENCES

[1] R.Y. Tsai and T.S. Huang "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 13–27, January 1984.

[2] J.K. Aggarwal and N. Nandhakumar "On the Computation of Motion from Image Sequences-A Review," *IEEE Proceedings*, vol. 76, pp. 917–935, August 1988.

[3] J. Weng, N. Ahuja and T.S. Huang "Optimal Motion and Structure Estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 864–884, September 1993.

[4] R. Laganiere and A. Mitiche "Direct Bayesian Interpretation of Visual Motion," in *IMACS Int. Symposium on Singal Processing, Robotics and Neural Networks*, pp. 140–144, 1994.

[5] F. Heitz, P. Perez and P. Bouthemy "Multiscale Minimization of Global Energy Functions in Some Visual Recovery Problems," *CVGIP-Image Understanding*, vol. 59, pp. 125–134, January 1994.

[6] A.A. Alatan and Levent Onural "Gibbs Random Field Model Based 3-D Motion Estimation by Weakened Rigidity," in *Proceedings of IEEE Int. Conf. on Image Processing '94, Austin, November*, pp. II 790–794, 1994.

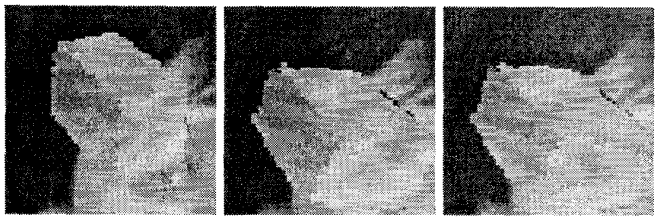

Figure 1: 2 original frames (a),(b) of "Cube" sequence; motion compensated frame (c) using estimated 3-D motion parameters and depth values

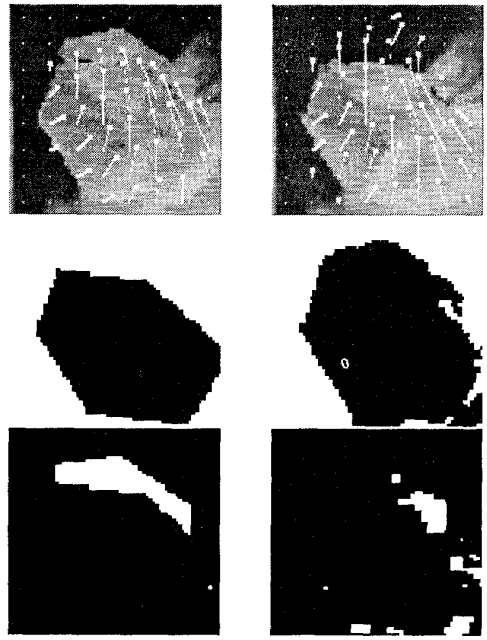

Figure 2: The needlegram of (a) true and (b) estimated 2D motion field,$\mathcal{D}$; (c) True and (d) estimated segmentation field, $\mathcal{R}$; (e) True and (f) estimated Model Failure Regions, $\mathcal{S}$

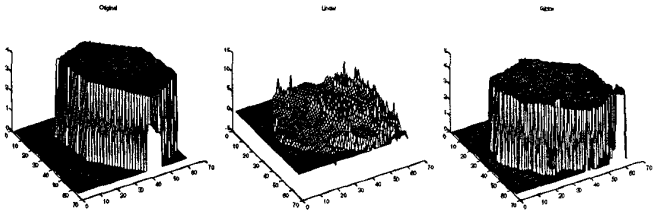

Figure 3: (a) True, and estimated (b,c) depth fields using linear (b) and Gibbs formulated (c) approach

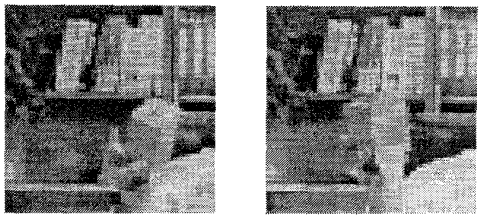

Figure 4: 64x64 portion from the (a)210 (b)220th frames of *Salesman* sequence

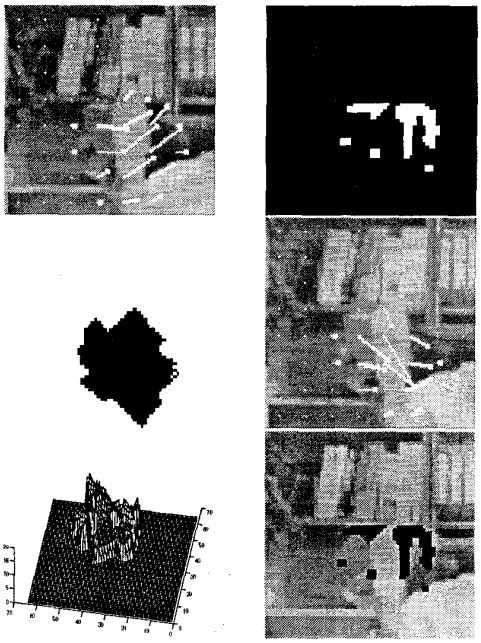

Figure 5: Estimated (a) needlegram of 2-D motion; (b) Model Failure Areas for 2-D motion field; (c) Segmentation field; (d) needlegram of the projected 3-D motion; (e) depth field; (f) Motion compensated frame using 3-D motion parameters and depth field