

Motion based 3D Target Tracking with Interacting Multiple Linear Dynamic Models

Zhen Jia and Arjuna Balasuriya

School of EEE, Nanyang Technological University, Singapore
jiazhen@gmail.ntu.edu.sg, earjuna@ntu.edu.sg

Subhash Challa

Faculty of Engineering, The University of Technology, Sydney, Australia
schalla@eng.uts.edu.au

Abstract

In this paper, an algorithm is proposed for vision-based object identification and tracking by autonomous vehicles. In order to estimate the speed of the tracking object, this algorithm fuses information captured by on-board sensors such as camera and inertial sensors. To formulate the tracking algorithm it is necessary to use a proper model which describes the dynamics of the tracking object. However due to complex nature of the moving object, it is necessary to have different dynamic models. Here, several simple and basic linear dynamic models are combined to approximate unpredictable, complex dynamics of the moving target. With these basic linear dynamic models, a detailed description of the three dimensional (3D) target tracking scheme using an interacting multiple model (IMM) along with an Extended Kalman Filtering is presented. The final state of the target is estimated as a weighted combination of the outputs from each different dynamic model. Performance of the proposed interacting multiple dynamic model tracking algorithm is demonstrated through experimental results.

1 Introduction

Vision-based target tracking is an active research area and has many applications. This paper focuses on applications of target tracking by autonomous vehicle navigation. Intelligent transportation systems play a vital role in the society. Target tracking schemes aid the intelligent navigation system by providing information about the dynamic behaviour of the moving objects in the vicinity of the vehicle[3].

The proposed vision system uses optical Flow as the main source of information to identify the target and its motion. Optical flow vectors reflect the relative motion of a target in the image stream. Past research [7] show that tracking can be achieved by fusing optical flow vectors with other visual cues. In this paper, optical flow vectors are fused with the motion data to estimate the 3D velocity of the target and fused with other visual cues to identify the region of interest (ROI) in the image.

Target tracking is a challenging problem due to the unpredictable dynamic nature of the tracking object. It is difficult to model these non-linear dynamics with a single set of equations. Tracking of such targets thus falls into the area of adaptive state

estimation (multiple models)[6]. According to Bar-Shalom [2] the Interacting Multiple Model (IMM) performs significantly better than other multiple model methods. The IMM algorithm computes the state of the object under each possible model using several filters. Each filter uses a different combination of the previous conditions estimated (mixed initial condition). The mixed initial condition of IMM can reduce the estimation error by adjusting the initial condition for each estimates. IMM is also faster in computation than other multiple model algorithms, for example IMM only requires r filters to operate in parallel while the generalized pseudo-Bayesian(GPB) multiple models algorithm requires r or r^2 to operate in parallel. Furthermore, the IMM algorithm is able to keep the estimation error less than the raw measurement error and provides a significant improvement (noise reduction). Thus, in this paper IMM algorithm is deployed to fuse several simple and basis linear dynamic models [4] for dynamic target tracking. In order to improve tracking performance, the Kalman Filter is combined with weighted models for state estimation.

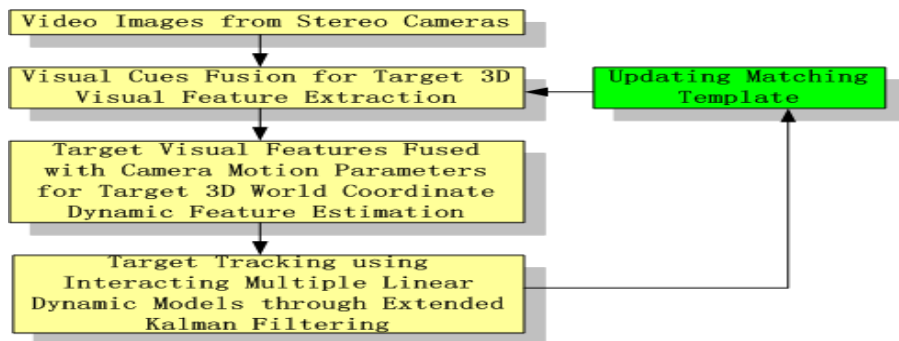


Figure 1: Proposed Tracking System

The proposed tracking system is described schematically in Fig.1. Here, clustered motion vector fields and color information in the image are fused together to identify the region of interest (Section 2). Next, optical flow vectors are fused with visual depth information and are combined with the kinematic model of the moving camera system to estimate the 3D position and velocity of the target(Section 3). In the final step, the target’s dynamic features represented by simple linear dynamic models by the IMM are used in the Extended Kalman Filter for tracking(Section 4).

2 Target Identification

2.1 Target 3D Visual Feature Extraction

In this paper, Horn’s gradient-based iterative algorithm is modified for optical flow estimation due to its fast computational speed and accuracy[1][5].

Since the cameras are moving, the background introduces optical flow vectors together with the motion of the target. Assuming that the moving target is a rigid object with uniform motions, the region of the target can be separated from the background using the motion vector field clusters through template matching[7]. Here, K-means algorithm is used to segment the optical flow vector fields into several clusters

to identify different motion fields in the image. With clustered optical flow vector fields a template matching technique is used to extract the ROI from the background motions. The ROI's coordinates are considered as 2D coordinates of the target in the image. Here, the initial template is obtained manually from the clustered optical flow motion fields. The extraction of the ROI is carried out using target's motion fields. This makes the tracking system robust to track unknown targets independent of its other optical properties such as shape, size etc.. During the tracking process, the template is updated in real-time with the estimated ROI.

The visual depth information is obtained through stereo image pairs. Assuming the camera axis are parallel and separated by b_1 and the focal length of the camera's to be f , visual depth Z can be estimated based on disparity [5] given by $z = b_1 \frac{f}{x'_l - x'_r}$. Here, coordinate difference of the same point in left and right images ($x'_l - x'_r$) is known as disparity d , which is estimated by the sum of absolute RGB difference (SAD) algorithm [8].

2.2 Target 3D Dynamic Information Estimation

After identifying 3D visual features of the target, this section looks at the fusion algorithm for estimating the 3D velocities and position of the target.

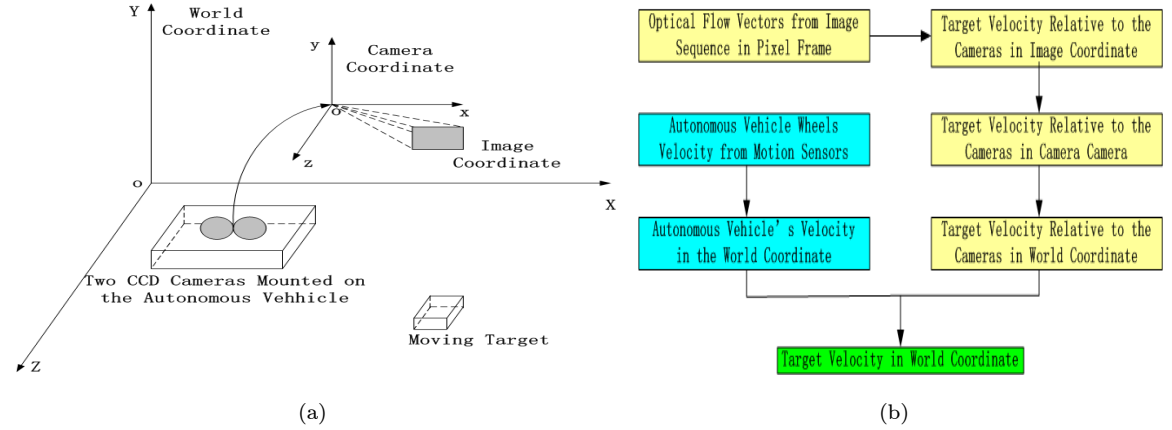


Figure 2: Experiment Setup with Target World Coordinate Velocity Estimation Procedure

First, assume the cameras mounted on an autonomous vehicle moving in the environment as shown in Fig.2(a). The target velocity can be obtained from the detailed image processing process depicted in Fig.2(b). First the motion of the target in the image stream is obtained as the optical flow vectors $\vec{V}_{O.F.}$. Then $\vec{V}_{O.F.}$ is transformed to estimate the target image velocity \vec{V}_{IM} . Next \vec{V}_{IM} is transformed into the camera frame to get the target velocity relative to the cameras \vec{V}_{IM}^C . The target velocity relative to the cameras in the world coordinate \vec{V}_{IM}^W can be estimated from \vec{V}_{IM}^C . At the same time the velocity of the moving camera in the world coordinate \vec{V}_C^W can be estimated from the kinematics model of the autonomous vehicle. Finally the target's

velocity in world coordinate \vec{V}_T^W can be written as:

$$\vec{V}_T^W = \vec{V}_{IM}^W + \vec{V}_C^W \quad (1)$$

This equation shows how the image visual features can be fused with the cameras motion parameters for the target velocity estimation.

In this paper, a simple two wheeled kinematic model of an autonomous vehicle is used to estimate the camera's velocity. Two tachometer sensors measure the wheels velocity. Since the vehicle only has the velocity in the X and Z directions as shown in Fig.2(a), the camera axis only has the Y axis rotation. Thus based on the process in Fig.2(b), the target velocity in the world coordinate \vec{V}_T^W can be estimated as

$$\begin{pmatrix} V_T^W x \\ V_T^W y \\ V_T^W z \end{pmatrix} = \begin{pmatrix} \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot x + S_x \cdot V_{O.F.x} \cdot b_1 \frac{f}{d}) \cdot \cos(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \\ \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot x + S_x \cdot V_{O.F.x} \cdot b_1 \frac{f}{d}) \cdot \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \\ + b_1 f \frac{\Delta d}{d^2} \cdot \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) + \frac{(k \cdot u_1 + k \cdot u_2) \cos(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t)}{2} \\ \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot y + S_y \cdot V_{O.F.y} \cdot b_1 \frac{f}{d}) \\ - b_1 f \frac{\Delta d}{d^2} \cdot \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) + \frac{(k \cdot u_1 + k \cdot u_2) \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t)}{2} \end{pmatrix} \quad (2)$$

Here (x, y) is the target 2D image coordinate, which is obtained from the previous ROI identification. S represent the effective sizes of the pixel in the image which transform the optical flow vectors $\vec{V}_{O.F.}$ to the image coordinate to get \vec{V}_{IM} . (u_1, u_2) are the tachometer output voltages, k is the linear parameter between the tachometer output voltages and the angular speeds of the wheels, which gives $v_1 = k \cdot u_1$ and $v_2 = k \cdot u_2$. b is the distance between two wheels. t is the time since the camera begins to move.

The target in the world coordinate position \vec{P}^W can be also obtained from (3).

$$\vec{P}^W = \begin{pmatrix} \cos(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \cdot \frac{S_x \cdot x \cdot Z}{f} - \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \cdot Z + t \cdot \frac{(k \cdot u_1 + k \cdot u_2) \cos(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t)}{2} \\ \frac{S_y \cdot y \cdot Z}{f} \\ \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \cdot \frac{S_x \cdot x \cdot Z}{f} + \cos(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t) \cdot Z + t \cdot \frac{(k \cdot u_1 + k \cdot u_2) \sin(\frac{k \cdot u_1 - k \cdot u_2}{2b} \cdot t)}{2} \end{pmatrix} \quad (3)$$

Equations 2 and 3 show the data fusion of target optical flow vectors $(V_{O.F.x}, V_{O.F.y})$, visual disparity d , image coordinate (x, y) and motion sensor data (u_1, u_2) to estimate the target 3D velocity and position in the world coordinate.

3 IMM Target Tracking System

At this stage an Extended Kalman Filter is used for the tracking of target position and velocity. Here, a rigid body with M points of motion is considered. Then the motion state equation can be written as Eq.(4).

$$\begin{pmatrix} \vec{P} \\ \vec{V} \end{pmatrix}_{k+1} = \Phi_r \begin{pmatrix} \vec{P} \\ \vec{V} \end{pmatrix}_k + \begin{pmatrix} 0 \\ n_v I \end{pmatrix}_k, \quad \vec{Z} = C(h)_k, \quad h = \{V_{O.F.x}; V_{O.F.y}; u_1; u_2; d; x; y\} \quad (4)$$

where Φ_r denotes the target dynamic models, $\vec{P} = (P_x, P_y, P_z)$ denotes the 3D position of the tracked point, $\vec{V} = (V_x, V_y, V_z)$ its velocity. The resulting position and velocity

measurement vector is \vec{Z} . Here C_k is the target position and velocity measurement function from Eq.(2) and Eq.(3). It is worth noting here that this measurement vector h comprise target visual features and vehicle motion sensor data as its different components, which provide a data fusion scheme for the target tracking.

The coordinate of the tracked point in the 2D image p can be obtained by (5)

$$\begin{pmatrix} P_x \\ P_y \end{pmatrix}_k = \Pi(\vec{P}_k) = \Pi(\Pi^{-1}(p_{k-1}, (P_z)_k)) \quad (5)$$

Where Π_{-1} denotes the inverse transformation of a 2D image point p and its depth P_z onto the 3D world coordinate space.

3.1 Fusion of Linear Dynamic Models for Kalman Filters

Several simple and basic dynamic models [4] are available for Extended Kalman Filtering.

1. Drifting Points Model. Drifting point model is represented by

$$\Phi_i = Id \quad (6)$$

That is the new position is same as the old position plus some Gaussian noise. This dynamic model is commonly used for objects with known dynamic behaviour.

2. Constant Velocity Model. p gives the position and v the velocity of a point moving with constant velocity. The constant velocity model is given in Eq.(7).

$$x = \begin{pmatrix} p \\ v \end{pmatrix}, \quad \Phi_i = \begin{pmatrix} Id & (\Delta t)Id \\ 0 & Id \end{pmatrix} \quad (7)$$

3. Constant Acceleration Model. a is the acceleration of a point moving with constant acceleration. Then the constant acceleration model is given in Eq.(8).

$$x = \begin{pmatrix} p \\ v \\ a \end{pmatrix}, \quad \Phi_i = \begin{pmatrix} Id & (\Delta t)Id & 0 \\ 0 & Id & (\Delta t)Id \\ 0 & 0 & Id \end{pmatrix} \quad (8)$$

4. Periodic Motion Model. It is assumed that a point is moving on a line with a periodic movement. Then stack the position and velocity into a vector $\vec{u} = (p, v)$ and with the time interval Δt , the motion model can be obtained with a forward Euler method as shown in Eq.(9).

$$\vec{u}_i = \vec{u}_{i-1} + \Delta t \frac{du}{dt} = \vec{u}_{i-1} + \Delta t S \vec{u}_{i-1} = \begin{pmatrix} 1 & \Delta t \\ -\Delta t & 1 \end{pmatrix} \vec{u}_{i-1} \quad (9)$$

Now with the above 4 linear dynamics models, one cycle of the IMM algorithm is described [2].

3.2 IMM Approach

First the prior probability that Φ_j is correct (the system is in Model j) is shown in Eq.(10).

$$P\{\Phi_j | Z^0\} = \mu_j(0) \quad (10)$$

where Z^0 is the prior information and $\sum_{j=1}^r \mu_j(0) = 1$.

1. Calculation of the mixing probabilities ($i, j = 1, \dots, r$). The probability that mode Φ_i was in effect at $k-1$ given that Φ_j is in effect at k conditioned on Z^{k-1} is Eq.(11).

$$\mu_{i|j}(k-1|k-1) = \frac{1}{\bar{c}_j} p_{ij} \mu_i(k-1), \quad \bar{c}_j = \sum_{i=1}^r p_{ij} \mu_i(k-1) \quad (11)$$

Here it is assumed that the model jump process is a Markov process (Markov chain) with known model transition probabilities p_{ij}

$$p_{ij} = P\{\Phi(k) = \Phi_j | \Phi(k-1) = \Phi_i\} \quad (12)$$

2. Mixing ($j = 1, \dots, r$). Starting with $\hat{x}^i(k-1|k-1)$ one computes the mixed initial condition for the filter matched to $\Phi_j(k)$ as

$$\hat{x}^{0j}(k-1|k-1) = \sum_{i=1}^r \hat{x}^i(k-1|k-1) \mu_{i|j}(k-1|k-1) \quad (13)$$

The covariance corresponding to the above is

$$P^{0j}(k-1|k-1) = \sum_{i=1}^r \mu_{i|j}(k-1|k-1) \{P^i(k-1|k-1) + [\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)][\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)]'\} \quad (14)$$

3. Model-matched filtering ($j = 1, \dots, r$). The estimate Eq.(13) and covariance Eq.(14) are used as input to the filter matched to $\Phi_j(k)$, which uses $z(k)$ to yield $\hat{x}^j(k|k)$ and $P^j(k|k)$. Then the likelihood functions corresponding to the r filters

$$\Lambda_j(k) = p[z(k) | \Phi_j(k), Z^{k-1}] \quad (15)$$

are computed using the mixed initial condition Eq.(13) and the associated covariance Eq.(14) as:

$$\Lambda_j(k) = p[z(k) | \Phi_j(k), \hat{x}^{0j}(k-1|k-1), P^{0j}(k-1|k-1)] \quad (16)$$

4. Model probability update ($j = 1, \dots, r$). This is done as follows:

$$\mu_j(k) = \frac{1}{c} \Lambda_j(k) \bar{c}_j, \quad c = \sum_{j=1}^r \Lambda_j(k) \bar{c}_j \quad (17)$$

where \bar{c}_j is the expression from Eq.(11).

5. Estimate and Covariance combination. Combination of the model-conditioned estimates and covariances is done according to the mixture equations

$$\hat{x}(k|k) = \sum_{j=1}^r \hat{x}^j(k|k) \mu_j(k) \quad (18)$$

$$P(k|k) = \sum_{j=1}^r \mu_j(k) \{P^j(k|k) + [\hat{x}^j(k|k) - \hat{x}(k|k)][\hat{x}^j(k|k) - \hat{x}(k|k)]'\} \quad (19)$$

3.3 IMM For Kalman Filter Estimation

According to the IMM, several simple linear dynamic models Φ introduced in the previous sections are used as the different state transition matrix (Eq.(4)) for the Kalman Filtering. In the IMM algorithm, the likelihood of each mode-conditioned estimates $\Lambda_j(k)$ in Eq.(16) are generated with the Kalman Filter measurement residue $residue_j(k)$

$$residue(k) = z(k) - C(k) * x(k|k-1) \tag{20}$$

where z is the measurement value, C is the measurement matrix and $x(k|k-1)$ is the Kalman Filter predict from $k-1$ to k . Then Eq.(16) can be written as

$$\Lambda_j(k) = N[residue_j(k); 0, S_j(k)] \tag{21}$$

where $S_j(k)$ is the residue variance which can be estimated from Eq.(22)

$$S_j(k) = C(k) * P^j(k|k-1) * C(k)^T + R(k) \tag{22}$$

here $R(k)$ is the covariance of the measurement noise.

Using the above recursive IMM algorithm, the interacting multiple linear dynamic model for Kalman Filter can be derived.

4 Experimental Results

A moving platform is designed with two sensors(Tachometers and CCD Cameras) to test the tracking algorithm. Tachometer measures the wheel speed of the platform, which is then used to estimate the speed of the camera with the aid of a kinematic model. Two color CCD cameras (focus length $f = 5.4\text{mm}$) are connected with the Hitachi IP5005 image processing cards for the image capturing and processing. The image size is 256×220 with a capturing rate $15f/s$.

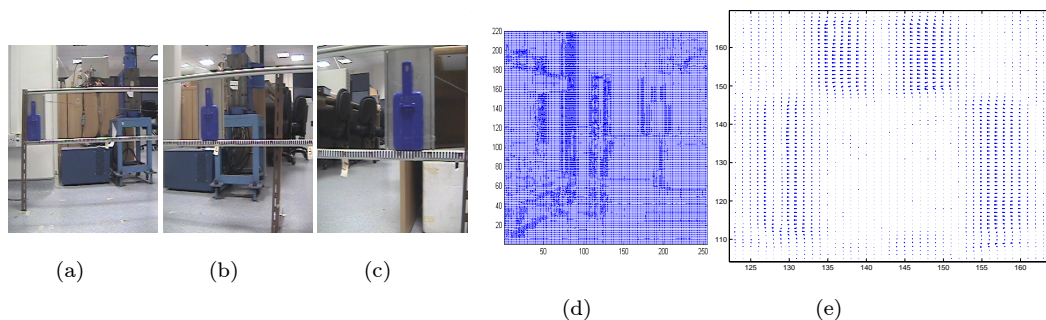


Figure 3: Moving Target with Moving Background

As shown in Fig.3 a blue target is moving in the image sequence. Here the cameras are also moving, as a result background is changing from frame to frame. Figures 3(d) and 3(e) show the estimated optical flow fields. The 100% dense optical flow field reflects the target motion together with the background motion. Fig.3(e) is the image with the enlarged optical flow vectors.

Next the K-means clustering is implemented to segment the flow vector fields into several regions(Fig.4). The K-means algorithm is sensitive to slight movements in the

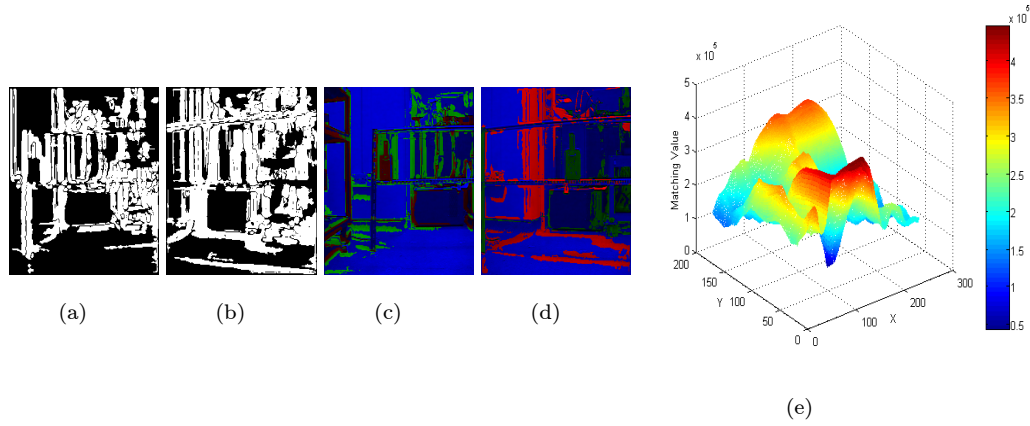


Figure 4: Motion Field Segmentation Result

image sequences. Therefore, it is necessary to eliminate the smaller motion vectors by defining a suitable threshold. Figures 4(a) and 4(b) are the segmented images after thresholding. Figures 4(c) and 4(d) show the K-means clustering result. Here, different colors represent different motion cluster regions. It is clear from the clusters that motion regions have the same color, thus it can be easily identified by a template matching technique. In this experiment the matching template is updated in real time through the target depth information. Fig.4(e) shows the template matching value. The smallest value indicates the region of interest.

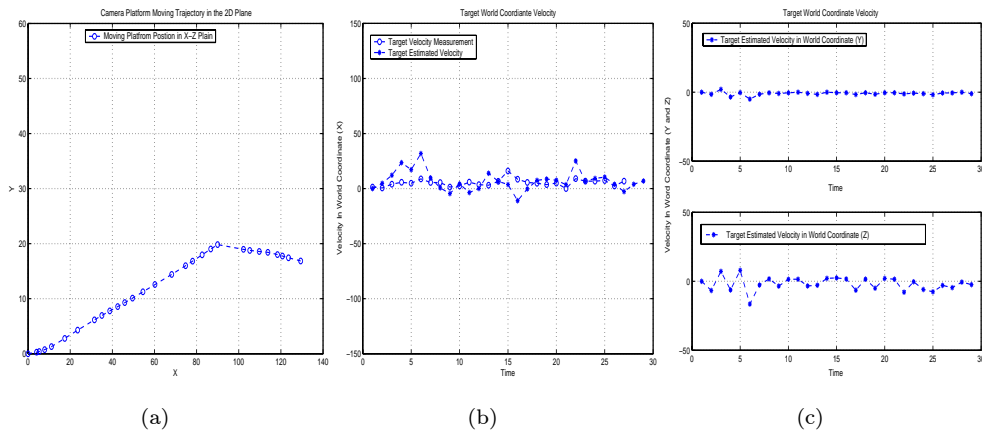


Figure 5: Target Velocity Estimation Result

Figure 5(a) shows the moving trajectory of the moving platform in the 2D plain,

which is estimated based on the two wheels' tachometer data and the kinematic model of the autonomous vehicle. This trajectory shows that the cameras are moving toward the targets. Figure 5(b) shows the target velocity in the X direction. The estimated velocity is quite close to the real target velocity. Figure 5(c) shows the target Y and Z direction velocity. In this experiment, the target has little movement in these direction, thus the values are almost zero.

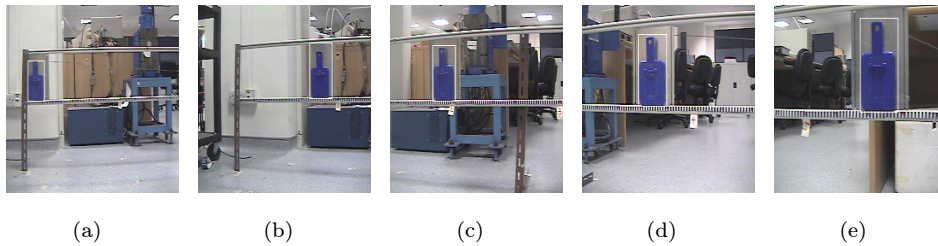


Figure 6: Target 3D Tracking Result with IMM

Figure 6 shows the 3D target tracking results. Here the rectangle in the images is obtained through the inverse pinhole camera projection of the estimated target world coordinate position from IMM based Extended Kalman Filtering (Eq.(5)). With the updating template, the size of the rectangle is scaled due to the 3D depth changes. From these images, the rectangle can accurately fit the target region which shows that the tracking system performs well.

As shown in Fig.7 the interacting multiple linear dynamics algorithm performs well for estimating the velocity and position of the target based on an Extended Kalman Filter. The errors from the IMM estimation are relatively smaller.

5 Conclusion

In this paper, a novel target-tracking scheme is proposed. Optical motion fields are used as the major source of information for tracking a moving object by autonomous vehicles. According to Forsyth[4] several simple and basic linear dynamic models are combined using an IMM algorithm to provide an accurate approximate model for the dynamics of the tracking target. Through experiments, the proposed tracking scheme demonstrated satisfactory results for dynamic target tracking. In conclusion it can be stated that the proposed target tracking scheme is suitable for target identification and tracking by autonomous vehicles.

References

- [1] J.L.Barron, D.J.Fleet, S.S.Beauchemin, T.A.Burkitt, "Performance of optical flow techniques " Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.236-242, 1992.
- [2] Y.Bar-Shalom, X.-R. Li, "Estimation and Tracking: Principles, Techniques, and Software" Artech House Boston MA, USA, 1993.

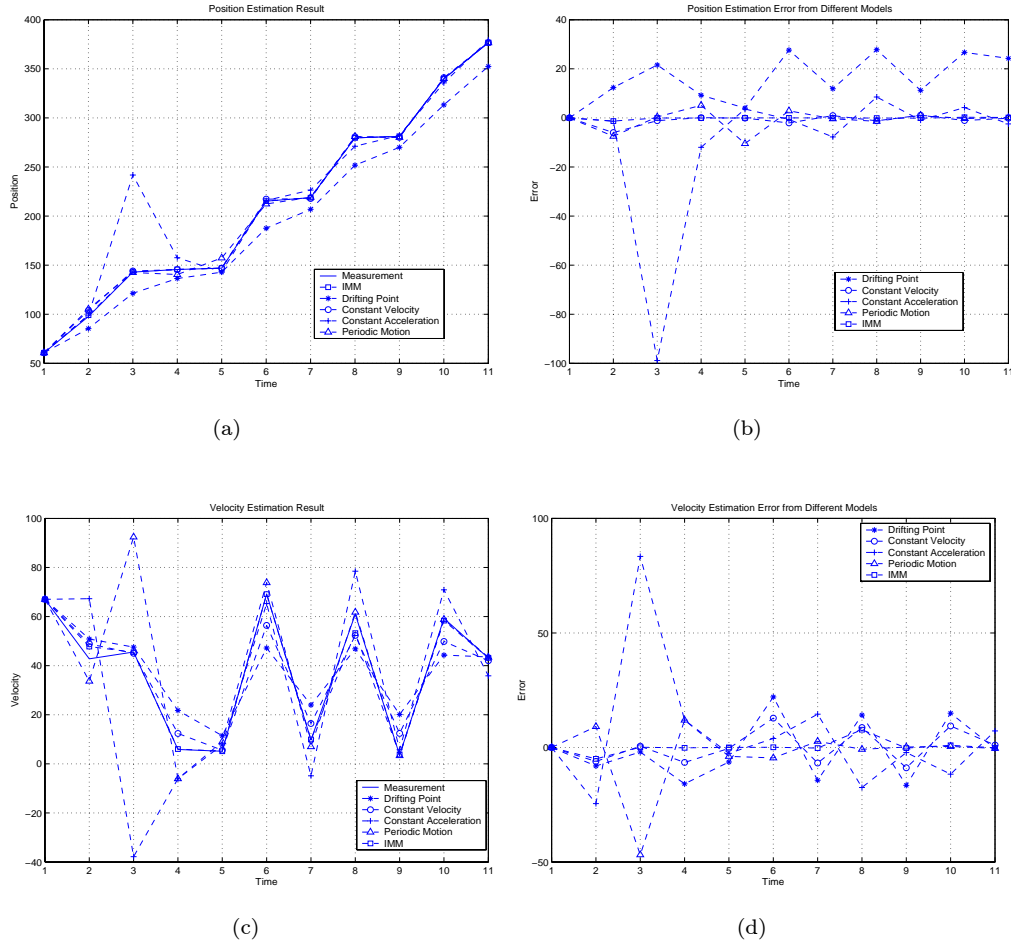


Figure 7: Estimation Result from Different Models

- [3] G.N.DeSouza, A.C.Kak, "Vision for Mobile Robot Navigation: A Survey" IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, No.2, pp.237-267, 2002.
- [4] D.A.Forsyth, J.Ponce, "Computer Vision: A Modern Approach (1st edition)" Prentice Hall, pp.523-527; pp.554-555, 2002.
- [5] B.K.P.Horn, "Robot Vision" MIT Press, pp.280-290; pp.401-403, 1986.
- [6] J.Evans, R.Evans, "Image-enhanced multiple model tracking" Automatica, 1999.
- [7] Zhen Jia, A.Balasuriya, S.Challa, "Vision data fusion for target tracking" Proceedings of the Sixth International Conference of Information Fusion, Volume:1, Pages:269-276, 2003
- [8] K.Muhlmann, D.Maier, R.Hesser, R.Manner, "Calculating dense disparity maps from color stereo images, an efficient implementation" Proceedings of the IEEE Workshop on Stereo and Multi-Baseline Vision, pp.30-36, 9-10 Dec.2001.