

Tracking Objects with Rigid Body Templates: An Iterative Constrained Linear Least Squares Approach

Satarupa Mukherjee¹, Nilanjan Ray¹, and Dipti Prasad Mukherjee²

¹ Department of Computing Science
University of Alberta, Edmonton, T6G2E8, AB, Canada

² Electronics and Communication Sciences Unit
Indian Statistical Institute, Kolkata
{satarupa,nray1}@ualberta.ca, dipti@isical.ac.in

Abstract. We formulate a novel iterative, constrained least squares technique for tracking rigid bodies. With barycentric representation of objects, we compute rigid body transformations under optical flow as iterative solutions of the optical flow constraint equation with a homogeneous, linear constraint. We show the efficacy of our method on cluttered videos.

Keywords: optical flow, constrained least squares, rigid body tracking.

1 Introduction

Rigid body tracking has many applications in image analysis. These include visual servoing of robotic arms, traffic monitoring and also medical imaging. There are different techniques available in the literature for rigid body tracking [3,8]. For rigid body tracking from a video, a popular approach is a two-step process [3,8]. The first step detects features or computes motion vectors. Various point features (e.g., corners), or structural features (e.g., edges) can be computed for this purpose. For motion vector computation, a common choice is optical flow. In the second step, a template is fitted to the features or motion vectors. A widely used mathematical tool for the second step is the linear least squares technique [8].

In this paper, we focus on rigid body tracking with the use of optical flow. There are different methods of optical flow computations available to date [2,9,10]. One significant disadvantage of these methods is that they lack prior motion information (e.g., rigid body motion). Thus, when used in tracking, these methods create a two-step pipeline, as mentioned earlier. The inherent difficulty with the family of two-step methods here is that parameters of the optical flow computation must be tuned properly to obtain meaningful tracking.

We propose a novel iterative and constrained linear least squares approach for tracking objects with rigid body motion. The principal and practical advantages of our method are that unlike the two-step tracking methods, there is

no user tuning parameter in our technique and it is fast. Also, our method is able to handle partial occlusions because of the rigid body motion constraint. In our proposed method, the object template is represented in an affine coordinate system (also known as barycentric representation). Then, we solve optical flow constraint equation with rigid body constraints. The affine coordinate representation of the object allows us to map this problem to an iterative, constrained linear least squares approach. Thus, the process is a one-step process, and we never explicitly compute optical flow. The computation only involves solving linear equations with homogeneous linear constraints. So, standard least squares solvers are applicable here. As a challenging application, a publicly available surgical tool video [12] has been considered, where the surgical tool is to be tracked in a cluttered environment. The proposed method has been compared with another one-step optical flow based method using subspace constraint [7], a two-step optical flow method [2] and also with the exhaustive search method. It is observed that the proposed method and the exhaustive search method are most consistent in tracking the tool throughout the video sequences, whereas the other two methods failed to track the tool when it underwent sudden large motions. The record of the running times of all the methods show that the proposed method is an order of magnitude faster than its competitors. Our algorithm is also tested on six different vehicle videos having clutter and partial occlusions.

2 Optical Flow and Two-Step Tracking

The motion between two image frames is calculated using the optical flow method described as follows. A pixel at location (x, y, t) with intensity $I(x, y, t)$ is assumed to move by $\delta x, \delta y$ within a small time period δt between the two image frames. As the image intensity of the pixel is assumed to remain constant, it can be written as [4]:

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t). \quad (1)$$

The movement of the pixel is assumed to be small. So, expanding the right hand side of (1) with Taylor series, we obtain:

$$I_x \dot{x} + I_y \dot{y} = -I_t, \quad (2)$$

where \dot{x} and \dot{y} are respectively the horizontal and the vertical velocities at pixel location (x, y) ; I_x, I_y and I_t are respectively the derivatives of the image in x, y and time directions. This is an equation with two unknown variables \dot{x} and \dot{y} . Hence, it is an underdetermined system. This is known as the ‘‘aperture problem.’’ Optical flow methods usually solve this problem by different regularization techniques, such as adding equations arising out of the assumptions about the smoothness of the flow.

The first step of a two-step tracking procedure computes optical flow in the region of interest typically using smoothness of the flow along with equation (2). We denote by (u_i, v_i) the displacement vector computed by optical flow at the

i^{th} pixel location. In the second step, rigid body motion of an object template is computed using the displacement vectors (u_i, v_i) . Let (x_i, y_i) denote the i^{th} pixel coordinates in the object template. In order to compute the parameters of the rigid body motion, centering is applied first:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i, p_i = x_i - \bar{x}, q_i = y_i - \bar{y}, \text{ for } i = 1 \dots N, \quad (3)$$

where N is the number of pixels that belong to the object template. Next, the rigid body transformation parameters a, b, c, d need to be solved from the following over-determined linear system:

$$\begin{bmatrix} p_i + u_i \\ q_i + v_i \end{bmatrix} = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} p_i \\ q_i \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix}, \text{ for } i = 1 \dots N, \text{ such that } a^2 + b^2 = 1. \quad (4)$$

First, we solve for c and d :

$$c = \frac{1}{N} \sum_{i=1}^N u_i, d = \frac{1}{N} \sum_{i=1}^N v_i. \quad (5)$$

From (4) and (5), we obtain :

$$\begin{bmatrix} p_i + u_i - c \\ q_i + v_i - d \end{bmatrix} = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} p_i \\ q_i \end{bmatrix}, \text{ such that } a^2 + b^2 = 1. \quad (6)$$

We solve for a and b from equation (6) with constrained least square minimization [6].

3 Proposed Method

Object tracking has been illustrated here as an application of affine combination [11]. The proposed tracking framework takes into account motion (affine/rigid body) and texture in the dense optical flow computation.

3.1 Affine Combination and Pixel Correspondence

If X (2-by-1 column vector) denotes a pixel location, then the 2D affine transformation of X is as follows: $Y = AX + f$, where A is a 2-by-2 matrix and f is a 2-by-1 column vector; A and f together represent a 2D affine transformation. X and Y are known as the corresponding point pair. For a set of points $X_1, X_2, X_3, \dots, X_n$, a linear combination: $\sum_{i=1}^n \alpha_i X_i$ is called affine combination when scalar coefficients sum up to unity : $\sum_{i=1}^n \alpha_i = 1$ [11]. It is easy to establish that for corresponding point pairs (X_i, Y_i) , the affine combinations $\sum_{i=1}^n \alpha_i X_i$ and $\sum_{i=1}^n \alpha_i Y_i$ are also corresponding points under the same affine transformation [11]. In other words, the affine combination parameters (α_i) are invariant under affine transformations. Since a rigid body transformation is a special type of affine transformation, the aforementioned invariance holds as well with a rigid body motion. We use this fact in our formulation for rigid body tracking.

3.2 Object Tracking Using Affine Combination

Let O denote the set of pixel locations belonging to the object to be tracked. This is obtained via segmentation (manual, semi-automatic, or automatic) on the initial video frame. Let us choose three corner points of the bounding box of O and denote them by X_1 , X_2 , and X_3 . The set of parameters related with the affine combination of X_1 , X_2 , and X_3 is denoted by: $S = \{(s_1, s_2) : X_1 + s_1(X_2 - X_1) + s_2(X_3 - X_1) \in O\}$. Since the set S is invariant under rigid body transformations, tracking the object is equivalent to tracking the corner locations X_1 , X_2 , and X_3 over the image sequence. Notice that any three non-collinear points X_1 , X_2 , and X_3 can be utilized for our tracking framework. Let (\dot{x}_k, \dot{y}_k) , $k = 1, 2, 3$, denote the velocity of the k^{th} corner X_k , then using (2), for any $(s_1, s_2) \in S$, we can write:

$$[(1 - s_1 - s_2)I_x \ s_1 I_x \ s_2 I_x \ (1 - s_1 - s_2)I_y \ s_1 I_y \ s_2 I_y][\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{y}_1 \ \dot{y}_2 \ \dot{y}_3]^T = -I_t, \forall (s_1, s_2) \in S. \quad (7)$$

Equation (7) has six unknown variables: $\dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{y}_1, \dot{y}_2, \dot{y}_3$. Solutions are found by the least squares method. Note these variables only provide the velocities at the template corners. The corner coordinates are updated as follows. Let the k^{th} corner have coordinates, i.e., $X_k = (x_k, y_k)$. After solving (7), the corner coordinates are updated by a line search [5] in the direction of the velocities:

$$[x_1 \ x_2 \ x_3 \ y_1 \ y_2 \ y_3]^T \leftarrow [x_1 \ x_2 \ x_3 \ y_1 \ y_2 \ y_3]^T + a[\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{y}_1 \ \dot{y}_2 \ \dot{y}_3], \quad (8)$$

where

$$a = \underset{b \geq 0}{\operatorname{argmin}} \sum_{(s_1, s_2) \in S} [I((1 - s_1 - s_2)(x_1 + b\dot{x}_1) + s_1(x_2 + b\dot{x}_2) + s_2(x_3 + b\dot{x}_3), (1 - s_1 - s_2)(y_1 + b\dot{y}_1) + s_1(y_2 + b\dot{y}_2) + s_2(y_3 + b\dot{y}_3)) - g(s_1, s_2)]^2. \quad (9)$$

$g(s_1, s_2)$ is a template function for the object. It is assigned the image intensity of the initial frame, i.e., the frame where tracking begins:

$$g(s_1, s_2) = I_0((1 - s_1 - s_2)x_1^0 + s_1x_2^0 + s_2x_3^0, (1 - s_1 - s_2)y_1^0 + s_1y_2^0 + s_2y_3^0). \quad (10)$$

The superscript 0 denotes the corners of the object on the initial frame I_0 . The line search illustrated in (8)-(9) compares the object template intensity profile $g(s_1, s_2)$ with the current frame only along the direction of the computed velocity $[\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{y}_1 \ \dot{y}_2 \ \dot{y}_3]^T$. So, while the least squares technique computes the direction of displacement, the line search technique complements it with the magnitude of the displacement. Notice that because of the line search our technique is much less dependent on the multi-resolution pyramid computations that are almost always a strict requirement in the traditional optical flow computations.

3.3 Tracking with Rigid Body Constraints

If the object is assumed to have a rigid motion, then our aforementioned computational framework can accommodate such motions easily. The constraints imposed here are that the lengths between any two corner points for the object template remain constant, *i.e.*,

$$\begin{aligned} (x_1 - x_2)^2 + (y_1 - y_2)^2 &= C_1, \\ (x_2 - x_3)^2 + (y_2 - y_3)^2 &= C_2, \\ (x_3 - x_1)^2 + (y_3 - y_1)^2 &= C_3. \end{aligned} \quad (11)$$

When the above three equations are differentiated with respect to time and collected in the matrix vector form, the following equation is formed:

$$\begin{bmatrix} (x_1 - x_2) & (x_2 - x_1) & 0 & (y_1 - y_2) & (y_2 - y_1) & 0 \\ 0 & (x_2 - x_3) & (x_3 - x_2) & 0 & (y_2 - y_3) & (y_3 - y_2) \\ (x_1 - x_3) & 0 & (x_3 - x_1) & (y_1 - y_3) & 0 & (y_3 - y_1) \end{bmatrix} [\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{y}_1 \ \dot{y}_2 \ \dot{y}_3]^T = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (12)$$

Thus, for a rigid body tracking, we solve (7) with the homogeneous linear constraint (12). This is our proposed constrained linear least squares solution for the rigid body tracking problem. Notice that for a rigid body motion constraint, at least three equations as in (12) are required.

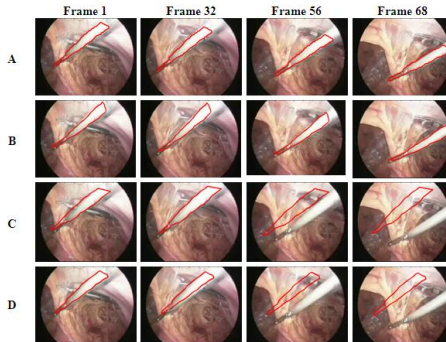


Fig. 1. Tracking results on tool sequence A. Proposed algorithm B. ES1 C. Irani's method [7] D. Brox optical flow method [2].

4 Results and Discussion

The proposed algorithm has been tested on 120 frames of a surgical tool video for tracking tool [12]. This video has significant amount of clutter, which makes

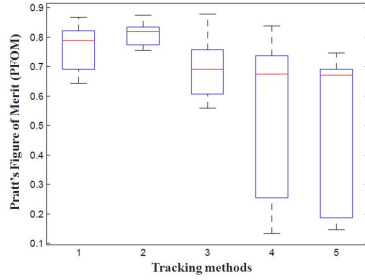


Fig. 2. Accuracy plot of four tracking methods : 1. Prpposed Method 2. ES1 3. ES2 4. Irani's Method [7] 5. Brox et al's Method [2]

tracking a challenging task. Our algorithm is able to track the tool successfully in all the subsequent 120 frames in spite of severe clutter. Figure 1, row A shows various frames and our proposed rigid body tracker. For implementing the proposed algorithm, we choose Matlab's lsqin function. To illustrate the accuracy of the algorithm Pratt's Figure of Merit (PFOM) [1] has been used and accuracies of segmentation for all the frames have been represented as a boxplot in Figure 2. The average time taken by our algorithms on an image frame with our Matlab implementation has been illustrated in Table 1, which shows that the proposed method is the fastest among its competitors.

The proposed method is compared with three other methods: exhaustive search method, a one-step method using subspace constraints [7] and a two-step method using optical flow computation by Brox et al.'s method [2]. Figure 1, rows B through D show the performances of these methods on various frames of the surgical tool video sequence. These competing methods are described here.

We use two different settings for the exhaustive search. The first setting uses unit spacing in horizontal, vertical and angular directions and is referred to as "ES1" method. It can be observed that although ES1 performs better in terms of accuracy Figure 2, it is about two orders of magnitude slower than our proposed algorithm (Table 1). So, it can be concluded that a better accuracy can be achieved by the exhaustive search algorithm only at the cost of running time. When we use a spacing of 4 for the horizontal, vertical and angular directions, the performance of tracking becomes inferior as Figure 2 shows. This method is referred to as "ES2." Note that the computation time of ES2 is still much inferior to the proposed method (Table 1).

The proposed method has been also compared with another well established optical flow based method [7], which has been developed on rigid body scenarios on the assumption that the set of all displacements of all points across several frames often resides in a low dimensional linear subspace. The main idea of this algorithm was to extend the two-frame Lucas-Kanade [10] region based constraint as well as the two-frame brightness constancy constraint to a generalized multi-frame method using the subspace constraints. Results show that our method is more competitive with respect to both accuracy and time (Figure 1, Figure 2, Table 1).

Table 1. Running time comparison

Tracking Algorithms	Time taken between two consecutive frames (s)
Proposed algorithm	0.3
ES1	17.79
ES2	4.52
Irani's method	9.52
Brox et al.	4.21

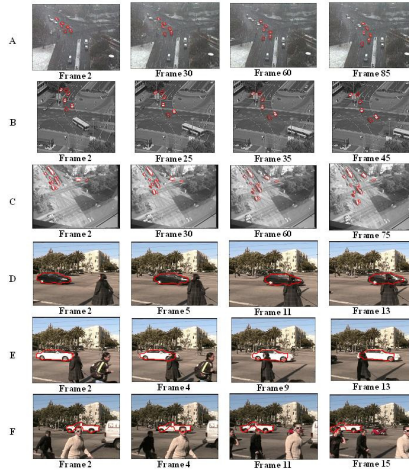


Fig. 3. Tracking results on vehicle sequences A. Car sequence in snowfall B. Car sequence (leftmost car having partial occlusion) C. Vehicle sequence having different types of vehicles D,E,F. Vehicles moving on a busy street occluded by pedestrians.

Finally, the proposed method has been compared with a two-step optical flow tracking [2], using rigid body constraints explained in Section 2. This optical flow method was implemented using multiresolution pyramid. This method needed some parameter tunings like the pyramid level for the multiresolution technique. In comparison, our algorithm did not need any parameter tuning. For the two-step algorithm, a 6 level pyramid with 1.25 resolution spacing has been considered. While for the proposed tracking method occurred at the original resolution. It is observed from Figure 2 that the two-step method has much inferior accuracy than the proposed method. Running times in Table 1 show that the two-step method is at least an order of magnitude slower than the proposed method.

In Figure 1, the tracking results of our algorithm with rigid body constraints have been compared with the results of the three other compared methods. It can be observed that from frame 56, when the tool underwent a large motion, Irani's method and the two step optical flow method failed to track it while our algorithm and the exhaustive search algorithm were able to keep track of it

correctly. A boxplot illustrating the PFOM values for all the four algorithms has been shown in Figure 2, which illustrates that the proposed algorithm is slightly less in accuracy than the exhaustive search, but it outperforms exhaustive search in running time which is shown in Table 1.

The visual results of tracking cars with the proposed method in different videos have been shown in Figure 3. The first row has shown the tracking results of a car sequence in a heavily cluttered video with snowfall. The second row has illustrated the tracking results of another car sequence where the left most car has been successfully tracked in spite of being partially occluded by a post on its way. The third row has demonstrated the tracking results of a sequence including different types of vehicles very small in size. Finally, fourth, fifth and sixth rows show the results from a publicly available dataset [13] where the vehicles are moving on a busy street and they are often occluded by pedestrians passing by. The video outputs and Matlab codes of our techniques are available here: <http://webdocs.cs.ualberta.ca/~satarupa/tracking.html>.

References

1. Abdou, I.E., Pratt, W.K.: Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proc. IEEE* 67, 753–763 (1979)
2. Brox, T., Bruhn, A., Papenberger, N., Weickert, J.: High Accuracy Optical Flow Estimation Based on a Theory for Warping. In: Pajdla, T., Matas, J.(G.) (eds.) *ECCV 2004*. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
3. Drummond, T., Cipolla, R.: Real-time visual tracking of complex structures. *IEEE Trans. Pattern Anal. Machine Intell.* 24, 932–946 (2002)
4. Fleet, D.J., Weiss, Y.: Optical Flow Estimation. In: *Handbook of Mathematical Models in Computer Vision*. Springer (2006)
5. Fletcher, R.: *Practical Methods of Optimization*. John Wiley and Sons, West Sussex (2006)
6. Golub, G.H., Van-Loan, C.F.: Constrained least squares. In: *Matrix Computations*. JHU Press, Baltimore (1996)
7. Irani, M.: Multi-Frame Correspondence Estimation Using Subspace Constraints. *IJCV* 48(3), 173–194 (2002)
8. Lipetit, V., Fua, T.: Monocular model based 3D tracking of rigid objects: a survey. *Foundations & Trends in Comp. Graph. & Vis.* 1(1), 1–89 (2005)
9. Horn, B.K.P., Schunck, B.G.: Determining optical flow. *AI* 17, 185–203 (1981)
10. Lucas, B.D., Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. *IJCAI*, 674–689 (1981)
11. Ray, N.: Image registration and object tracking via affine combination. In: *ICAPR*, pp. 175–179 (2007)
12. http://www.dailymotion.com/video/x6if7d_less-price-ventral-hernia-surgery-i.tech
13. <http://www.eetimes.com/design/military-aerospace-design/4017685/Background-subtraction-part-1-MATLAB-models>