Robust Real-time 3D Trajectory Tracking Algorithms for Visual Tracking Using Weak Perspective Projection

Wei Guan Yau¹, Li-Chen Fu^{1,2}, and David Liu¹

Department of Electrical Engineering¹

Department of Computer Science and Information Engineering²

National Taiwan University, Taipei, Taiwan, R.O.C.

Abstract

In this paper, motion estimation algorithms for the most general tracking situation are developed. The proposed motion estimation algorithms are used to predict the location of target and then to generate a feasible control input so as to keep the target stationary in the center of image. The work differs from the previous algorithm of motion estimation in that it is capable to decouple the estimation of motion from the estimation of structure. The weak perspective projection is used to solve this problem. The modified optical flow is first calculated and then fed to motion estimation algorithms so as to generate an appropriate camera motion that achieve tracking. The important contribution of this work is that simple, numerically stable, none computation intensive, and correspondence-free 3D motion estimation algorithms are derived. A visual tracking system can be easily implemented and run in real-time due to the simplicity of the proposed algorithms and thus increases their efficiency. The robustness and feasibility of the proposed algorithms has been validated by a number of experiments.

1. INTRODUCTION

Understanding the object motion is crucial to achieve the purpose of tracking. With a good knowledge of the object motion, one can improve the performance of tracking and thus increase its potential applications in machine vision systems. In this paper, we focus on the trajectory tracking task in general visual tracking system. Several experiments reported in [20], [13] revealed that their goal is only to pursue a moving target. However, the tracking problem to be solved in this research is not only to keep track of a moving target but also to locate the target at the image center of camera. Thus, the tracking problem to be investigated here is more challenging. In addition, the mathematical theory devised in this research is suitable to the most general tracking situation, i.e., three-dimensional (3D) target moving in three-dimensional space.

As we know, trajectory tracking or visual tracking is not a new research topic nowadays. Recently, several algorithms have emerged to handle this problem. In particular, the Interacting Multiple Model (IMM) algorithm for target tracking has received a lot of attention in [11], [6], [9]. The IMM algorithm has been proved a very efficient and high performance algorithm for tracking the target [9]. However, several models are needed to account for the whole range target motion, which requires

considerable computational burden [18], [12]. In addition, appropriately determine the number of models is crucial since increasing the number of models does not necessarily guarantee better performance [28].

For visual servoing in robotics, Hashimoto and Kimura have proposed an image-based or feature based method [29], [30], Espiau has developed the differential geometry approach [21], and Papanikolopoulos has introduced the adaptive control method [19]. Application of adaptive control in visual tracking is presented in [19], [14], [23], and [24]. As described in [19], the disadvantage of this approach is that the error will become significant when abrupt changes of the trajectories occur since the regulator cannot compensate for the abrupt changes fast enough.

Aloimonos and Tsakiris [3] have described a method which decomposes the tracking problem into two steps. First, they estimate the motion parameters of the object. Second, they solve the minimization problem by using the estimation of the target motion parameter and then derive the desired rotational velocities of the camera that will achieve tracking. Their method however is based on the knowledge of the target shape and is subject to numerical instability problem.

The methods described above are in complicated form or have their own drawbacks, which in turn increase the computational burden or deteriorates the tracking performance. Therefore, this paper proposes another close form solution for estimating the object motion during the tracking phase. This yields better performance than the previous methods. Recently, researches on the relationship between structure and motion have been widely conducted and have resulted in a variety of schemes for reconstructing motion for known structure [26], structure from motion [27], or both motion and structure [5], [1]. Some researches on 3D space curves can be found in [22]. On the other hand, research about the motion estimation is presented in [10], [15]. Further, some motion understanding and trajectory planning which based on the Frenet-Serret formula are described in [4], [16], [17]. Using the knowledge of the motion and the structure, identification of the target dynamics may be accomplished.

The organization of this paper is as follows: Section 2 outlines the procedure of designing the trajectory tracking algorithm. The implementation of these algorithms and the experimental results are presented in Section 3. Finally, conclusions are drawn in Section 4.

2. DESIGN OF TRAJECTORY TRACKING ALGORITHM

We will make clear our main concept and notion for designing the trajectory tracking algorithm in this section. Our goal here is to design a simple, none computation intensive, and numerically stable algorithm. Our main idea is that if the target motion is known, then the camera motion which will eliminate the relative motion of the target may be generated such that the tracking is achieved. In particular, if the motion of target is known well, then the "perfect tracking" may be achieved, i.e., image position of the target can always be kept at the center of image.

It is apparent that the estimation of the target motion is the first step in our proposed algorithm. In general, the motion of target is not known *a priori*. However, it can be estimated through the change of the image position which is called "optical flow" or image displacement. Using the observation of image displacement or optical flow which induced by target motion, one can estimate the motion of target if the camera is stationary. For the stationary camera and moving object, the optical flow induced by the target motion is:

$$u_o = \dot{x} = -\left(\frac{xT_x}{Z} - \frac{fT_x}{Z}\right) - \left(\frac{xy}{f}\omega_x - \frac{x^2 + f^2}{f}\omega_y + y\omega_z\right) \quad (1)$$

$$v_o = \dot{y} = -\left(\frac{yT_x}{Z} - \frac{fT_y}{Z}\right) - \left(\frac{y^2 + f^2}{f}\omega_x - \frac{xy}{f}\omega_y - x\omega_z\right). \quad (2)$$

where u_o and v_o are known as the optical flow or image displacements and subscript o is denoted as object. If we assume a moving camera and a stationary object instead of a moving object and a stationary camera, then we can obtain the same result as (1) and (2) except for a sign reversal [19].

However it is not the case for the tracking phase since both the camera and target are moving during the tracking phase and therefore the optical flow is subject to both camera and target motion. In light of this observation, we must modify the optical flow equation such that the optical flow is induced only by the target motion and irrelevant to the camera motion. Briefly, there are derived as follow. Suppose now that the optical flow at the time instant k during the tracking phase is $[u(k), v(k)]^T$, then the optical flow is [19], [3]

$$u(k) = u_o(k) + u_{ca}(k) \quad (3)$$

$$v(k) = v_o(k) + v_{ca}(k) \quad (4)$$

where $u(k)$ and $v(k)$ are the total optical flow, $u_o(k)$ and $v_o(k)$ are the optical flow induced by the target motion, whereas $u_{ca}(k)$ and $v_{ca}(k)$ are the optical flow induced by the tracking motion of the camera. Therefore, the optical flow induced by the target motion during the tracking phase can be obtained by modifying (3) and (4) as follows:

$$u_o(k) = u(k) - u_{ca}(k) \quad (5)$$

$$v_o(k) = v(k) - v_{ca}(k). \quad (6)$$

Now, we would like to introduce our method of

finding this observable modified optical flow. Generally, the sum-of-differences (SSD) method is used to find the optical flow [19], [27]. Our method is different from SSD method since we have our own target detection algorithm as described in [7]. Using this target detection algorithm, the target image position can be obtained at any instant. Assume that the target image position is $[x(k-1) y(k-1)]^T$ at frame $k-1$ and our desired camera angular velocities are $\omega_x(k-1) = \phi/\Delta t$ and $\omega_y(k-1) = \theta/\Delta t$, where Δt is the time interval between two consecutive frames, ϕ is the pan angle and θ is the tilt angle. Let the target move to the position $[x(k) y(k)]^T$ at the next frame, namely frame k , then the $u(k-1)$, $v(k-1)$, $u_{ca}(k-1)$, $v_{ca}(k-1)$ at frame k are derived as follows:

$$u(k-1) = x(k) - x(k-1) \quad (7)$$

$$v(k-1) = y(k) - y(k-1) \quad (8)$$

$$u_{ca}(k-1) = \omega_x(k-1) \left(\frac{x(k-1)y(k-1)}{f} \right) - \omega_y(k-1) \left(\frac{x^2(k-1) + f^2}{f} \right) \quad (9)$$

$$v_{ca}(k-1) = \omega_x(k-1) \left(\frac{y^2(k-1) + f^2}{f} \right) - \omega_y(k-1) \left(\frac{x(k-1)y(k-1)}{f} \right). \quad (10)$$

Thus,

$$u_o(k-1) = (x(k) - x(k-1)) - \left(\omega_x(k-1) \left(\frac{x(k-1)y(k-1)}{f} \right) - \omega_y(k-1) \left(\frac{x^2(k-1) + f^2}{f} \right) \right) \quad (11)$$

$$v_o(k-1) = (y(k) - y(k-1)) - \left(\omega_x(k-1) \left(\frac{y^2(k-1) + f^2}{f} \right) - \omega_y(k-1) \left(\frac{x(k-1)y(k-1)}{f} \right) \right). \quad (12)$$

Another practical way to obtain this modified optical flow is as follow:

$$u_o(k-1) = x(k) - x(k-1) + (x_{\text{predict}}(k-1) - x_o) \quad (13)$$

$$v_o(k-1) = y(k) - y(k-1) + (y_{\text{predict}}(k-1) - y_o) \quad (14)$$

where x_o , y_o are the image center and $x_{\text{predict}}(k-1)$ and $y_{\text{predict}}(k-1)$ are the predicted target image position.

We can then estimate the target motion by using the modified optical flow. Two motion estimation algorithms are proposed here. The first motion estimation algorithm consists of estimation of translational velocity and rotational velocity, namely "full motion model" whereas the second algorithm composes only of estimation of translational velocity, namely "modest motion model". Motion estimation task becomes peculiarly difficult in trajectory tracking since neither the structure (depth) nor the target motion is known. Accordingly, we propose a novel motion estimation algorithm which is independent of the estimation of structure. In other words, our algorithm is capable to decouple the estimation of motion from the estimation of structure. Thus, the structure (depth) need not be known or estimated before solving the motion estimation task. To this end, the weak perspective projection [8], [2], [25] is used to alleviate this problem. The weak perspective projection model gives a good approximation when the size and the depth variation of the object are small compared with the distance between the object and the camera [25]. Details of theoretical error analysis of using this approximation model have been omitted for the sake of brevity and can be found in [7].

Suppose we are in frame k and we are now ready to estimate the target motion by using the modified version of optical flow. First, we abbreviate (1), (2) to the discrete time expression as

$$\begin{bmatrix} u_s(k-1) \\ v_s(k-1) \end{bmatrix} = J(x(k-1), y(k-1), Z(k-1)) \begin{bmatrix} T(k-1) \\ \omega(k-1) \end{bmatrix} \quad (15)$$

where

$$J(x(k-1), y(k-1), Z(k-1)) = \begin{bmatrix} \frac{f}{Z(k-1)} & 0 & \frac{x(k-1)}{Z(k-1)} & \frac{y(k-1)}{Z(k-1)} & \frac{x^2(k-1)+y^2(k-1)}{Z(k-1)} & -y(k-1) \\ 0 & \frac{f}{Z(k-1)} & \frac{y(k-1)}{Z(k-1)} & \frac{x(k-1)}{Z(k-1)} & \frac{y^2(k-1)+x^2(k-1)}{Z(k-1)} & x(k-1) \end{bmatrix}. \quad (16)$$

The matrix $J(x(k-1), y(k-1), Z(k-1))$ is called the *image Jacobian* and $T = [T_x, T_y, T_z]^T \in \mathbb{R}^3$ and $\omega = [\omega_x, \omega_y, \omega_z]^T \in \mathbb{R}^3$ is the translational and rotational velocity.

Let $x_s = [x_s, y_s, z_s]^T$ be the reference point and this point can be arbitrary chosen at either the center or the tip of area of the target. In order to keep the target's center coincident with the image center, the center of area of the target is thus chosen as reference point. The edge points which lie around this reference point are chosen as the feature points. Combining N feature points $[x_1, y_1]^T, \dots, [x_N, y_N]^T$ in (15) and using the weak perspective projection model [7], yields,

$$\begin{bmatrix} u_1^s(k-1) \\ v_1^s(k-1) \\ \vdots \\ u_N^s(k-1) \\ v_N^s(k-1) \end{bmatrix} = \begin{bmatrix} f & 0 & -x_1(k-1) & -\frac{x_1(k-1)y_1(k-1)}{f} & \frac{x_1^2(k-1)+y_1^2(k-1)}{f} & -y_1(k-1) \\ 0 & f & -y_1(k-1) & \frac{y_1^2(k-1)+x_1^2(k-1)}{f} & \frac{x_1(k-1)y_1(k-1)}{f} & x_1(k-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f & 0 & -x_N(k-1) & -\frac{x_N(k-1)y_N(k-1)}{f} & \frac{x_N^2(k-1)+y_N^2(k-1)}{f} & -y_N(k-1) \\ 0 & f & -y_N(k-1) & \frac{y_N^2(k-1)+x_N^2(k-1)}{f} & \frac{x_N(k-1)y_N(k-1)}{f} & x_N(k-1) \end{bmatrix} \begin{bmatrix} T_x(k-1) \\ T_y(k-1) \\ Z(k-1) \\ T_x(k-1) \\ T_y(k-1) \\ \omega_x(k-1) \\ \omega_y(k-1) \end{bmatrix}. \quad (17)$$

Let

$$\mathbf{c} = \begin{bmatrix} u_1^s(k-1) \\ v_1^s(k-1) \\ \vdots \\ u_N^s(k-1) \\ v_N^s(k-1) \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} f & 0 & -x_1(k-1) & -\frac{x_1(k-1)y_1(k-1)}{f} & \frac{x_1^2(k-1)+y_1^2(k-1)}{f} & -y_1(k-1) \\ 0 & f & -y_1(k-1) & \frac{y_1^2(k-1)+x_1^2(k-1)}{f} & \frac{x_1(k-1)y_1(k-1)}{f} & x_1(k-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f & 0 & -x_N(k-1) & -\frac{x_N(k-1)y_N(k-1)}{f} & \frac{x_N^2(k-1)+y_N^2(k-1)}{f} & -y_N(k-1) \\ 0 & f & -y_N(k-1) & \frac{y_N^2(k-1)+x_N^2(k-1)}{f} & \frac{x_N(k-1)y_N(k-1)}{f} & x_N(k-1) \end{bmatrix} \quad \text{and}$$

$$\mathbf{v} = \begin{bmatrix} T_x(k-1) \\ T_y(k-1) \\ Z(k-1) \\ T_x(k-1) \\ T_y(k-1) \\ \omega_x(k-1) \\ \omega_y(k-1) \end{bmatrix}.$$

Rewrite (17) as

$$\mathbf{c} = \mathbf{G}\mathbf{v} \quad (18)$$

where $\mathbf{c}$ is the displacement or optical flow induced by target motion, $\mathbf{G}$ is composed of the feature's image coordinate and focal length and $\mathbf{v}$ consist of depth as well as translational and rotational velocities of the target with respect to the camera coordinate system. Specifically, matrix $\mathbf{G}$ and vector $\mathbf{c}$ depends only on observable quantities whereas vector $\mathbf{v}$ contains the quantities which we want to estimate. In particular, for $N=3$, it will be proved in Theorem 1 shown below that the matrix $\mathbf{G}$ is nonsingular, and hence (18) can be easily solved as follow:

$$\mathbf{v} = \mathbf{G}^{-1}\mathbf{c}. \quad (19)$$

Theorem 1: The matrix $\mathbf{G}$ is nonsingular if and only if three distinct, noncollinear feature points $[x_1, y_1]^T, [x_2, y_2]^T, [x_3, y_3]^T$ satisfy the following condition

$$\begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ x_1^2 + y_1^2 & x_2^2 + y_2^2 & x_3^2 + y_3^2 \end{vmatrix} \neq 0.$$

Proof: Please see [7].

Remark: In particular, the above condition is easily satisfied by any three distinct points.

Let

$$\mathbf{T}_i(k-1) = \begin{bmatrix} T_x(k-1) & T_y(k-1) & T_z(k-1) \\ Z_i(k-1) & Z_i(k-1) & Z_i(k-1) \end{bmatrix},$$

then $\mathbf{T}(k-1)$ is obtained as follow:

$$\mathbf{T}(k-1) = \frac{\mathbf{T}_1(k-1)}{\|\mathbf{T}_1(k-1)\|}. \quad (20)$$

For the modest motion model, we assume that the translational velocity dominates the rotational velocity given that the sampling time is short enough for two consecutive image frames. Therefore, we restrict ourselves to estimate the translational velocity due to the above assumption. In this way, (15) can be simplified as:

$$\begin{bmatrix} u_s(k-1) \\ v_s(k-1) \end{bmatrix} = \begin{bmatrix} \frac{f}{Z(k-1)} & 0 & \frac{-x(k-1)}{Z(k-1)} \\ 0 & \frac{f}{Z(k-1)} & \frac{-y(k-1)}{Z(k-1)} \end{bmatrix} \begin{bmatrix} T_x(k-1) \\ T_y(k-1) \\ T_z(k-1) \end{bmatrix}. \quad (21)$$

Similar to the full motion model presented previously, combining N feature points $[x_1, y_1]^T, \dots, [x_N, y_N]^T$ in (21) and using the weak perspective projection model, yields,

$$\begin{bmatrix} u_1^s(k-1) \\ v_1^s(k-1) \\ \vdots \\ u_N^s(k-1) \\ v_N^s(k-1) \end{bmatrix} = \begin{bmatrix} f & 0 & -x_1(k-1) \\ 0 & f & -y_1(k-1) \\ \vdots & \vdots & \vdots \\ f & 0 & -x_N(k-1) \\ 0 & f & -y_N(k-1) \end{bmatrix} \begin{bmatrix} T_x(k-1) \\ T_y(k-1) \\ T_z(k-1) \end{bmatrix}. \quad (22)$$

$$\text{Let } \mathbf{b} = \begin{bmatrix} u_1^s(k-1) \\ v_1^s(k-1) \\ \vdots \\ u_N^s(k-1) \\ v_N^s(k-1) \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} f & 0 & -x_1(k-1) \\ 0 & f & -y_1(k-1) \\ \vdots & \vdots & \vdots \\ f & 0 & -x_N(k-1) \\ 0 & f & -y_N(k-1) \end{bmatrix} \quad \text{and} \quad \mathbf{a} = \begin{bmatrix} T_x(k-1) \\ T_y(k-1) \\ T_z(k-1) \end{bmatrix}. \quad (23)$$

Then

$$\mathbf{b} = \mathbf{H}\mathbf{a}. \quad (24)$$

This is not surprising for solving (23) by using the least-squares formulation:

$$\mathbf{a} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{b} \quad (25)$$

Specifically, we show that the matrix $(\mathbf{H}^T\mathbf{H})$ is invertible if and only if at least two of the N feature points are in distinctive image position. This will be revealed in Proposition 1.

Proposition 1: The matrix $(\mathbf{H}^T\mathbf{H})$ is invertible if and only if at least two of the N feature points are in distinctive image position.

Proof: See [7].

Using (25), $\mathbf{T}(k-1)$ is obtained as follows:

$$\mathbf{T}(k-1) = \frac{\mathbf{a}}{\|\mathbf{a}\|}. \quad (26)$$

The second step in our proposed algorithm is prediction. The prediction task is to predict where the target image location in frame k will 'move to' in the next frame $k+1$. It is easy to extract the image displacement at frame k by using the following equation if $T(k) \approx T(k-1)$, $\omega(k) \approx \omega(k-1)$, and $Z(k) \approx Z(k-1)$ for the short sampling time.

$$\begin{bmatrix} u_c(k) \\ v_c(k) \end{bmatrix} = \begin{bmatrix} f & 0 & -x_c(k) \\ 0 & f & -y_c(k) \end{bmatrix} \begin{bmatrix} T_x(k) \\ T_y(k) \\ Z(k) \end{bmatrix} + \begin{bmatrix} -\frac{x_c(k)y_c(k)}{f} & f + \frac{x_c^2(k)}{f} & -y_c(k) \\ -\left(f + \frac{y_c^2(k)}{f}\right) & \frac{x_c(k)y_c(k)}{f} & x_c(k) \end{bmatrix} \begin{bmatrix} \omega_x(k) \\ \omega_y(k) \\ \omega_z(k) \end{bmatrix} \quad (27)$$

where (x_c, y_c) is the target's centroid coordinate in image plane.

Thus, the prediction can be obtained as follow:

$$x_c(k+1) = x_c(k) + u_c(k) \quad (28)$$

$$y_c(k+1) = y_c(k) + v_c(k). \quad (29)$$

This value is fed to the tracking module to generate the desired camera motion which will achieve the goal of tracking.

The last step is tracking. The tracking task is generate an accommodated pan and tilt angle so that the image center coordinate $[x_{org} \ y_{org}]^T$ will be coincident with the target coordinate $[x_t \ y_t]^T$ after the pan and tilt movement.

From Fig. 1, we have

$$x_a = x_{org} - x_t \quad (30)$$

$$y_a = y_{org} - y_t \quad (31)$$

where $[x_t \ y_t]^T$ is the actual image coordinates and $[x_a \ y_a]^T$ is the ideal image coordinates. To keep notation simple and without loss of generality (w.l.o.g.), we use the ideal image coordinate in the following mathematical analysis. The equations used in the following mathematical analysis obviously can be re-expressed in terms of actual image coordinates by using the transformation above.

From the prediction step, we predict that the target image position will appear in actual image coordinates $[x_t \ y_t]^T = [x_c(k+1) \ y_c(k+1)]^T$ in the next frame. Assume that the 3D position of target's ideal image coordinates $[x_t \ y_t]^T$ is

$$P = [\alpha x_t \ \alpha y_t \ \alpha f]^T \quad (32)$$

where $\alpha = Z/f$ and the 3D position of image center coordinates $[x_{org} \ y_{org}]^T$ is

$$Q = [0 \ 0 \ k]^T \quad (33)$$

where k is arbitrary nonzero constant.

In order to track the target, we first generate the pan angle ϕ and then output the tilt angle θ so that the error between image center coordinate and target image coordinate can be compensated as shown in Fig. 1. Since a rotation of the camera is equivalent to the inverse rotation of the point in the scene. Therefore, we have

$$\begin{aligned} Q &= R^T(\theta) R^T(\phi) P \\ &= (R(\phi) R(\theta))^T P \\ &= \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}^T P \\ &= \begin{bmatrix} \cos \phi & 0 & -\sin \phi \\ \sin \phi \sin \theta & \cos \theta & \cos \phi \sin \theta \\ \sin \phi \cos \theta & -\sin \theta & \cos \phi \cos \theta \end{bmatrix} P \end{aligned} \quad (34)$$

Using (32), (33) and (34), we have

$$\alpha x_t \cos \phi - \alpha f \sin \phi = 0 \quad (35)$$

$$\alpha x_t \sin \phi \sin \theta + \alpha y_t \cos \theta + \alpha f \cos \phi \sin \theta = 0. \quad (36)$$

Fig. 1. Compensation of the error between image center coordinate and target image coordinate.

Solving the equations (35) and (36), we can readily obtain the pan and tilt angle as follow:

$$\text{pan: } \phi = \tan^{-1}\left(\frac{x_t}{f}\right), \quad (37)$$

$$\text{tilt: } \theta = \tan^{-1}\left(\frac{y_t}{-x_t \sin \phi - f \cos \phi}\right). \quad (38)$$

Using (30) and (31), (37) and (38) can be re-expressed in terms of actual image coordinates as follow:

$$\phi = \tan^{-1}\left(\frac{x_{org} - x_c(k+1)}{f}\right), \quad (39)$$

$$\theta = \tan^{-1}\left(\frac{y_{org} - y_c(k+1)}{-(x_{org} - x_c(k+1)) \sin \phi - f \cos \phi}\right). \quad (40)$$

Finally, these algorithms are summarized in Fig. 2 and 3.

- Full Motion Model Algorithm**

 1. Use (13), (14) to calculate target induced optical flow.
 2. Select three different edge points which lie around the centroid of target and use (19) to estimate the target motion parameters.
 3. Evaluate the predicted target's centroid coordinate by using (27)-(29).
 4. Use (39) and (40) to compute the accommodated tracking command.

Fig. 2. Algorithm 1: full motion model

- Modest Motion Model Algorithm**

 1. Use (13), (14) to calculate target induced optical flow.
 2. Select two different edge points which lie around the centroid of target and use (35) to estimate the translational velocity.
 3. Evaluate the predicted target's centroid coordinate by using (27)-(29).
 4. Use (39) and (40) to compute the accommodated tracking command.

Fig. 3. Algorithm 2: modest motion model

3. IMPLEMENTATION AND EXPERIMENTS

This section illustrates the performance of the trajectory tracking algorithms under a variety of circumstances. All experiments were performed on live image sequences which are grabbed by JAI MCL-1500 DSP color camera and then processed by the MATROX

CORONA image processing card in the Pentium III 450 PC. The control command is then sent to ADVANTECH stepping motor control card so as to generate an appropriate pan/tilt motion that achieves the goal of tracking.

To validate the performance of the proposed trajectory tracking algorithms, we will use the target detection method in simple environment as described in [7]. Timing of the control cycle indicates that the proposed visual tracking system can perform frame rate (30 Hz) tracking of image regions of up to 640×480 pixels.

3.1 Indoor Experiments

An aircraft model is used as our target. At any time instant, the location of the target in the image plane is fed to the trajectory tracking algorithm to predict the future position as shown in Fig. 4(a) and 4(b). The tracking error is shown in Fig. 4(c). Note that the center of the image plane is $[320 \ 240]^T$. It is clear that the predicted target position is close to the true target position. This implies that our proposed algorithm provides accurate prediction of the maneuvering target trajectory.

Fig. 4. Experimental results of indoor experiment.

3.2 Outdoor Experiments

• Correspondence-free

Image sequence in Fig. 5 shows the correspondence-free property of this algorithm in the outdoor scene. The big square is the neighbourhood of the origin of the image plane whereas the three centers of small squares represent three feature points used in this algorithm. This image sequence shows that the target is still in the neighbourhood of the origin by using any three different feature points. This is due to the fact that the feature point in the two consecutive image frames will not change too much for the fast sampling rate. Hence, the calculation of the optical flow is still correct.

• Robustness

Figure 6 shows the image sequence in which the rotation motion dominates the translational motion. For the rotation motion which dominates the translational motion, the target will change the pose, and this will yield the error in the computation of optical flow. However, the experimental results shown in Fig. 7 demonstrate that this effect will not hurt our algorithm.

Fig. 5. Image sequence.

Fig. 6. Image sequence.

Fig. 7. Experimental results of the robustness.

4. CONCLUSIONS

In this paper, motion estimation algorithms for the

most general tracking situation are developed. The contribution of this work is that simple, none computation intensive, correspondence-free and numerical stable 3D motion estimation algorithms are developed. An important factor for the successful application of the extremely simple algorithms is based on the use of weak perspective projection model. The trajectory tracking algorithm based on these proposed motion estimation algorithms provides accuracy of the prediction of the trajectory of the maneuvering target.

The proposed algorithms have been implemented in our visual tracking system [7]. Experimental results show that the proposed algorithms are accurate, robust, and feasible. In addition, all the experiments were done in the real-time. The potential of the proposed methods has been demonstrated by the presenting experimental results. We believe that the simplicity and the robustness of the proposed approaches can be applied to other general visual tracking system.

REFERENCES

- [1] G. Adiv, "Determining Three-Dimensional Motion and Structure from Optical Flow Generated by Several Moving Objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no. 4, pp. 384-401, Jul 1985.
- [2] J. Y. Aloimonos, "Perspective Approximations," *Image and Vision Computing*, vol. 8, no. 3, pp. 179-192, Aug 1990.
- [3] J.(Y.) Aloimonos and D. P. Tsakiris, "On the visual mathematics of tracking," *Image and Vision Computing*, vol. 9, no. 4, pp. 235-251, Aug 1991.
- [4] J. Angeles, A. Rojas, and C. S. Lopez-Cajun, "Trajectory Planning in Robotics Continuous-Path Applications," *IEEE J. Robotics and Automation*, vol. 4, no. 4, pp. 380-385, Aug 1988.
- [5] A. Azarbayejani and A. P. Pentland, "Recursive Estimation of Motion, Structure, and Focal Length," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 6, pp. 562-575, June 1995.
- [6] Y. Bar-Shalom, K. C. Chang and H. A. P. Blom, "Tracking a Maneuvering Target Using Input Estimation Versus the Interacting Multiple Model Algorithm," *IEEE Trans. on Aero. and Electro. Sys.*, vol. AES-25, no. 2, pp. , Mar 1989.
- [7] W. G. Yau, "Design and Implementation of Visual Servoing System for Realistic Air Target Tracking," Master's thesis, Dept. Elec. Eng., National Taiwan University, 2000.
- [8] Andrew Blake and Michael Isard, *Active Contours*, London : Springer-Verlag, c1998.
- [9] H. A. P. Blom and Y. Bar-Shalom, "The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients," *IEEE Trans. Automatic Control*, vol. 33, no. 8, pp. 780-783, Aug 1988.
- [10] S. Soatto, R. Frezza, and P. Perona, "Motion Estimation via Dynamic Vision," *IEEE Trans. Automatic Control*, vol. 41, no. 3, pp. 393-413, Mar 1996.
- [11] K. J. Bradshaw, I. D. Reid, and D. W. Murray, "The Active Recovery of 3D Motion Trajectories and Their Use in Prediction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 219-233, Mar 1997.
- [12] W. S. Chaer, R. H. Bishop, and J. Ghosh, "A Mixture-of-Experts Framework for Adaptive Kalman Filtering," *IEEE Trans. System, Man, and Cybernetics-Part B: Cybernetics*, vol. 27, no. 3, pp. 452-464, June 1997.
- [13] N. Sawasaki, T. Morita, and T. Uchiyama, "Design and Implementation of High-Speed Visual Tracking Systems for Real-Time Motion Analysis," *IEEE Int. Conf. Pattern Recognition*, vol. 3, 478-483, 1996.
- [14] L. Chen and S. Chang, "A Video tracking system with adaptive predictors," *Pattern Recognition*, vol. 25, no. 10, pp. 1171-1180, Oct 1992.
- [15] Z. Duric, J. A. Fayman, and E. Rivlin, "Function From Motion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 6, pp. 579-591, June 1996.
- [16] Z. Duric, E. Rivlin, and A. Rosenfeld, "Understanding the Motions," *Image and Computing*, vol. 16, pp. 785-797, 1998.
- [17] Z. Durić, A. Rosenfeld, and L. S. Davis, "Egomotion Analysis Based on the Frenet-Serret Motion Model," *Proc. IEEE 4th Int. Conf. Computer Vision*, pp.703-712, April 1993.
- [18] M. Efe and D. P. Atherton, "Maneuvering Target Tracking With an Adaptive Kalman Filter," *Proc. 37th IEEE Conf. Decision & Control Tampa, Florida*, pp. 737-742, Dec 1998.
- [19] N. P. Papanikolopoulos and P. K. Khosla, "Adaptive Robotic Visual Tracking: Theory and Experiments," *IEEE Trans. Automatic Control*, vol. 38, no. 3, pp. 429-445, Mar 1993.
- [20] M. W. Eklund, G. Ravichandran, M. M. Trivedi, and S. B. Marapane, "Adaptive Visual Tracking Algorithm and Real-time Implementation," *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 2657-2662, 1995.
- [21] B. Espiau, F. Chaumette, and P. Rives, "A New Approach to Visual Servoing in Robotics," *IEEE Trans. Robotics and Automation*, vol. 8, no.3, pp. 313-326, June 1992.
- [22] O. Faugeras and T. Papadopoulos, "A Theory of the Motion Fields of Curves," *Int. J. of Computer Vision*, vol. 10, no. 2, pp. 125-156, 1993.
- [23] J. T. Feddema, C. S. G. Lee, "Adaptive image feature prediction and control for visual tracking," *IEEE Trans. Systems, Man and Cybernetics*, vol. 20, no. 5, pp. 1172-1183, Sept-Oct 1990.
- [24] J. T. Feddema, C. S. G. Lee, "Adaptive image feature prediction and control for visual tracking with a moving camera," *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*, pp. 20-24, Nov 1990.
- [25] T. Otsuka and J. Ohya, "Real-time Estimation of Head Using Weak Perspective Epipolar Geometry," *IEEE Workshop on Application Computer Vision*, pp. 220-225, 1998.
- [26] D. B. Gennery, "Visual Tracking of Known Three-Dimensional Objects," *Int. J. Computer Vision*, vol. 7, no. 3, pp. 243-270, 1992.
- [27] L. Matthies, T. Kanade, and R. Szeliski, "Kalman Filter-based Algorithms for Estimating Depth from Image Sequences," *Int. J. Computer Vision*, vol. 3, pp. 209-236, 1989.
- [28] X. R. Li, and Y. Bar-Shalom, "Multiple-Model Estimation with Variable Structure," *IEEE Trans. Automatic Control*, vol. 41, no. 4, pp. 478-493, April 1996.
- [29] K. Hashimoto, T. Ebine, and H. Kimura, "Visual Servoing with Hand-Eye Manipulator—Optimal Control Approach," *IEEE Trans. Robotics and Automation*, vol. 12, no. 5, pp. 766-774, Oct 1996.
- [30] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator Control with Image-Based Visual Servo," *Proc. IEEE Int. Conf. Robotics & Automation Sacramento, California*, pp. 2267-2272, Apr 1991.