Filter Algorithm for Visual Tracking of Maneuvering Target

Feili Hou\textsuperscript{1,2} Feng Zhu\textsuperscript{1}

\textsuperscript{1}Robotics Laboratory, Shenyang Institute of Automation, Chinese Academy of Sciences
Shenyang, 110016, China
\textsuperscript{2}Graduate School of the Chinese Academy of Sciences
Beijing, 100039, China

Abstract - A filter algorithm is presented in this paper for visual tracking of a maneuvering target. Emphasis is given to find a solution for the degradation in relative position and orientation estimation, which is incurred by the measurement noise in the image coordinate of feature points. Superior to previous approaches that were limited to the assumption that the target motion is slow and smooth, this algorithm is implementable for a maneuvering target that acts in an unknown manner. First, by analyzing the effect of noise in 2-D images on the position and orientation estimation, linear measure equations based on the sequence of motion parameters are given. Then, two filter schemes are introduced respectively. The first filter uses maneuver detection technique, in which optimized detectors for fast and slow maneuver are deduced respectively, and limited memory filtering is adopted to update the filter. The second filter uses numerical differentiation technique, in which a fading factor is adaptively estimated to restrain the divergence caused by truncation errors of estimate model of numerical differentiation. Finally, generalized pseudo Bayes algorithm is employed to combine the two filters for a higher tracking precision. Simulation and experiment results illustrate the capacity of this algorithm.

1. INTRODUCTION

One of the central problems in visual tracking is the determination of the 3D relative pose (position and orientation) of a randomly moving target with respect to the tracker. In recent years, the model-based monocular vision method has been widely used to estimate the 3D relative pose. However, noise in the image will cause error in the extraction of image feature location, and thereby, result in poor individual pose estimates. Based on the fact that a long image sequence can provide more information to supress noise, several authors have applied extended Kalman filter theory to reduce the estimate error.

However, although these methods were found to be successful in most cases, there are still some limitations on their ability. The statistics of the dynamic noise must be known prior to its application, and the object motion should be slow and smooth. Therefore, in applications where the object may travel through a sudden movement or a step discontinuity, more work is needed.

In this paper, a robust filter algorithm is presented for visual tracking of maneuvering target, with no need of the prior knowledge of the object trajectory. The whole filter algorithm is illustrated in Fig. 1. Because it avoids the need of extended Kalman filter, the structure of this filter becomes much simpler and computationally more efficient. Moreover, six separate filters for the six pose parameters run in parallel, which helps increase the ability for real time application.

In this filter algorithm, better tracking performance have been achieved by combing two filter modules with generalized pseudo Bayes algorithm, especially at the maneuvering time when the target goes through a sudden change. As for the two separate filter schemes, improvements to pervious methods have been made on them respectively. In the first filter that uses maneuver detection technique, the detectors that fit for fast and slow maneuvers are proposed.

![Fig. 1 Flowchart of the filter algorithm](image)

0-7803-8416-4/04/$20.00 ©2004 IEEE.
respectively. And, limited memory filtering, rather than the acceleration input estimation method \([9][10][11]\), is introduced for maneuver correction and filter update. Therefore, the opposite influence caused by poor estimation of the starting time of maneuvering is avoided, and better filter performance has been acquired. In the second filter algorithm, the estimate model of numerical differentiation is robust to diverse dynamics. And, a fading factor is imported to overcome the possibility of filter divergence. This factor has the capacity of being estimated adaptively.

This filter algorithm has some other advantages as well as reducing the effect of noise. Based on its predictive capability, pose parameters in the next step time can be acquired, and then, appropriate regions to which feature extraction is restricted can be easily computed. This allows the vision sensor system to process only small areas in the image plane to obtain the required feature point measurements, and thus results in a significant reduction in image processing time.

This paper consists of 8 sections. Section 2 briefly presents model-based monocular vision system for the individual 3D object pose estimation. In section 3, linear measurement equations of the pose parameters are derived. Section 4 describes the first filter scheme using maneuver detection technique, and section 5 proposes the second filter scheme based on the estimate model of numerical differentiation. After the combination of the two filter modules is given in section 6, simulation and experimental results are presented in section 7. Finally, a conclusion is made in section 8.

II. MODEL-BASED MONOCULAR VISION SYSTEM

This section will describe the reconstruction of 3D pose by the model-based monocular vision method. The mathematical relationship between the pose parameters and the image plane feature measurements is shown in Fig. 2. A camera coordinate frame is set up on the image plane, while an object coordinate frame is attached to the object. First, we select three feature points on the object. Then, after acquiring image of the object, each feature points in the image plane can be extracted by image processing. With their image coordinates, and their known location in the object frame, each 3D feature position in the coordinate frame C can be reconstructed by the Perspective-3-Point algorithm. Based on the fact that the transformation of the feature measurement in the object frame to the camera frame can be performed by a translation vector \(T\) and a rotation matrix \(R\), a solution for the pose parameters, including the position parameters (forward \(x(t)\), lateral \(y(t)\), and vertical \(z(t)\) translation), and the orientation parameters (roll \(\theta_r(t)\), pitch \(\theta_p(t)\) and yaw \(\theta_y(t)\) angle), are obtained.

Since some pose parameters have high sensitivities to the small image measurement errors, a filter method is presented next to provide a series of better pose estimates.

III. LINEAR MEASUREMENT EQUATIONS

In conventional measurement models, noisy image feature vector serves as measurements. However, considering its nonlinear relationship with the pose parameters, here we choose the pose parameters, instead of the image feature vector, as a measurement. Therefore, the linear measurement equation that relates nominal parameters (obtained by 3D reconstruction) and the true ones is acquired as

\[
\begin{bmatrix}
\theta_r(t) \\
\theta_p(t) \\
\theta_y(t) \\
x(t) \\
y(t) \\
z(t)
\end{bmatrix} =
\begin{bmatrix}
\theta_r(t) \\
\theta_p(t) \\
\theta_y(t) \\
x(t) \\
y(t) \\
z(t)
\end{bmatrix} +
\begin{bmatrix}
n_{\theta_r}(t) \\
n_{\theta_p}(t) \\
n_{\theta_y}(t) \\
n_x(t) \\
n_y(t) \\
n_z(t)
\end{bmatrix}
\]

(1)

where \(n_{\theta_r}(t), n_{\theta_p}(t), n_{\theta_y}(t)\) are the noise components, which are assumed to be Gaussian processes with mean zero.

Now, we will derive the measurement noise covariance by the 3 \(\sigma\) rationale. Obviously, the pose noise has a varying statistics. However, its maximum value can be derived from the statistics of noise in the image feature measurements. According to the 3 \(\sigma\) rationale, if the maximum error of some pose parameter is \(L(t)\), then, its covariance can be approximated as \((L(t)/3)^2\).

Since the six pose parameters are independent with each other, six filters can run on them respectively for less computation time. Next, we will give two filter methods respectively, and further, propose the filter fusion process. Filtering on the forward translation \(x(t)\) is chosen as representative. Other pose parameters are filtered in the similar way and therefore omitted.

IV. FILTER USING MANEUVER DETECTION

A. Dynamic Equation

The dynamic equation is given by a three-state model

\[
X_{k+1} = \phi X_k + BU_k + W_k
\]

(2)

where \(X_k = \begin{bmatrix} x(k) \\ \dot{x}(k) \\ \ddot{x}(k) \end{bmatrix}^T, \phi = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\).
and $U_k$ is the sudden acceleration of Maneuver.

B. Maneuver Detection

Once there exists maneuver in the object motion, we should detect it as soon as possible. Here, we will propose a maneuver detector that has the maximum probability of detection.

Suppose that the maneuver starts at time $m$ with acceleration input of $U$, and then, the maneuver detector composed of weighted sum of innovation $\delta_i$ has the form of

$$L_k = \sum_{i=1}^{d} b_i \delta_{k-d+i} / \sqrt{\sum_{i=1}^{d} b_i^2 \sigma_{k-d+i}^2}$$

(3)

If $L_k$ exceeds a certain threshold, we accept the hypothesis that a maneuver has taken place. In (3), $d$ is the number of measurements used to estimate the detector, $\sigma^2_i$ is the covariance of innovation $\delta_i$, and $b_i$ is the weight coefficient.

Now, we will find the proper weigh coefficient $b_i$, under which (3) is optimal.

Let $e_k$ as the unbiased innovation, then the relation between the biased innovation $\delta_k$ and the unbiased one $e_k$ is

$$\delta_k = H_{k-m}U + e_k$$

(4)

where $H_{k-m} = C \phi_k$,

$$C \phi_k \Pi_{i=m+1}^{k-1} (I-K_i)C \phi_k \phi_k$$

$k > m + 1$

So, $\delta_k$ is normally distributed as

$$\delta_k \sim N(H_{k-m}U, \sigma^2_k)$$

(5)

From (3) and (4), we get the distribution of $L_k$

$$L_k \sim N\left(\sum_{i=1}^{d} b_i H_{k-m-d+i}U, \sum_{i=1}^{d} b_i^2 \sigma_{k-d+i}^2\right)$$

(6)

Define $J_k = \sum_{i=1}^{d} b_i H_{k-m-d+i}U / \sqrt{\sum_{i=1}^{d} b_i^2 \sigma_{k-d+i}^2}$, it can be proven that under a given probability of false alarm, the probability of detection will be maximum if $|J_k|$ is maximized. Therefore, by differentiating $J_k$, we reach the conclusion that under the condition of

$$b_{opt}^i = \frac{H_{k-m-d+i} \sigma_{k-d+i}^2}{\sum_{j=1}^{d} H_{k-m-d+j} \sigma_{k-d+j}^2}$$

(7)

$L_k$ in (3) is the optimal detector.

However, in fact, the maneuver starting time $m$ is not obtainable. So, we need to probe sub-optimized detectors for various circumstances.

Analyzing the input estimation based detector given in [9][10][11]

$$L_k = \hat{U}_k / \sqrt{\text{Var}(\hat{U}_k)}$$

(8)

where $\hat{U}_k$ is the estimated input at time $k$

$$\hat{U}_k = \sum_{i=1}^{d} b_i \delta_{k-d+i} = \sum_{i=1}^{d} \frac{H_i \sigma_{k-d+i}^2}{\sum_{j=1}^{d} H_j \sigma_{k-d+j}^2}$$

(9)

we find that the shorter the maneuver detection delay, the closer the detector of (8) is to the optimized detector of (7). So, (8) is suitable for fast maneuver.

In slow maneuver, since the detection delay is long, we can approximate that $H_{k-m-d+1} \cdots H_{k-m}$ equals to each other. And $\sigma_{k-d+i}(i=1 \cdots d)$ can also be deemed as the same. So, the weight coefficient in (7) can be approximated as $b_{opt}^i = 1/d$.

.i.e., the detector

$$L_k = \sum_{i=1}^{d} \delta_{k-d+i} / \sqrt{\sum_{i=1}^{d} \sigma_{k-d+i}^2}$$

(10)

is suitable for slow maneuver.

C. Maneuver correction

Limited memory filtering is adopted here for filter update when maneuver is declared. Compared with the acceleration input estimation method in [9][10][11], this correction scheme performs better because it does not need to estimate the starting time of maneuver, and therefore, avoids the adverse effects caused by its poor estimation. The update process is illustrated in Fig. 3. If maneuver is detected at time $k$, the estimated state $\hat{X}_k$ and its error covariance matrix $\hat{P}_k$ will be updated using the observations acquired after $(k-N)$ ($N$ is the memory length), and replaced by $\hat{X}_k$ and $\hat{P}_k$:

![Fig. 3 Limited memory filtering for maneuver correction](image_url)
\begin{align}
    \hat{X}_k &= (P_k - P_{k-1})^{-1} (P_k \hat{X}_{k-1} - P_{k-1} \hat{X}_{k-1}) \\
    \hat{P}_k &= (P_k - P_{k-1})^{-1}.
\end{align}
(11)

V. FILGER USING NUMERICAL DIFFERENTIATION

A. Dynamic Equation

Numerical differentiation technique \cite{17,18} of three points is adopted here to describe the velocity. This estimate model is robust, which is applicable to the process of diverse dynamics. The continuous time dynamic model is

\begin{equation}
    X_t = AX_t + W_t
\end{equation}
(12)

where \( X_t = 
\begin{bmatrix}
    x_t \\
    \dot{x}_t \\
    \ddot{x}_t \\
    \dot{\ddot{x}}_t
\end{bmatrix} \), \( A = 
\begin{bmatrix}
    0 & 0 & 0 & 0 \\
    1 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 \\
    0 & 0 & 1 & 0
\end{bmatrix} \), \( W_t \).

Discretize (12), we get the discrete dynamic model

\begin{equation}
    X_{k+1} = \phi X_k + W_k
\end{equation}
(13)

where \( X_k = 
\begin{bmatrix}
    x_k \\
    \dot{x}_k \\
    \ddot{x}_k \\
    \dot{\ddot{x}}_k
\end{bmatrix} \), \( \phi = 
\begin{bmatrix}
    1 & 2T/12 & -4T/3 & 5T/12 \\
    0 & 1 & 0 & 0 \\
    0 & 0 & 1 & 0 \\
    0 & 0 & 0 & 1
\end{bmatrix} \), \( W_k \).

B. Divergence Avoidance

In \cite{17,18}, the truncation error is incorporated in the process noise, which makes it difficult to determine the dynamic noise covariance. Poor estimation of the dynamic noise covariance matrix can degrade the accuracy of the filter performance, and may even cause divergence.

To avoid the divergence, we imported a fading factor \( S_k \) here, which acts by reducing the weight of the old observations and increasing that of the new observations. The divergence avoidance scheme is:

If
\begin{equation}
    \delta_k^T \delta_k > \gamma \text{tr}(E(\delta_k \delta_k^T))
\end{equation}
(14)

the filter is supposed to be divergent. In (14), \( \gamma \) is a scale factor, \( \delta_k \) is the innovation with its covariance
\begin{equation}
    E(\delta_k \delta_k^T) = C P_{k-1} C^T + R_k
\end{equation}
(15)

If filter divergence is detected, the covariance matrix of the prediction error will be corrected by
\begin{equation}
    P_{k-1} = S_k \phi P_{k-1} \phi^T + Q_{k-1}
\end{equation}
(16)

In (16), the fading factor \( S_k \) can be estimated adaptively according to measurements:

\begin{equation}
    S_k = \frac{1}{m} \text{tr}\{[\delta_k \delta_k^T - C Q_{k-1} C^T - R_k] [C \phi P_{k-1} \phi^T C^T]^{-1}\}
\end{equation}
(17)

, in which \( m \) is the observation dimension, \( C \) is the measurement matrix.

VI. FILTER FUSION

To get higher tracking quality, generalized pseudo Bayes algorithm is introduced here for combination of the two filter modules. This fusion approach is composed of the two above filters, a model probability evaluator and an estimate combiner at the output of the filters.

Let the notation \( M_i(k) \) stands for “model i in effect during the period ending at time kT,” which is assumed a finite Markov chain with the transition probability matrix \( H \). And, define \( Z^k \) = “the cumulative data at time kT,” \( \mu_i(k) \) = “the model probability of the ith filter at time kT.” Then, one cycle of the fusion approach consists of the following:

1. Each of the pairs \( \hat{X}_i(k|k-1) \) and \( P_i(k|k-1) \) is used as input to the filter matched to \( M_i(k) \). Time-extrapolation yields \( \hat{X}_i(k|k) \) and \( P_i(k|k) \), and measurement update provides \( \hat{X}_i(k|k) \) and \( P_i(k|k) \).

2. The likelihood function corresponding to the ith filter is computed according to
\begin{equation}
    \Delta_i(k) = p(z_k|M_i(Z^{k-1}))
    = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} \frac{1}{\Delta_j(k)} \frac{p(z_j|P_j(k|k-1)) \delta_j(k)}{p(z_j|P_j(k|k-1)) \delta_j(k)} \right)
\end{equation}
(18)

3. The model probability of each filter is updated according to the Bayes theorem
\begin{equation}
    \mu_i(k) = \frac{p(z_k|M_i(Z^{k-1})) \mu_i(Z^{k-1})}{\sum_j p(z_k|M_j(Z^{k-1})) \mu_j(Z^{k-1})}
    = \frac{\Delta_i(k)}{\sum_j \Delta_j(k) \mu_j(k-1)} \mu_i(k-1)
\end{equation}
(19)

4. For output, the total probability theorem is applied as
\begin{equation}
    X(k|k) = \mu_i(k) \hat{X}_i(k|k) + \mu_j(k) \hat{X}_j(k|k)
\end{equation}
(20)

VII. SIMULATION AND EXPERIMENTAL RESULTS

A computer simulation was performed to test the performance of our filter approach. The noisy measurements were made in the following way: Noise free 3D feature points generated by a computer program are projected onto the image plane. Then white independent Gaussian noise is added to the true 2D image coordinates. With the 2-D noisy image points, reconstruction of 3D pose by the Perspective-3-Point algorithm is carried out to simulate noisy measurements of pose parameters.

A Monte Carlo simulation of 50 runs was obtained. The
rms values of the estimation errors of both rotational and translational parameters were computed. Results on forward translation x(t) and roll angle $\theta_z(t)$ were given in following figures. Other four parameters are similar and therefore omitted.

Simulation data were obtained from the following scenario: As denoted in Fig. 4, motion in x direction is on a constant velocity except at time t=30s-40s and t=80-120, there are acceleration input of $a_1=0.1m/s^2$ and $a_2=0.01875m/s^2$. Fig. 5 shows the motion of roll angle $\theta_z$, which is on a constant acceleration 0.05 deg/s$^2$ until t=50s with acceleration input of $a=0.058 deg/s^2$.

Fig. 6 compares the rms errors in x-position by two maneuver correction methods, the limited memory filtering (LM) proposed in this paper and the input estimation (IE) given by [9][10][11]. It is seen that the two trackers appear to be equally effective in the constant course of the target trajectories. During the maneuvering period, however, the superiority of the maneuver correction method of limited memory filtering is shown by lower rms errors.

Fig. 7 is a plot of rms errors in x-position under the maneuver detection based filter (MD), the numerical differentiation based filter (ND) and the fusion of the two filters (FF). Curve UF denotes the rms errors of measurements before filtering. From Fig. 7, we find that each filter algorithm performs well separately. After combining the two filters, the peak of rms errors is decreased at maneuvering time. Therefore, it is verified that after filter fusion, better position estimation will be achieved during the time that the target motion goes through a sudden change.

![Fig. 4 Motion of x-position](image1)

![Fig. 5 Motion of $\theta_z$-roll angle](image2)

![Fig. 6 Comparison of rms errors in x-position](image3)

![Fig. 7 rms errors in x-position](image4)

![Fig. 8 Output of x-translational velocity](image5)

![Fig. 9 rms errors in x-translational velocity](image6)

![Fig. 10 rms errors in $\theta_z$-roll angle](image7)

![Fig. 11 Output of $\theta_z$-roll velocity](image8)

![Fig. 12 rms errors in $\theta_z$-roll velocity](image9)
Fig. 13 Robot tracking control

Fig. 8 and Fig. 9 demonstrate the good estimation of the x-translational velocity. It can be seen that the estimated velocity is very close to the real one, and the rms error in x-translational velocity is decreased quickly.

Simulation results for roll angle $\theta_r$ are shown in Fig. 10-Fig. 12. These figures show the equally good tracking ability of the filter algorithm on the orientation parameters.

Real-time experiment on the robot tracking control was also performed to verify the simulation results. The experimental setup is shown in Fig. 13. Robot II tracked the trajectory of Robot I, which was commanded to travel through a variety of maneuvers. The camera was mounted on the end-effector of Robot I, and three feature points were set up on Robot II. After acquiring the image of the feature points, independent relative pose between Robot II and Robot I was derived by the model-based monocular vision method. Noisy pose measurements were input to the filter to produce much more confident pose estimates, 3D motion velocity, and predictions of 3D pose in the next sampling time. Output signals of the filter were fed back to the controller of Robot I to track Robot II. In the meantime, predictions of the pose parameters, along with the 3D motion velocity, were input to the image processor. Hereby, the process to extract the feature points became much simplified, and the image processing time was greatly reduced.

VIII. CONCLUSION

In this paper, we have presented a robust filter algorithm for visual tracking of maneuvering targets. This algorithm could accurately estimate the pose of a maneuvering target without prior knowledge about its trajectory. Linear measurement equations on the relative pose parameters are derived, two filter algorithms based on the maneuver detection method and the numerical differentiation technique are improved respectively, and better filter performance is acquired by combining the two filters. Moreover, six filters working in parallel, so the computational time is much decreased for real-time implementation. Simulation and Experiment results show that this filter will lead to a more robust and autonomous motion-tracking scheme.

REFERENCES


