MVA '96 IAPR Workshop on Machine Vision Applications, November. 12-14, 1996, Tokyo, Japan

Model Based Tracking on 3D Objects

Zheng Li, Han Wang
School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore 639798
E-mail: e94079d52@ntuvax.ntu.ac.sg, hw@ntuix.ntu.ac.sg

Abstract

In this paper, a new tracking algorithm, based on local minimum energy, is proposed for matching between a projected model and corresponding image features in real-time. The algorithm is simple and accurate comparing with known literature. It has added advantage of robustness to changes in lighting or background and requires only a workstation with a frame grabber card installed. By using the proposed algorithm, a real-time motion tracking is performed. In practice, rigid objects with known geometric features are to be tracked in 6 degrees of freedom arbitrary motion.

1 Introduction

Motion tracking can provide more information about the tracked object. Generally, image-based tracking algorithms are employed for 2-Dimensional positioning and recognition. For a rigid object, motion can be estimated from consecutive image sequences by region segmentation[4] or feature point matching[11]. Snakes[7] have been used to track the deformation of a certain image contour. In some special cases, the whole six degrees of freedom are required, while not only the position but also the orientation of tracked objects need to be estimated. One possible and practicable way for this purpose is called model-based tracking which has some prior geometric knowledge of rigid objects being tracked. More researches[1][9][8] and applications[3][6][5] focus on this field in recent years. There are four major stages in each computation cycle for model-based tracking: (1)matching between the projected model and corresponding image features; (2)measurement of the matched differences; (3)motion estimation; (4) model projection. Harris presented a real-time tracking program, RAPID[6], which tracks a known object executing arbitrary motion with a standard video camera. The 3D object model consists of selected control points on edges. While others' works just like Harris' consist of all the above stages. Disadvantages among them are complex computation in low level image processings and nonlinear iteration.

The RAPID algorithm tracks a moving object by finding the best fit between the image and the projection of its model on the image plane. The minimization is carried out by a least squares method and the energy was defined as the discrepancy of the object edge and the model edge. Further, control points along object edges were selected instead of the edge segments. The introduction of control points has the added advantage of reducing dramatically the computation. Therefore, the tracking algorithm is reduced to "best" fit the control points.

In this paper, an approach on 3D model-based tracking is presented. The matching algorithm is based on minimum energy something like snakes[7], which is simpler and more effective than the previous ones by using edge detection and candidate selection. Also a real-time motion tracking is performed by employing all four stages computation.

2 Edge Match Using Minimum Energy

Assuming that at the initial stage, the projected model frame has perfect match with the tracked object. Then the object is moved to a new place in the next frame, the tracking algorithm proceeds to find the match again by adjusting the model's pose. This action is then translated into a search for the new position of the control points. A 1-D search is executed to find high contrast image edge, which is assumed as its new position of the model edge. In the search process, some drawbacks are noted:

1. The control points are treated individually. In fact, some correlations among those control points at the same feature could give more informations about the measurement.

2. Less accuracy in edge matching. The edge search in different orientation could produce result in

different direction. In addition, when the object is subject to substantial rotation, the calculated perpendicular distance differs from the real one to a large extend.

To overcome the above problems, a new approach of feature tracking is proposed and implemented. The feature tracking was based on localization of the local minimum of the energy defined as the negative absolute value of the edge strength.

$$E = -|\nabla(G_\sigma * I(w))|^2 \tag{1}$$

Where G_σ is a 1-dimensional Gaussian of standard deviation σ in the direction orthogonal to the edge.

The image edge has large gradient value compared with the flat intensity region. In ideal case, the intensity distribution is a step at an edge. So the gradient of the intensity is a narrow impulse at the edge and zero otherwise. This situation is not suitable for our energy minimizing search because the zero gradient has no clue for a control point to find smaller energy nearby at current location.

The Gaussian blurring is selected to convolve with the image data. Thus the nonzero scope of the gradient is expanded, centered at the edge, so that the location of a control point has nonzero gradient, which can lead it to the place where the energy is locally minimum. Figure 1. shows the function of Gaussian and how to find the local minimum. As the ball is at the slope surface of energy, it has a trend of rolling down to the extreme bottom. The rolling direction and step may be determined by the image energy gradient. This movement r is expressed as

$$r = -k\nabla(E) \tag{2}$$

Where k is a positive constant. If k is appropriate, equation(2) may lead the control point to a place with minimum energy correctly and rapidly.

The advantages of using the minimum energy tracking are as follows.

- *Robust to light change. Our search strategy is not sensitive to illumination, no selected threshold is necessary for the matching process.*
- *Overcome the defects that edge detection always faces[10]. The early edge detectors, such as Roberts, Sobel and Prewitt, are very sensitive to noise when using derivatives, and need appropriate thresholds to yield a binary edge map. The improved detectors are based on the detection of extrema in the output of the convolution of the image with an impulse response to be determined by considering three criteria as (1) good detection (2) good localization and (3) uniqueness of response. However, there are still some parameters to be estimated according to the signal-to-noise ratio[2].*
- *Effective and reduced complexity. Only 1-Dimensional search of each control point for matching is enough for feature tracking. In algorithms of [9][8][1][3], both global and local edge detection needs whole range searching, and for the detected edges, determination must be done to assure which one is the best matching for a certain projected model edge.*
- *Redundancy, able to recover the model pose ever through some feature points are lost temporarily.*
- *More accurate matching than that of Harris[6] whose matches are independent and random selected from four fixed directions.*

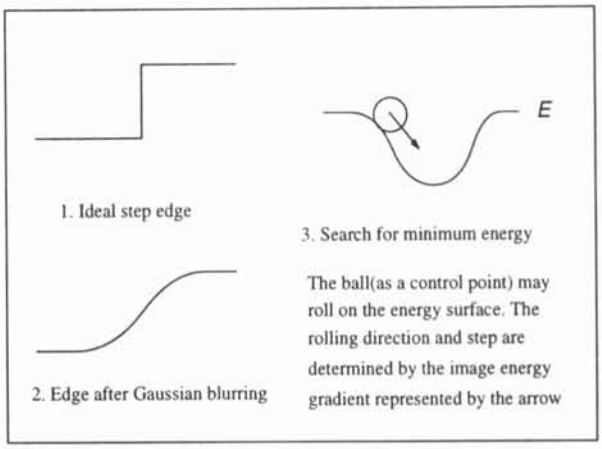

Figure 1: Function of Gaussian for finding local minimum energy.

Since the control points on a certain feature(edge) are aligned, they should still be at the same straight line on the image plane under the perspective projection. The edge detection procedure is shown in Figure 2. The control points $P_1, P_2, ...P_n$ on the same edge form a straight line on the image called control edge. The motion is assumed small between the two consecutive frames, there exists some difference between the control edge and the image edge, which can be minimized in order that the model is approximate to the object in 3D space. The measurement of the difference is the perpendicular distance from a control point to the corresponding image edge orthogonal to the control edge. The points to be matched on the image edge are $E_1, E_2, ...E_n$.The search line passes through a control point with the orientation perpendicular to the control edge. The Gaussian blurring and the minimum energy searching is along the search line. For each control point, the intersection of its search line and the image edge is used to calculate the perpendicular distance.

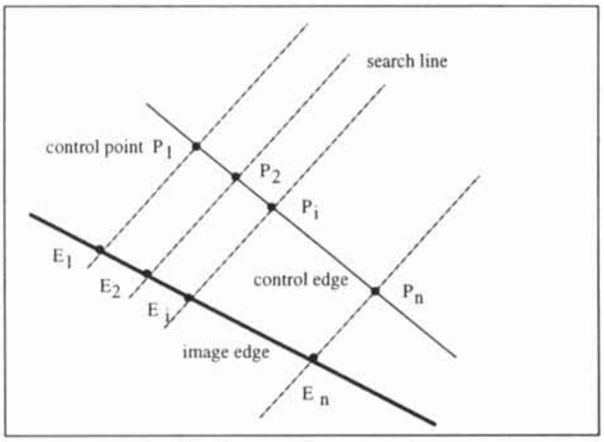

Figure 2: Detection of edge by minimizing image energy along search line.

3 Tracking Performance

As Known that there are four stages in tracking performance. The matching process is discussed in last section, the following computations: errors measurement, motion estimation and model projection are demonstrated in the section.

3.1 Error Measurement

The perpendicular distance can be calculated from a control point and its corresponding matched image edge.

$$l = k_y d_y cos\theta - k_x d_x sin\theta \tag{3}$$

Where k_x, k_y are the dimensions of a pixel in the x and y directions respectively. d_x, d_y are the distances from control point to corresponding matched point in pixels in the image plane orthogonally, θ is the angle from x axes positively to the projected model edge.

3.2 Motion Estimation

The pose estimation requires linear least squares based. We now use the linear approximate from Harris[6]. Consider rotating the model about the model origin by small angle θ, and translating it by a small distance $\mathbf{\Delta}$. A "six-vector", $\mathbf{q}$ represents these two small displacement. If there are N valid mathes and l_i is the corresponding measurement of the perpendicular distance, C_i is a 6×6 matrix donating the coefficients of the normal equations of first order approximation from an arbitrary motion. Then we have

$$\sum_{i=1}^{N} l_i C_i = -\sum_{i=1}^{N} C_i C_i^T \mathbf{q} \tag{4}$$

This is a set of 6 simultaneous linear equations, and so can be solved by using standard linear algebra. The pose change, $\mathbf{q} = (\theta, \mathbf{\Delta})^T$, then must be used to update the model's pose. The rotation and translation will be treated differently, due to changes in model coordinates and camera coordinates, respectively.

The translation can be updated directly in the camera coordinates

$$\mathbf{c} := \mathbf{c} + \mathbf{\Delta} \tag{5}$$

But for rotation, situation is more complex. Because of the nonlinear transformation for rotation, an appropriate representation for rotation update is needed.

Although many ways can be used for rotation representation, only quaternions and orthonormal rotation matrix are suitable for tracking use[1]. The orthonormal rotation matrix R_x, R_y and R_z can be calculated from the estimation θ. Then the pose update on rotation in the model coordinates is

$$\mathbf{m} := R_x R_y R_z \mathbf{m} \tag{6}$$

3.3 Model Projection

After updates, the model should be projected onto the image plane for use in next computation cycle.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{k_x \cdot Z} & 0 & 0 \\ 0 & \frac{1}{k_y \cdot Z} & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{7}$$

Where x, y denote the image coordinates, k_x, k_y are defined in section 2, X, Y, Z stand for the 3D coordinates in camera coordinates.

The object being tracked is something like a wedge, shown in Figure 3. Hidden features always occur from a single viewpoint. The hidden features are no use for tracking, on the contrary, the measurement based on them will cause incorrect pose estimation. So after each model projection, invisible features must be detected. A simple and effective way for elinimating hidden features is employed by computing the orientation of the surface normal of the model in 3D space.

4 Implementation Results

The real time performance achieves on a Sun sparc-20 workstation with an installed frame grabber s2200. A wedge liked model with selected control points on the edges is tracked with an arbitrary motion in 6 degrees of freedom. A selected tracking sequence is displayed in Figure 3. The tracked object has both translation and rotation in front of the camera. The updated model is superimposed on the image with white lines under the perspective projection. From Figure 3, the aimed object is well tracked by using our algorithm.

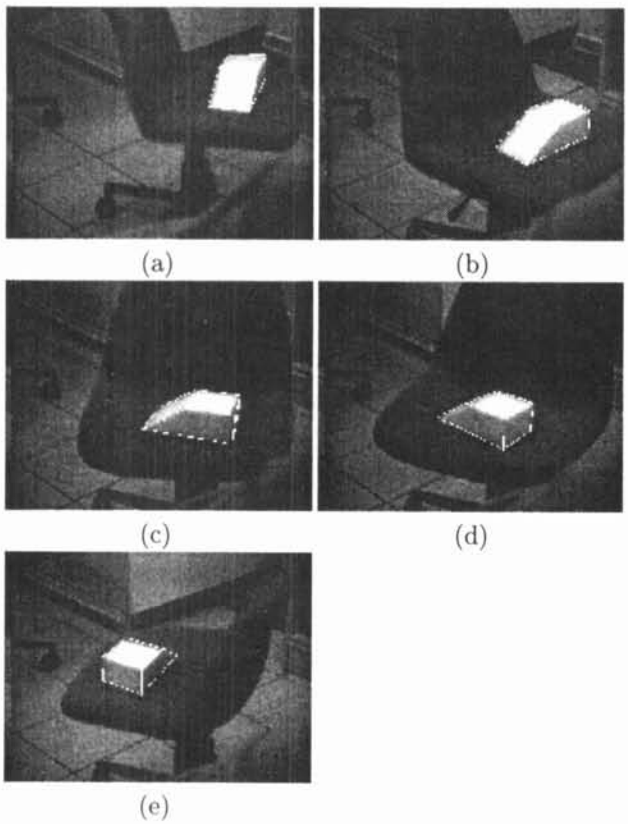

Figure 3: Snap shots of the real time tracking. The wedge model is predefined in the database and the tracking is performed based on the projection of the model matching against the sensed image with defined control points.

5 Conclusion

From the discussion above, our tracking algorithm is efficient and robust for tracking purpose. The matching strategy based on local minimum energy completes the tasks which are made up of edge detection and feature matching in most previous works. These two parts are more complex and time consuming compared with ours. At the same time, the energy-based search is more accurate than that of Harris[6], the practical real time tracking algorithm. Because the energy defined in our approach is not sensitive to illumination, the algorithm can work even under very low light. The result shows that the energy minimizing search gives the pose estimate accurately and stably.

References

[1] Donald B.Gennery. Visual tracking of known 3D objects. *International Journal on Computer Vision*, 7(6):243–270, 1992.

[2] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[3] D.Koller, K.Daniilidis, T.Thorhallson, and H.-H.Nagel. Model-based object tracking in traffic scenes. In *Proc. of 2nd European Conference on Computer Vision*, pages 437–452, 1992.

[4] Duncan.J.H and Chou.T.C. On the detection of motion and computation of optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-14(3):346–352, 1992.

[5] R.J Evans. Kalman filtering of pose estimates generated by the rapid tracker in application. In *Proc. of the British Machine Vision Conference*, pages 79–84, Oxford, UK, September 1990.

[6] C G Harris. Tracking with rigid models. In Andrew Blake and Alan Yuille, editors, *Active Vision*, pages 59–73. MIT Press, 1992.

[7] M Kass, A Witkin, and D Terzopoulos. Snakes: active contour models. In *Proc. of 1st Int. Conf. on Computer Vision*, pages 259–268, London, 1987.

[8] David.G Lowe. Integrated treatment of matching and measurement errors for robust model-based motion tracking. In *Proc. of Int. Conf. on Computer Vision*, pages 436–440, 1990.

[9] David.G Lowe. Robust model-based motion tracking through the integration of search and estimation. *International Journal on Computer Vision*, 8(2):113–122, 1992.

[10] O.Faugeras. *Three Dimensional Computer Vision*. The MIT Press., London, England, 1993.

[11] Han Wang and Wee-Leng Goh. Real time object tracking based on affine transformation. In *Proc. of 2nd. Asian conf. on Computer Vision*, volume 1, pages 39–43, Singapore, 1995.