Active Camera Tracking using Affine Motion Compensation

Young-Kee Jung[a] and Yo-Sung Ho[b]

[a]Honam University, 59-1 Seobong-dong Kwangsan-gu, Kwangju, Korea
[b]K-JIST, 1 Oryong-dong Puk-gu, Kwangju, Korea

ABSTRACT

This paper describes a feature-based tracking system that can track moving objects with an active camera. A robust camera motion estimation algorithm is proposed to obtain a stable global motion for feature tracking. After we identify background and foreground motions based on dominant motion estimates, we estimate camera motion on the background by applying a parametric affine motion model. After compensating for camera motion, we trace multiple corner features in the scene and segment foreground objects by clustering motion trajectories of the corner features. We can also command the pan-tilt controller to position the moving object at the center of the camera.

Keywords: Active Camera, Feature Detection, Feature Tracking, Affine Motion Estimation

1. INTRODUCTION

In this paper, we develop an automated camera system that can watch moving objects in the restricted area with a camera having active pan and tilt control. If objects move outside the field of view, the camera should pan or tilt such that the objects always stay within the filed of view. In those applications, motion estimation and motion tracking are key components.

There have been various research works addressing these areas[1],[2],[3],[4], but it is difficult to design general and robust solutions to the problems involved. This difficulty stems from the complicated relationship between the motion of objects in the 3-D scene and the apparent motion of brightness patterns in the sequence of 2-D projections of the scene. Information about the relative depth of objects is lost in the projection, and the observed motion in the image plane can result from other phenomena than the object motion in the scene, such as changes in the lighting conditions. Moreover, the presence of observation in the 2-D image sequence is in itself a non-trivial task because of the presence of observation noise, occlusions and temporal aliasing. Especially, in case of active camera, because the moving camera creates image changes due to its own motion, object tracking with a mobile camera is very a challenging task.

The work described in this paper attempts to address these problems. First, we utilize an affine model to describe camera motion variation within a sequence. Affine models provide greater flexibility in modelling camera motion, being able to represent rotation, dilation and shear as well as translation. Second, after discriminating between background and foreground motions, camera motion is robustly estimated on the background. Therefore, the camera motion estimate is not disturbed by the presence of outliers due to foreground objects whose motion is not representative of the camera motion.

2. THE PROPOSED ACTIVE CAMERA TRACKING SYSTEM

As shown in Fig. 1, the proposed tracking system consists of five functional parts: background/foreground separation, camera motion estimation, feature detection and tracking, clustering motion trajectories, and control of the pan-tilt camera. The system first identifies background and foreground regions based on dominant motion estimates. Camera motion is then estimated on the background by applying parametric affine motion estimation. After camera motion compensation, we trace many features of moving objects in the scene and cluster the motion trajectories of the feature points by grouping the attributes of the feature trajectories that show similar characteristics. Finally, we command the pan-tilt controller to follow the object such that the object should always lie at the center of the camera.

Further author information: (Send correspondence to Young-Kee Jung)
Young-Kee Jung: E-mail: ykjung@honam.ac.kr, Telephone: 82 62 940 5404
Yo-Sung Ho: E-mail: hoyo@kjist.ac.kr, Telephone: 82 62 970 2211

Visual Communications and Image Processing 2003, Touradj Ebrahimi, Thomas Sikora, Editors, Proceedings of SPIE Vol. 5150 (2003) © 2003 SPIE · 0277-786X/03/$15.00

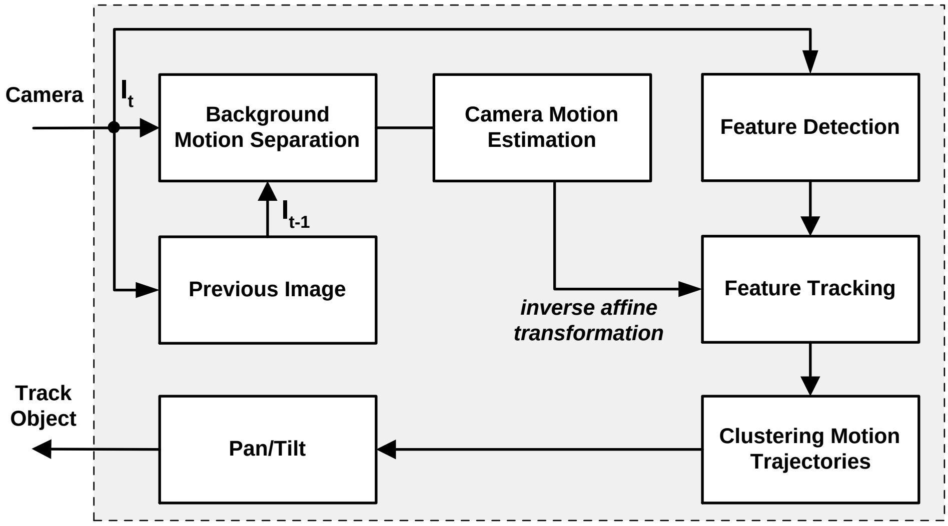


Figure 1. Proposed Object Tracking Algorithm.

2.1. Background Motion Separation

The separation of background is based on block-based motion estimation. The modified block-based estimator is used to track changes of individual blocks and the background motion is classified by using a dominant motion vector.

Each frame of 320x240 resolution is divided into non-overlapping 32x24 blocks. For a block motion estimation, a 9x9 window region having the maximum standard deviation is extracted as the block feature within each block, as shown in Fig. 2. However, in low contrast areas, the resulting motion vectors are unreliable. In order to overcome this problem, we apply the activity criterion to filter out unreliable blocks with lower standard deviation than a certain threshold value. The 9x9 template that was extracted is correlated in the search region. After we locate the correlation peak, a motion vector is associated with each block. The block motion vector holds the displacement of the block between the current and the previous frames.

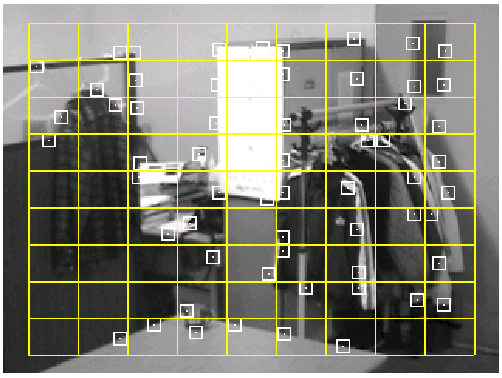


Figure 2. Selected 9x9 Region for Block Motion Estimation.

In order to extract the background motion, we compute a dominant motion by averaging the block motion vectors derived in the previous step. Then, we filter out the noise or foreground motion vectors that have much deviation from the average motion vector.

2.2. Camera Motion Estimation

After discriminating between background motion and the other motions, the camera motion is estimated on the background. In this way, the camera motion estimate is not spoilt by the presence of outliers due to foreground objects whose motion is not representative of the camera motion.

The camera motion is modelled by a parametric affine motion model having six parameters. We first estimate the six parameters using the least square method from the background motion vectors. Once motion parameters are obtained, we compensate the camera motion through the inverse affine motion transformation.

Let (x, y) be a block vector position in the previous frame and (x', y') be the position in the current frame. Then, we can represent the motion vector (v_X, v_Y) by

$$\begin{bmatrix} v_X(x,y) \\ v_Y(x,y) \end{bmatrix} = \begin{bmatrix} x' - x \\ y' - y \end{bmatrix} \tag{1}$$

Since we use the affine motion model of six parameters, the motion vector can be expressed as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_4 & a_5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_3 \\ a_6 \end{bmatrix} \tag{2}$$

In order to estimate six affine motion parameters, we define an error function to be minimized by

$$E(a) = \sum_{i=1}^{N} \{[v_X(x_i, y_i) - v_X(x_i, y_i)]^2 + [v_Y(x_i, y_i) - v_Y(x_i, y_i)]^2\} \tag{3}$$

where N is the number of motion vectors in the same frame.

By substituting Eqs. 2 into Eqs. 3, we have

$$E(a) = \sum_{i=1}^{N} \{[v_X(x_i, y_i) - (a_1 x + a_2 y + a_3)]^2 + [v_Y(x_i, y_i) - (a_1 x + a_2 y + a_3)]^2\} \tag{4}$$

The optimal values of the six parameters are estimated by the least square method. The resulting equation is represented by

$$\sum_{i=1}^{N} \begin{bmatrix} x_i^2 & x_i y_i & x_i & 0 & 0 & 0 \\ x_i y_i & y_i^2 & y_i & 0 & 0 & 0 \\ x_i & y_i & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_i^2 & x_i y_i & x_i \\ 0 & 0 & 0 & x_i y_i & y_i^2 & y_i \\ 0 & 0 & 0 & x_i & y_i & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = \sum_{i=1}^{N} \begin{bmatrix} v_X(x_i, y_i)x \\ v_X(x_i, y_i)y \\ v_X(x_i, y_i) \\ v_Y(x_i, y_i)x \\ v_Y(x_i, y_i)y \\ v_Y(x_i, y_i) \end{bmatrix} \tag{5}$$

2.3. Feature Detection and Tracking

Since the corner feature is viewpoint invariant and naturally leads to the representation of the object shape, corner points are used as features in the scene. For corner point detection, we take gradient operations along the x and y directions over the 9x9 window, and compute the second moment matrix Z by taking average of the gradient values[5],[6].

$$Z = \begin{bmatrix} g_x^2 & g_x g_y \\ g_x g_y & g_y^2 \end{bmatrix} \tag{6}$$

If the matrix Z has two large eigenvalues, the original window contains a corner feature of high spatial frequency. Therefore, we can declare the corner point if $min(\lambda_1, \lambda_2) > \lambda_c$, where λ_1 and λ_2 are two eigenvalues of the matrix Z and λ_c is a predefined threshold value.

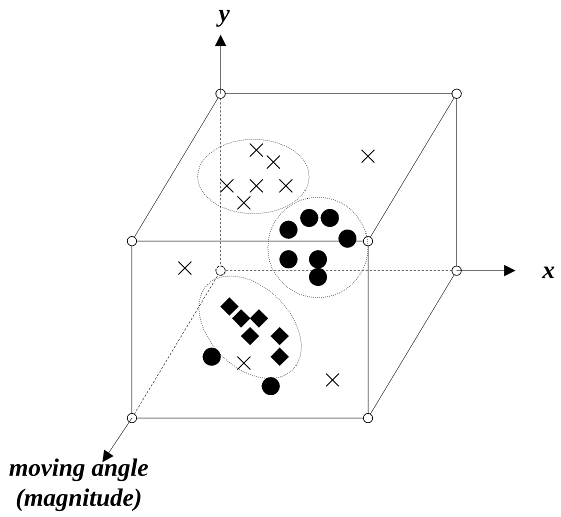


Figure 3. Clustering in the Multi-dimensional Feature Space.

Once a corner point is detected, we compensate the corner position using inverse affine transformation for camera motion and trace the feature efficiently by predicting the next coordinate from the observed coordinate of the feature point. We design a 2D token-based tracking scheme using Kalman filtering[7]. The center position of the feature is used as the token t(k). After we define the system model and the measurement model, we apply the recursive Kalman filtering algorithm to obtain linear minimum variance (LMV) estimates of motion parameters.

2.4. Clustering Motion Trajectories

We separate two heterogeneous motions by grouping the attributes of the corner points according to their spatial and temporal displacements. The key attributes for classifying the global and local motions are the position $C(C_x, C_y)$, the average moving direction A_a, and the average moving magnitude M_a of the corner points. The attribute of each feature is computed by Eqs. 7 and Eqs. 8.

$$\begin{aligned} M_a &= \frac{1}{N}\sum_{i=1}^{N} M_i \\ M_i &= \sqrt{(C_x(i) - C_x(i-1))^2 + (C_y(i) - C_y(i-1))^2} \end{aligned} \tag{7}$$

where i is the time segment, M_i is the moving distance of the corner point at time i, C_x and C_y are the horizontal and vertical positions of the corner point at the current image, and N is the trajectory length.

$$\begin{aligned} A_a &= \frac{1}{N}\sum_{i=1}^{N} A_i \\ A_i &= arctan\frac{C_y(i) - C_y(i-1)}{C_x(i) - C_x(i-1)} \end{aligned} \tag{8}$$

where A_i is the moving direction of corner point at time i.

As shown in Fig. 3, the attributes are arranged in the three-dimensional feature space. We cluster the corner points by grouping the attributes of similar characteristics.

After the dynamic range of each attribute is normalized, we cluster the attributes by the K-means algorithm that is extended to three parameters. The full set U of the corner points q is given by

$$U = \{q_0, q_1, q_2, \cdots, q_n\} \tag{9}$$

We compute the first-order moment from the elements of U and denote it as the initial center $\overrightarrow{m}_0$. If the standard deviation obtained from U and $\overrightarrow{m}_0$ is greater than the predetermined threshold, a new center vector of a cluster $\overrightarrow{m}_1$is determined by

$$\overrightarrow{m}_1 = \overrightarrow{m}_0 + \alpha\sigma_0, \alpha : constant \tag{10}$$

The cluster points are reassigned based on the Euclidean distances, $d(\overrightarrow{m}_0, q_k)$ and $d(\overrightarrow{m}_1, q_k)$, from $\overrightarrow{m}_0$ and $\overrightarrow{m}_1$. The criterion for reassignment of the cluster points is described by

$$\begin{aligned} C_0 &= \{q_k : d(\overrightarrow{m}_0, q_k) \geq d(\overrightarrow{m}_1, q_k)\} \\ C_1 &= \{q_k : d(\overrightarrow{m}_0, q_k) < d(\overrightarrow{m}_1, q_k)\} \; k = 1, 2, 3, \cdots, n \end{aligned} \tag{11}$$

where C_0 and C_1 are cluster numbers, respectively.

Consequently, the sets of elements of new clusters are defined by

$$\begin{aligned} C_0 &= \{q_{00}, q_{01}, q_{02}, \cdots, q_{0x_0}\},\; 1 \leq x_0 < n \\ C_1 &= \{q_{10}, q_{11}, q_{12}, \cdots, q_{1x_1}\},\; 1 \leq x_1 < n \; and \; x_0 + x_1 = n \end{aligned} \tag{12}$$

where $x0$ and $x1$ are numbers of elements in the cluster sets C_0 and C_1, respectively.

After finding the new first moments $\overrightarrow{m}'_0$ and $\overrightarrow{m}'_1$ with elements of the sets C_0 and C_1, we perform the reassignment process for the elements classified before by computing $d(\overrightarrow{m}'_0, q_k)$ and $d(\overrightarrow{m}'_1, q_k)$ for all elements of the set U.

We repeat the process recursively until each standard deviation σ_k is smaller than the specific threshold value.

Eventually, the cluster set C_k comprises all the corner points.

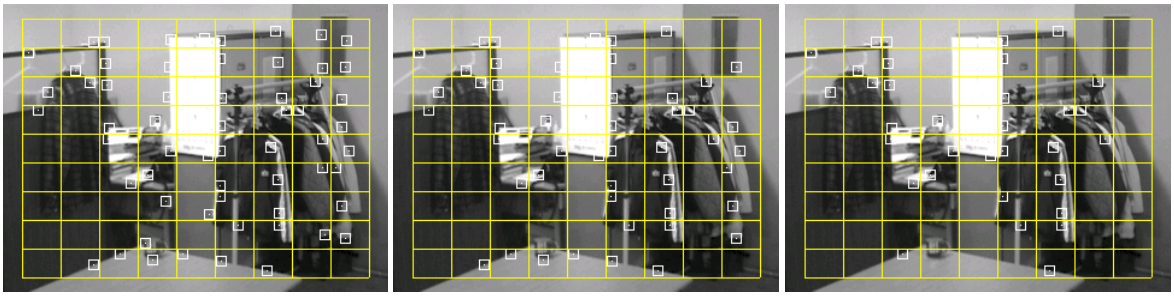
$$C_k = \{q_{k0}, q_{k1}, q_{k2}, \cdots, q_{kx_k}\},\; 1 \leq x_k < n \; and \; x_0 + x_1 + \cdots + x_k = n \tag{13}$$

3. SIMULATION RESULTS

The proposed tracking system has been tested on several video sequences in indoor environments. Fig. 4 shows the block feature selection results for three activity thresholds. A high activity threshold diminishes the number of the block features. We use 35.0 as the threshold value for the tracking system.

Four types of video sequences are captured, as shown in Fig. 5; right-panning and left-moving person, right-panning and right-moving, left-panning and right-moving person, and left-panning and left-moving person. As shown in Fig. 5(a), the right panning of camera makes one motion. A moving person occurs the other motion. The background motion is separated by extracting dominant motion vectors. The center image of Fig. 5(a) displays the results before the camera motion compensation. The results after the global motion compensation are represented in the most right image of Fig. 5(a).

Fig. 6 and Fig. 7 show the tracking results for the scene of the person who moved to the left and the right directions, respectively. As shown in Fig. 6, a number of corners are selected as the active corners. In Fig. 6, we note that there are several feature paths corresponding to the person in the scene. Since the global motion by camera movement is eliminated, the result shows the only local motions of the person. The pan-tilt operation is commanded to move the camera to the centroid of local motion.



(a) (b) (c)

Figure 4. Block Feature Selection for Activity Threshold Values: (a)TH(25.0) (b)TH(35.0) (c)TH(45.0)

4. CONCLUSIONS

In this paper, we proposed an active camera tracking system based on feature-based object tracking. In the proposed system, we estimate the camera motion using the affine motion model and compensate the camera motion using inverse affine transformation and trace the feature efficiently by predicting the next coordinate from the observed coordinate of the feature point. Finally, the local motion trajectories of the corner features that represent the motion coherence property are clustered. In case of a single moving object, the proposed algorithm demonstrates robust tracking results. In the future, we plan to improve our algorithm by applying active zooming and multiple objects tracking.

ACKNOWLEDGMENTS

This work was supported in part by grant NO. R05-2002-000-00868-0 from the Basic Research Program of the Korea Science & Engineering Foundation. This work was supported in part by K-JIST, in part by KOSEF through UFON, and in part by MOE through BK 21.

REFERENCES

1. G. Johansson, "Visual Perception of Biological Motion and a Model for Its Analysis", *Perception and Psychophysics*, Vol.14, pp. 201-211, 1973.

2. K. Gould and M. Shah, "The Trajectory Primal Sketch: A Multi-sale Scheme for Representing Motion Characteristics", *IEEE Conf. of CVPR*, pp. 79-85, 1989.

3. O. Rouke and Badler, "Model-based Image Analysis of Human Motion using Constraint Propagation", *IEEE Trans. on PAMI", Vol.3, pp. 522-537, 1980.*

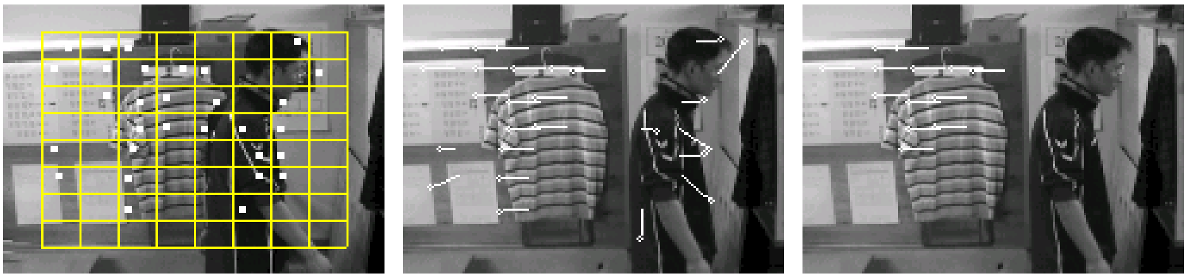
4. K.W. Lee, S.W. Ryu and K.T. Park, "Motion-based Object Tracking with Mobile Camera," *IEE Electronics Letters*, Vol.34, No.3, pp.256-258, Feb. 1998.

5. W. Forstner, E. Gulch, "A Fast Operator for Detection and Precise Location of Distinct Points, Corners, and Centers of Circular of Features," *Proc. of the Intercommission Conf. On Fast Processing of Photogrammetric Data*, pp. 281-305, 1987.

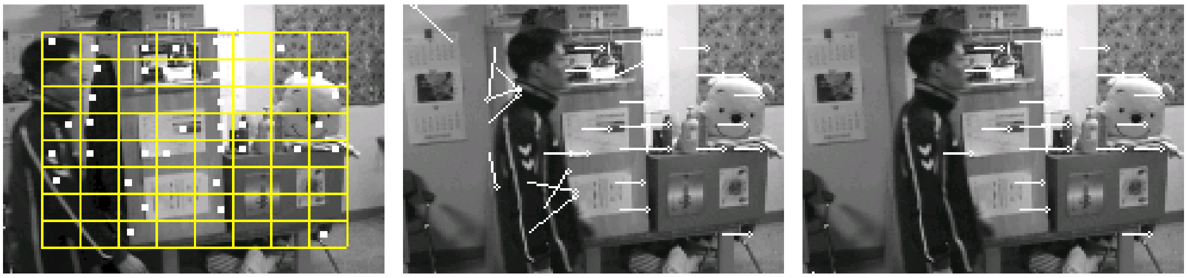
6. D. Beymer, P. McLauchlan, J. Malick, "A Real-time Computer Vision System for Measuring Traffic Parameters, *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 12, pp.495-501, 1997.

7. Y.K. Jung and Y.S. Ho, "Robust Vehicle Detection and Tracking for Traffic Surveillance," *Picture Coding Symposium*, pp.227-230, 1999.

(a)



(b)



(c)

(d)

Figure 5. Background Motion Separation: (a)right-panning and left-moving person, (b) right-panning and right-moving, (c)left-panning and right-moving person, (d)left-panning and left-moving person person

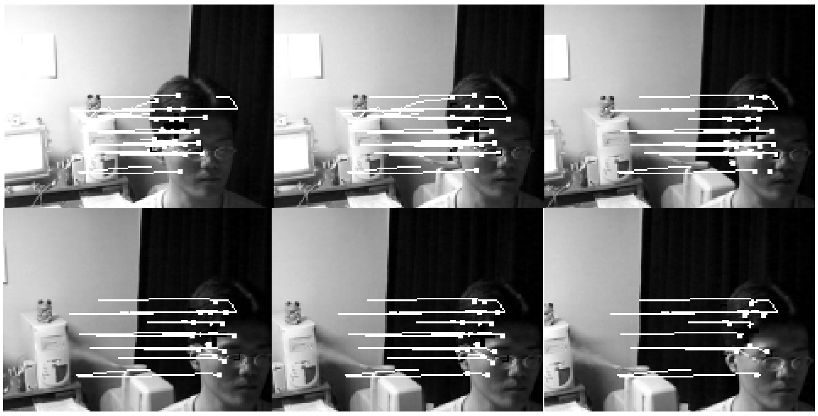


Figure 6. Tracking Results for the Scene of Right Moving Person. Frame numbers (top to bottom, left to right) are 202, 205, 208, 211, 214, 217

Figure 7. Tracking Results for the Scene of Left Moving Person. Frame numbers (top to bottom, left to right) are 244, 247, 250, 253, 256, 259