Available online at www.sciencedirect.com
ScienceDirect
Computer Vision and Image Understanding 109 (2008) 1–21
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Vision based data fusion for autonomous vehicles target tracking using interacting multiple dynamic models

Zhen Jia [a,*], Arjuna Balasuriya [b], Subhash Challa [c]

[a] United Technologies Research Center, Shanghai, 201206, P.R. China
[b] Department of Mechanical Engineering, MIT, Cambridge, MA 02139, USA
[c] Information and Communication Group, Faculty of Engineering, The University of Technology, Sydney, Australia

Received 29 March 2004; accepted 1 December 2006
Available online 23 December 2006

Abstract

In this paper, a novel algorithm is proposed for the vision-based object tracking by autonomous vehicles. To estimate the velocity of the tracked object, the algorithm fuses the information captured by the vehicle's on-board sensors such as the cameras and inertial motion sensors. Optical flow vectors, color features, stereo pair disparities are used as optical features while the vehicle's inertial measurements are used to determine the cameras' motion. The algorithm determines the velocity and position of the target in the world coordinate which are then tracked by the vehicle. In order to formulate this tracking algorithm, it is necessary to use a proper model which describes the dynamic information of the tracked object. However due to the complex nature of the moving object, it is necessary to have robust and adaptive dynamic models. Here, several simple and basic linear dynamic models are selected and combined to approximate the unpredictable, complex or highly nonlinear dynamic properties of the moving target. With these basic linear dynamic models, a detailed description of the three-dimensional (3D) target tracking scheme using the Interacting Multiple Models (IMM) along with an Extended Kalman Filter 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 fusion based IMM tracking algorithm is demonstrated through extensive experimental results.

Keywords: Optical flow; Extended Kalman Filtering; Image segmentation and clustering; Stereo vision; Target tracking; Autonomous vehicles; Linear dynamics model; Kinematic model; Interacting multiple models (IMM); Pinhole camera projection model; Template matching and updating; Sensor data fusion

1. Introduction

Computer vision for Autonomous Guided Vehicle (AGV) navigation has been an active research area for a very long time [8]. This paper proposes a vision based tracking algorithm whose potential applications are: autonomous vehicles target following and obstacle avoidance [8]. For such applications the tracking system is expected to provide the complete 3D information of the target in the world coordinate with a moving environment, i.e., the target's size, location, velocity, and depth (distance) [10].

In most of the proposed target tracking algorithms, autonomous vehicles burden the target with special equipments, for example the intelligent space (ISpace) from [23]. Generally in some unknown environments or outdoor situations, such special sensors may not be available for experimental testing. Thus in order to achieve a more robust target tracking, the vehicle's on-board sensors should estimate the motion parameters of the tracking object accurately. These sensors detect the relative dynamics between the autonomous vehicle and target.

Using on-board sensors for target tracking imposes four major challenges [8].

- First, as the vision sensors (CCD stereo cameras pair) are moving with the vehicle, the background changes continuously. With a changing background, it is hard to identify the object's dynamics based only on the vision sensors.

* Corresponding author. Fax: +86 21 58998020.
E-mail address: jiazhen@pmail.ntu.edu.sg (Z. Jia).

1077-3142/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2006.12.001

- Second, for autonomous vehicles in the outdoor environment, the optical properties of the target may be unpredictable or inconsistent (such as the changes in the environmental lighting condition) making the feature extraction extremely difficult.
- Third, previous visual target tracking systems estimate the relative position and velocity of the object to the vehicle. In order to achieve more general applications with the target's dynamic information and obtain more accurate estimation results, the target's 3D velocity and position need to be estimated in the world coordinate.
- Fourth, natural moving targets often have unpredictable, complex or highly nonlinear motion dynamics. Commonly used linear models will not produce these dynamic properties.

The motivation of this paper is to develop a data fusion based target tracking system to solve the above problems. The novelty of this paper is the combination of different data fusion techniques to achieve a reliable tracking system. Normally data fusion can be achieved either by using multiple (redundant) sensors or by combining several measurements of a single sensor [30]. In this paper, three different data fusion steps are developed and integrated together to formulate the novel tracking scheme.

- First, different visual features from CCD cameras are fused to accurately extract the region of interest (ROI) from the image sequence with a changing background.
  Most of the vision-based tracking schemes depend on optical properties of the object of interests [31]. This creates practical difficulties when these features change in the scene, which is the case with objects in complex environments such as underwater. Thus the aim of this paper is to use motion fields in the scene rather than using any special property of the object for target identification and tracking, which is quite challenging.
  Optical flow vectors reflect the motion of target pixels in the image sequence, and it requires a minimal prior knowledge of the observed environment. From the previous research work [7,18,26,32,33], by fusing optical flow vectors with the target's other visual features, a reliable object tracking system could be designed to track the object's 3D state. Also optical flow vectors can be employed as the velocity measurements for tracking the boundaries of the nonrigid objects such as the method of Kalman Snakes [28]. In this paper optical flow vectors are used as the main source for the target's 3D features identification [18]. First, optical flow vectors are fused with the image various visual features to segment the motion fields, which is used to identify the region of interest (ROI) from background motion fields. Next, optical flow vectors are fused with the motion sensors data to estimate the target's 3D velocity and position.
- Secondly, the vision sensor data are fused with the vehicle's inertial motion sensors data for estimating the target's position and velocity in the 3D world coordinate frame.
  In this paper the kinematic model of the autonomous vehicle is used to estimate the velocity of the cameras' motion based on the vehicle's inertial sensors. The kinematic model is able to provide information such as self rotation/translation about/along the optical axis. Next optical flow vectors are fused with the moving cameras' motion parameters based on the motion compensation approach [5]. In such a way, the target's 3D velocity and position are estimated in the world coordinate for the MCMO target tracking.
- Thirdly, several simple and basic linear dynamic models are fused with the multiple models algorithm to approximate the target's complex motion properties for an accurate target tracking.
  In visual target tracking, the dynamic information aids the tracking process in the presence of occlusions and measurement noises [16]. However, the tracking is complicated by the fact that natural moving targets do not exhibit one type of motion but rather have complex, unknown, highly nonlinear or time-varying dynamics. The single model approaches do not make full use of the information available and often rely on somewhat ad hoc methods of incorporating uncertainty about the mode estimates (like adjusting the filter noise covariance in proportion to the variances in the model estimate [18]), so the traditional single linear dynamic models are not quite applicable. Tracking of such targets falls into the area of adaptive state estimation (multiple models methods) [9,19].
  The distinguishing feature of multiple models approaches is a stochastic (typically Markov) process switching between the models and the simultaneous filtering of the observations through all of the possible systems. The final state estimate is obtained as a weighted combination of the outputs of estimates from each different model, which is in contrast to single model algorithms that tend to use one model at any given time.
  Recently Forsyth et al. [13] reviewed the linear dynamic models for target tracking. These models are quite simple and easy to implement and cover most of the target's linear dynamic properties. In the proposed tracking system, these simple linear dynamic models are combined together with the multiple models algorithm to represent the target's complex motion properties.
  Research works have been carried out in target tracking using Multiple Models (MM) filtering. Bar-Shalom and Li [1] reviewed the models in which both the state dynamics and output matrices switch, and the switching follows the Markovian dynamics (IMM). They have also presented several different methods for approximately solving the state-estimation problems in switching models. Some other multiple models tracking systems use the Bayesian network based multi-model fusion methods

[11], and [21]. More recently, switching linear dynamic system (SLDS) models have been studied in [3]. SLDS models and their equivalents have also been studied in statistics, time-series modelling, and target tracking since the early 1970's in [27]. It is concluded by Bar-Shalom [1] that the IMM algorithm performs significantly better than other multiple models methods. It computes the state estimates under each possible current model using several filters, with each filter using a different combination of the previous model-conditioned estimates (mixed initial condition). The mixed initial condition of IMM can reduce the estimation errors by adjusting the initial condition for each estimate. IMM is also faster in computation than other multiple models algorithms, for example IMM requires only 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 has been shown to be able to keep the estimation errors not worse than the raw measurement errors and provide significant improvements (noise reduction).

In this paper the IMM approach is developed to fuse several simple and basis linear dynamic models for the target dynamic information tracking. The interacting multiple linear dynamic models algorithm provides an approximation of the types of the target's employed motion, and then the target's dynamic state estimates are obtained with the combination of the weighted different models' Extended Kalman Filter estimates, which can improve the tracking performance. The Interacting Multiple Models (IMM) algorithm here also allows the system to minimize tracking errors by adjusting the gain of the Extended Kalman Filter.

The rest of this paper is organized as follows: In Section 2, the proposed tracking system is described. Section 3 talks about the data fusion based K-means image segmentation and clustering and template matching algorithms. In Section 4, optical flow vectors are fused with the target's depth disparities information and then combined with the kinematic model of the camera system to estimate the target's $3D$ position and velocity in the world coordinate. With the target dynamic features, several simple linear dynamic models with the IMM algorithm are used to conduct the target Extended Kalman Filter tracking in Section 5. Section 6 presents the extensive experimental results obtained by nature image sequences. Section 7 concludes the research work in this paper and makes suggestions for future work.

2. Proposed tracking system

The proposed tracking scheme has the ability to track the target's 3D dynamics in an unknown environment in the Moving Camera and Moving Object (MCMO) state [25]. From the literature reviews [15], the prerequisites for effectively tracking the moving target using CCD cameras generally include detection of features (feature segmentation and classification) and target tracking. In such a way, the novel tracking methodology is designed based on the strategies shown in Fig. 1.

In the first step, the stereo images are captured from CCD cameras. In order to accurately identify the target from the image sequences with a changing background, the image motion properties (optical flow vectors) are extracted from the image sequence. Since the system is designed to track the target's 3D dynamics, the target's

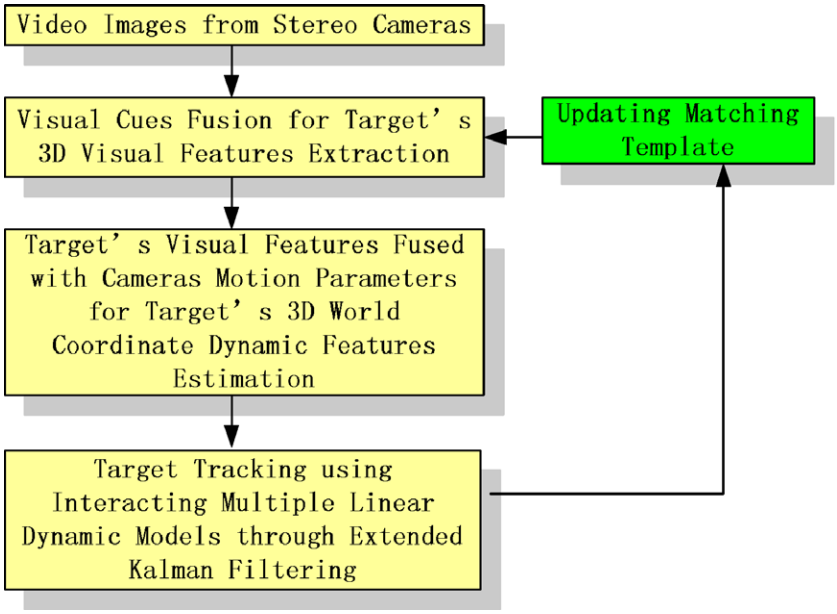

Fig. 1. The proposed tracking system.

3D visual depth is estimated from stereo images. In the second step, because different objects have different visual features, for example the motion properties and there are also background features together with the target's features, these visual features are classified and localized by the image processing algorithms such as image clustering and template matching to find the useful ones. In the third step, optical flow vectors are fused with the target's depth disparity information and then combined with the kinematic model of the camera system to estimate the target's 3D position and velocity in the world coordinate. These information is used later for the target dynamics tracking. In the final step, the target 3D dynamics is estimated with Extended Kalman Filter. Through the Extended Kalman Filter measurement equation, sensor data fusion is achieved for the proposed tracking scheme. In the final step, several simple and basic linear dynamic models with the IMM approach are used to make the approximation of the target complex motion properties.

3. Fusion based image segmentation and clustering and template matching

In this paper, the image motion fields are clustered to find the major motion regions. As long as the motion region's optical flow vectors are not zero, they can be successfully clustered out from the image sequence even if they are less accurately estimated. Therefore, the most commonly used Horn and Schunck's differential (gradient-based) technique is tested for the calculation of optical flow due to its fast computational speed, easiness to implement, 100% density and accuracy [2,14]. At the same time, the disparity (denoted as d) is estimated from the stereo images. This paper chooses the normally used and simple Sum of the Absolute RGB Differences (SAD) algorithm to estimate the disparity d. The Sum of Absolute RGB Differences (SAD) performs better than the normally used Sum of Squared Differences in the presence of outliers [24]. Using color information instead of gray values improves the signal to noise ratio by 0.2–0.5.

Next, in the case of target tracking by autonomous vehicles, one big challenge is to separate the optical flow vectors field of the target from the background motion field. Assuming that the moving target is a rigid object with uniform motion properties within its body (the assumption is quite reasonable for the normal target tracked by autonomous vehicles.), the target region can be identified from the background with the motion vectors clustering and segmentation [18].

Recently some research works have been done on the motion based image segmentation [4,6,12]. These kinds of work do the optical flow estimation and image segmentation simultaneously. As stated earlier, the optical flow estimation is based on the classic Horn and Schunck's method. The image segmentation needs a separate and simple method. Therefore the well-known K-means algorithm is applied to segment the optical flow vectors field by dividing the image scene into different clusters. The feature set used in the K-means algorithm is given in Eq. (1).

$$\vec{x} = [v_x, v_y, x, y, R, G, B] \tag{1}$$

where x, y are the pixel's locations in the 2D images, v_x and v_y are the optical flow vectors and R, G, B are the intensities of the color components at that pixel point. The RGB color space instead of the HSV color space is used here because the RGB color values can be read directly from the image and it is much easier for the algorithm implementation.

The distance between two feature vectors is measured using a Mahalanobis distance, defined by

$$d(\vec{x}_1, \vec{x}_2) = \sum_{\text{K-means}} (\vec{x}_1 - \vec{x}_2)^{\mathrm{T}} S^{-1} (\vec{x}_1 - \vec{x}_2) \tag{2}$$

where $\sum_{\text{K-means}} = \text{diag}(1,1,1,1,2,2,2)$. S^{-1} is the inverse of the variance–covariance matrix of $\vec{x}$. The purpose of the matrix $\sum$ is to weight the color intensities since they are of different scales (and units) from the other 4 components in the feature vector. The color features can be generally weighted more than the others since normally the color information is more apparent and is the dominating visual feature [18]. After some trial-and-error, $\sum_{\text{K-means}} = \text{diag}(1,1,1,1,2,2,2)$ in $\sum$ for the RGB components is found to give good results.

Here, the target's optical flow vectors are fused with the color and spatial information to make the image segmentation and clustering more robust. The object is assumed to be a rigid one with uniform motion. Integration of color with motion fields solves some of the image segmentation problems. For example, if the object is a textured one with different visual color properties, the K-means algorithm still can make an accurate clustering based on the object's uniform motion properties from the image sequences.

In this paper, 5 and 3 clusters are generally selected for the natural video images K-means clustering. This number can be determined automatically for video images segmentation by incorporating the intra-cluster and inter-cluster distance validity measures developed by [29]. There is a tendency to select smaller cluster numbers for natural images, which is due to the inter-cluster distance being much greater and greatly affecting the validity measures. The normal smallest number of clusters that can be selected is 4 [29]. In fact the clusters' number may be changed. With more image clusters, the video images segmentation may be more accurate while at the same time the computational time will also increase. Also the number of clusters in the experiments can be adapted to the contents of the video frames. For example, a small number of clusters for the object based image segmentation of frames with 10 objects is too ambiguous, even though the object of interest can be well covered. So when the background is very cluttered and there are more objects like 8–10 in the image, 8–9 clusters would be more appropriate. If there are fewer major objects like 3–5 or the environment is structured, 3–5 clusters would be all right.

After the image segmentation and clustering, the major motion regions are left in the clustered optical flow vectors field. The smaller motion regions considered as noise are removed. A template is defined and correlated with the clustered optical flow vectors fields to extract the ROI from the background motion field. The template is the target's region in the 2D clustered optical flow vectors field. The matching is done with the simple cost function by the sum of squared differences (SSD) between the template and the whole clustered motion vectors field.

$$\mathrm{SSD} = \sum ||T - I||_2 \tag{3}$$

where T is the template and I is the clustered optical flow vectors field. The minimization of the SSD (Eq. (3)) is obtained by sequentially scanning the whole optical flow vectors field and finding the minimum value of Eq. (3). The coordinates of the ROI are considered as the 2D position of the target in the image, which are used in the next section for estimating the target's 3D dynamics in the world coordinate.

The template consists of only optical flow vectors, thus the extraction of the ROI's 2D position is carried out only with the target's dynamic features, which can make the tracking system robust to localize targets independent of their other visual features such as shape or color. For example, if a target is moving in the images and its color changes, template matching using color information may go wrong. In such a situation, the target's motion is still the same, so the template matching by optical flow vectors can give better results. Also using the 2D motion vectors instead of the image's 3D *RGB* color information reduces the whole system's computational cost.

At present, the initial template can be automatically identified from the clustered optical flow motion field if there is only one target moving or only one new object entering the image scene for tracking. If there are several objects moving together in the image, the initial template must be selected manually from the first clustered optical flow vectors field because the tracking algorithm does not have the intelligence to tell which object is of interest. In Section 6, the tracker's automatic initialization, termination or multiple people tracking are discussed with experimental results. After the matching template is initialized, the matching value (Eq. (3)) should be very small when the target is moving in the image scene. If the matching value is much larger than a preset threshold, it means that the target has exited and the tracker should be automatically terminated. During the tracking process, the template is updated frame by frame with the target's previous estimated 3D dynamic information. This updating process is known as the "naive update" template updating algorithm [22].

4. Sensor data fusion for target's 3D dynamic information estimation

Once the target is identified in the image scene, it is then necessary to estimate its velocity and position to track it by autonomous vehicles. This section presents the proposed estimation procedures.

In this paper, the target's 3D position and velocity in the world coordinate are estimated for the visual target tracking system. In the previous autonomous vehicles target tracking systems, the target's relative dynamic information from CCD cameras is used directly for the navigation purpose. There are several drawbacks using the relative information in autonomous vehicles target tracking. First, these data are only relative from the vision sensors to the target, which are not available for other applications in the same environment. Secondly, in such a situation the vehicle's self rotational and translational motion in the world coordinate are generally not considered, which makes the target dynamics estimation less accurate. So in the proposed visual target tracking scheme, in order to achieve more general applications and get more accurate estimation results than the methods like [7,8,33], the target's 3D velocity and position are estimated in the world coordinate. For this purpose, the cameras' motion parameters should be known and therefore the moving vehicle's kinematic model should be integrated in the tracking system. With the kinematic model of the vehicle, tracking problems involving the cameras' 3D motion can also be solved.

Let's first consider the cameras mounted on an autonomous vehicle moving in the 3D world coordinate as shown in Fig. 2a. Estimation of the target's velocity can be achieved using the algorithm explained in Fig. 2b. Here, the motion of the target in the image stream is obtained as optical flow vectors $\vec{V}_T^{O.F.}$. Then $\vec{V}_T^{O.F.}$ is transformed into the image coordinate frame as $\vec{V}_T^{IM}$. Next $\vec{V}_T^{IM}$ is transformed into the camera frame to get the target velocity relative to the cameras $\vec{V}_T^{C}$.

Because the cameras are moving in the world coordinate, its axis has the rotational motion R_C^W and translational motion T_C^W relative to the world coordinate. The velocity of the moving cameras in the world coordinate $\vec{V}_C^W$ (consists of R_C^W and T_C^W) can be estimated based on the kinematic model of the autonomous vehicle. Then the target's velocity in the world coordinate $\vec{V}_T^W$ can be estimated as:

$$\vec{V}_T^W = R_C^W \cdot \vec{V}_T^C + T_C^W \tag{4}$$

This equation shows how the visual features can be fused with the cameras' motion parameters for estimating the target's velocity.

The following section first explains how $\vec{V}_T^C$ is estimated. Here, the 2D visual cues are fused with the disparity information using a pinhole camera projection model (Fig. 3 and Eq. (5)) to obtain the target's 3D velocity relative to the cameras.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{5}$$

The time derivation of Eq. (5) gives the velocity.

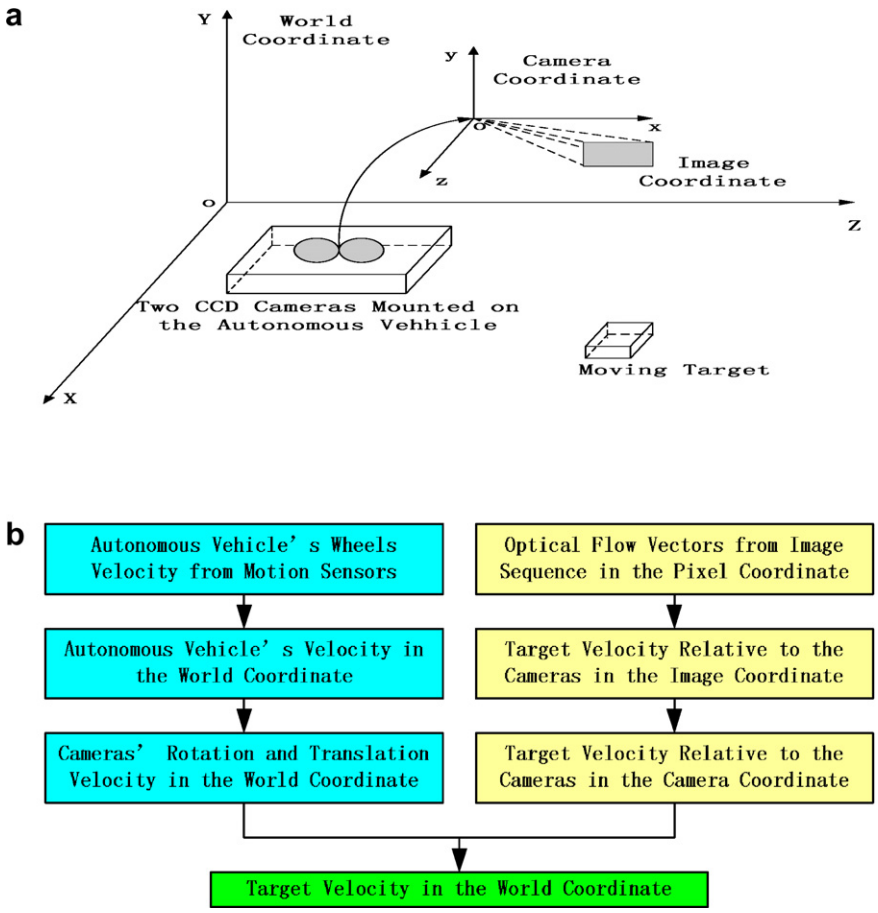

Fig. 2. Experiment setup with the target's 3D world coordinate velocity estimation procedures.

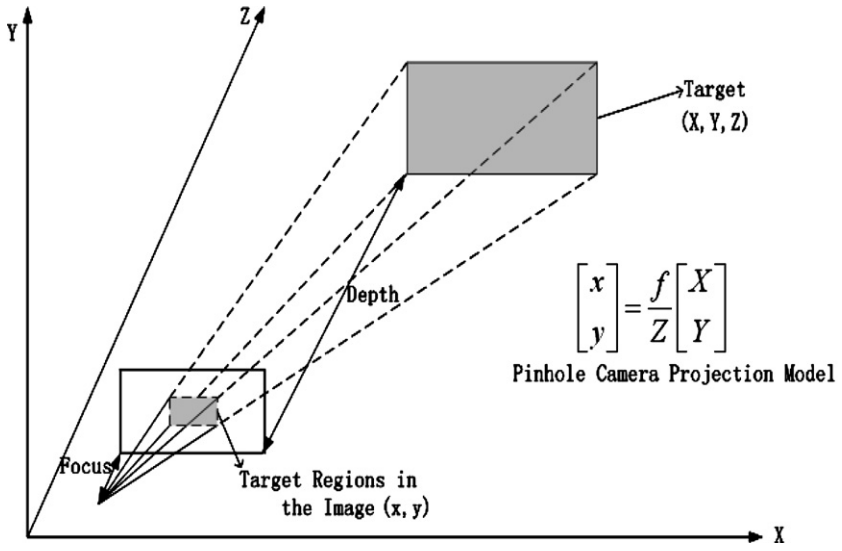

Fig. 3. The pinhole camera projection model.

$$X = \frac{Z}{f}x \Rightarrow \dot{X} = \frac{\dot{x}\cdot Z + \dot{Z}\cdot x}{f} \tag{6}$$

where $\dot{X}$ is the target's velocity relative to the camera frame known as $\vec{V}_T^C$ and $\dot{x}$ is the target's image pixel velocity $\vec{V}_T^{IM}$, which can be estimated from the optical flow vectors $\vec{V}_T^{O.F.}$ (Eq. (7)).

$$\vec{V}_T^{IM} = S\cdot\vec{V}_T^{O.F.} \tag{7}$$

where S is the effective size of the pixels in the image, which transforms the image pixel velocity (optical flow vectors) into the real image coordinate displacement. In Eq. (6), Z is the target depth information, which can be estimated from the disparity d. The deviation of the depth $\dot{Z}$ is obtained as $\dot{Z} = -b_1 f \frac{\Delta d}{d^2}$, Δd is the difference of the disparity d between two time stamps. From Eqs. (5) and (7), $\vec{V}_T^C$ can be obtained as in Eq. (8)

$$\vec{V}_T^C = \begin{pmatrix} V_{Tx}^C \\ V_{Ty}^C \\ V_{Tz}^C \end{pmatrix} = \begin{pmatrix} \frac{1}{f}(\dot{Z}\cdot x + V_{T\ x}^{IM}\cdot Z) \\ \frac{1}{f}(\dot{Z}\cdot y + V_{T\ y}^{IM}\cdot Z) \\ \dot{Z} \end{pmatrix} = \begin{pmatrix} \frac{1}{f}(-b_1 f\frac{\Delta d}{d^2}\cdot x + S_x\cdot V_{T\ x}^{O.F.}\cdot b_1\frac{f}{d}) \\ \frac{1}{f}(-b_1 f\frac{\Delta d}{d^2}\cdot y + S_y\cdot V_{T\ y}^{O.F.}\cdot b_1\frac{f}{d}) \\ -b_1 f\frac{\Delta d}{d^2} \end{pmatrix} \quad (8)$$

where (x, y) is the target 2D image coordinate, which can be estimated from the previous ROI identification. The stereo cameras' optical axes are parallel and separated by a distance b_1.

In order to get the cameras' rotation and translation matrixes R_C^W and T_C^W, the cameras' movement in the world coordinate should be known. In this paper a simple two-wheel kinematic model for autonomous vehicles is used for the experimental testing (Fig. 4). Two tachometers measure the wheels' velocities $v1$ and $v2$. Here $u1$, $u2$ are the tachometers' output voltages, and k_T is the linear parameter between the tachometers' output voltages and the wheels velocities, which gives the simple relationship $v_1 = k_T \cdot u_1$ and $v_2 = k_T \cdot u_2$. In order to simplify the tracking system, it is just assumed that the wheels' slippage can be ignored. The vehicle's velocity along X and Z directions in Fig. 4 can then be estimated as Eq. (9).

$$\begin{pmatrix} v_X \\ v_Z \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \frac{(v_1+v_2)\cos(\theta)}{2} \\ \frac{(v_1+v_2)\sin(\theta)}{2} \\ \frac{v_2-v_1}{2b} \end{pmatrix} = \begin{pmatrix} \frac{(k_T\cdot u_1+k_T\cdot u_2)\cos(\theta)}{2} \\ \frac{(k_T\cdot u_1+k_T\cdot u_2)\sin(\theta)}{2} \\ \frac{k_T\cdot u_2-k_T\cdot u_1}{2b} \end{pmatrix} \quad (9)$$

where b is the distance between the two wheels and θ is the angle between the X axis and the vehicle's velocity in Fig. 4.

In real world applications, there may be many changes in the vehicle's attitude. In order to simplify the algorithm and make the estimation more accurate, the vehicle's motion is assumed to be on a flat ground. Therefore, the velocities in the X and Z directions (Fig. 2a) are considered as the most effective movements of the vehicle. The vehicle's velocity in the Y-coordinate is too small compared with those of the other two directions and it can be ignored. The camera coordinate has only the Y axis rotation relative to the world coordinate and the rotational velocities in the other two directions can be neglected. Based on this assumption, the whole system becomes much less complex and the following experimental results are still good. The whole computation is simplified in a reasonable way. Let θ denote the angle between the camera velocity and X direction, then the camera axis rotation matrix R_C^W can be estimated as:

$$R_C^W = \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix}; \quad \theta_k = \theta_{k-1} + \Delta t\cdot\dot{\theta} \quad (10)$$

Here Δt is the time interval from the time $k-1$ to time k to estimate θ from $\dot{\theta}$ in Eq. (9). In the following equations, the time k for θ is omitted because all the estimations are done at the same time stamp k.

At the same time the vehicle's translational vector in the world coordinate can be estimated as T_C^W. The vehicle only has velocities in the X and Z directions. Based on the autonomous vehicle's kinematic model, T_C^W can be written as Eq. (11):

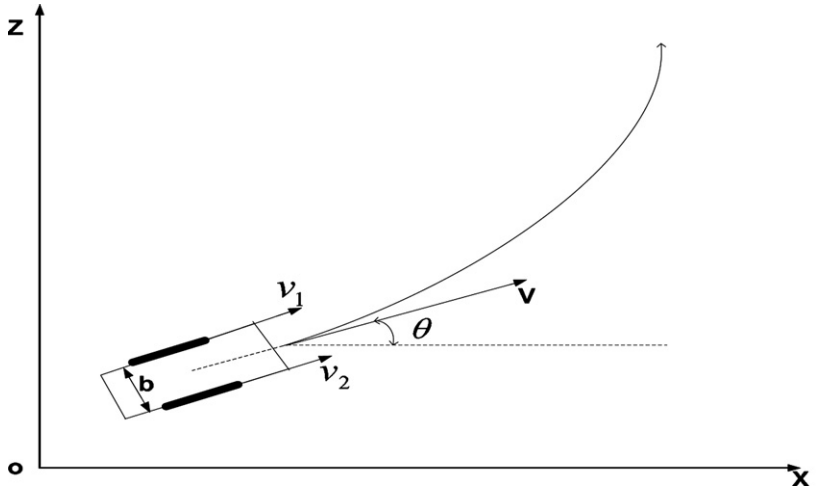

Fig. 4. The kinematics model of autonomous vehicle.

$$T_C^W = \begin{pmatrix} T_{C\,x}^W \\ T_{C\,y}^W \\ T_{C\,z}^W \end{pmatrix} = \begin{pmatrix} \frac{(k_T \cdot u_1 + k_T \cdot u_2)\cos(\theta)}{2} \\ 0 \\ \frac{(k_T \cdot u_1 + k_T \cdot u_2)\sin(\theta)}{2} \end{pmatrix} \tag{11}$$

Finally from Eq. (4), (8), (10) and (11), the target's 3D velocity $\vec{V}_T^W$ in the world coordinate can be estimated as

$$\begin{pmatrix} V_{T\,x}^W \\ V_{T\,y}^W \\ V_{T\,z}^W \end{pmatrix} = \begin{pmatrix} \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot x + S_x \cdot V_{T\,x}^{O.F.} \cdot b_1 \frac{f}{d}) \cdot \cos(\theta) - b_1 f \frac{\Delta d}{d^2} \cdot \sin(\theta) + \frac{(k_T \cdot u_1 + k_T \cdot u_2)\cos(\theta)}{2} \\ \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot y + S_y \cdot V_{T\,y}^{O.F.} \cdot b_1 f \frac{1}{d}) \\ \frac{1}{f}(-b_1 f \frac{\Delta d}{d^2} \cdot x + S_x \cdot V_{T\,x}^{O.F.} \cdot b_1 \frac{f}{d}) \cdot \sin(\theta) + b_1 f \frac{\Delta d}{d^2} \cdot \cos(\theta) + \frac{(k_T \cdot u_1 + k_T \cdot u_2)\sin(\theta)}{2} \end{pmatrix} \tag{12}$$

The target's 3D position in the world coordinate $\vec{P}_T^W$ can also be obtained from Eq. (13).

$$\vec{P}_T^W = R_C^W \cdot \vec{P}_T^C + T_C^W \tag{13}$$

Here R_C^W is the camera axis's rotational matrix (Eq. (10)). $\vec{P}_T^C$ is the target position relative to the cameras in the camera frame. T_C^W is the translational vector between the camera coordinate and the world coordinate (Eq. (11)). So $\vec{P}_T^W$ can be written as:

$$\vec{P}_T^C = \begin{pmatrix} P_{T\,x}^C \\ P_{T\,y}^C \\ P_{T\,z}^C \end{pmatrix} = \begin{pmatrix} \frac{S_x \cdot x \cdot Z}{f} \\ \frac{S_y \cdot y \cdot Z}{f} \\ Z \end{pmatrix}$$

$$\vec{P}_T^W = \begin{pmatrix} \cos(\theta) \cdot \frac{S_x \cdot x \cdot Z}{f} - \sin(\theta) \cdot Z + t \cdot \frac{(k_T \cdot u_1 + k_T \cdot u_2)\cos(\theta)}{2} \\ \frac{S_y \cdot y \cdot Z}{f} \\ \sin(\theta) \cdot \frac{S_x \cdot x \cdot Z}{f} + \cos(\theta) \cdot Z + t \cdot \frac{(k_T \cdot u_1 + k_T \cdot u_2)\sin(\theta)}{2} \end{pmatrix} \tag{14}$$

These equations (Eqs. (12) and (14)) show the data fusion of the target's optical flow vectors $(V_{T\,x}^{O.F.}, V_{T\,y}^{O.F.})$, visual disparity d, image coordinates (x,y) and motion sensors data (u_1,u_2) into one equation to estimate the target's 3D velocity and position in the world coordinate. These two equations are used later to obtain the Extended Kalman Filter measurement function for estimating the target dynamics, which is the proposed data fusion scheme to track the target dynamic states.

5. Multiple models fusion based target tracking system

At this stage, the position and velocity of the target are estimated together using a Kalman Filter. Since the velocity and position measurement models of the target are obtained from the inverse calculations of Eq. (12) and Eq. (14), the derived Kalman Filter will always contain nonlinear components. Therefore, an Extended Kalman Filter is used to cater for the nonlinearities present. In the tracking exercise the tracked body is assumed to be a rigid one. Let $\vec{X}$ denote the target's dynamic state, $\vec{X} = \begin{pmatrix} \vec{P} \\ \vec{V} \end{pmatrix}$. Then the state equation for its movement can be written as Eq. (15).

$$\vec{X}_{k+1} = \Phi_r \vec{X}_k + v_k \tag{15}$$

where Φ_r denotes the target dynamic models, $\vec{P} = (P_x, P_y, P_z)^{\mathrm{T}}$ denotes the 3D position of the tracked point, and $\vec{V} = (V_x, V_y, V_z)^{\mathrm{T}}$ denotes its velocity. v_k is the Gaussian white noise. v_k has a 6×6 covariance matrix and over the increment from the time k to time $k-1$, this process noise enters into both the position and velocity states.

The coordinate of the tracked point in the 2D image p can be obtained from the 3D world coordinate by Eq. (16)

$$\begin{pmatrix} P_x \\ P_y \end{pmatrix}_k = \Pi(\vec{P}_k) \tag{16}$$

In the above equation, Π denotes the camera's pinhole projection transformation of a 3D world coordinate point $\vec{P}$ onto the 2D image coordinate space (P_x, P_y).

The resulting position and velocity measurement vector Z for the system can be obtained as given in Eq. (17)

$$\vec{Z}_k = H(k, \vec{X}_k)_k + w_k, \quad \vec{Z}_k = \begin{pmatrix} V_{O.F.x} \\ V_{O.F.y} \\ u_1 \\ u_2 \\ d \\ x \\ y \end{pmatrix} \tag{17}$$

$\vec{Z}_k$ represents the measurements of the target from different sensors. H_k is the measurement function, which can be obtained from Eq. (12) and Eq. (14). It is worth noting that this measurement vector $\vec{Z}_k$ and measurement function $H(h)_k$ comprise of the optical flow vectors $(V_{T\,x}^{O.F.}, V_{T\,y}^{O.F.})$, disparity d, tachometer sensors data (u_1,u_2) and image spatial information (x,y) into one equation. w_k is the measurement noise, which has a covariance matrix of 7×7. This equation (Eq. (17)) provides a scheme to fuse different visual cues with the cameras' motion parameters as the inputs to EKF for tracking the target's 3D dynamics as the outputs from EKF.

5.1. Linear dynamic models for extended Kalman filter

There are several simple and basic linear dynamic models available for Extended Kalman Filter to cover the normal dynamic situations of the moving target [13], which cover all the linear dynamic properties of the moving object. The dynamic model is achieved by multiplying the state with some known matrix Φ as in the motion state equation (Eq. (15)), and then adding a normal random variable of a zero mean and known covariance. The work in this paper is to apply the IMM algorithm to fuse these simple and basic models to approximate the complex and nonlinear properties of the tracking system, which has not been

done by other researchers and also is the novelty of this paper.

1. *Drifting Points Model.* If the dynamic model is
$$\Phi_i = I \tag{18}$$
where I is the identity matrix. Then the point's new position is its old position, plus some Gaussian noise terms. This drifting points dynamic model can be commonly used for the object where no better dynamic model is known.
2. *Constant Velocity Model.* p gives the position and v gives the velocity of a point moving with the constant velocity. The constant velocity model is given in Eq. (19).
$$\vec{X} = \begin{pmatrix} p \\ v \end{pmatrix}, \quad \Phi_i = \begin{pmatrix} I & (\Delta t)I \\ 0 & I \end{pmatrix} \tag{19}$$
3. *Constant Acceleration Model.* a is the acceleration of a point moving with a constant acceleration. Then the constant acceleration model is given in Eq. (20).
$$\vec{X} = \begin{pmatrix} p \\ v \\ a \end{pmatrix}, \quad \Phi_i = \begin{pmatrix} I & (\Delta t)I & 0 \\ 0 & I & (\Delta t)I \\ 0 & 0 & I \end{pmatrix} \tag{20}$$
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 periodic motion model can be obtained with a forward Euler method as shown in Eq. (21).
$$\vec{u}_i = \vec{u}_{i-1} + \Delta t \frac{\mathrm{d}u}{\mathrm{d}t} = \vec{u}_{i-1} + \Delta t S \vec{u}_{i-1} = \begin{pmatrix} 1 & \Delta t \\ -\Delta t & 1 \end{pmatrix} \vec{u}_{i-1} \tag{21}$$

5.2. Interacting multiple models approach

In the Interacting Multiple Models approach (IMM), it is assumed that the system obeys one of a finite number of models. A Bayesian framework is used here starting with the prior probabilities of each model being correct. The model, assumed to be in effect throughout the process, is one of r possible models (the system is in one of r modes) as $\{\Phi_j\}_{j=1}^{r}$.

Then the prior probability that Φ_j is correct (the system is in mode j) is shown in Eq. (22).

$$P\{\Phi_j | Z^0\} = \mu_j(0) \tag{22}$$

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

1. Calculation of the mixing probabilities $(i,j = 1, \ldots, r)$. The probability that the 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. (23).
$$\mu_{i|j}(k-1|k-1) = \frac{1}{\bar{c}_j} p_{ij} \mu_i(k-1); \quad \bar{c}_j = \sum_{i=1}^{r} p_{ij} \mu_i(k-1) \tag{23}$$
It is assumed that the model jump process is a Markov process (Markov chain) with the known model transition probability p_{ij}
$$p_{ij} = P\{\Phi(k) = \Phi_j | \Phi(k-1) = \Phi_i\} \tag{24}$$
here μ_i is estimated later from Eq. (29).
2. Mixing $(j = 1, \ldots, r)$. Starting with the previous estimate $\hat{X}^i(k-1|k-1)$ one computes the mixed initial condition for the filter matched to $\Phi_j(k)$ as
$$\hat{X}^{0j}(k-1|k-1) = \sum_{i=1}^{r} \hat{X}^i(k-1|k-1) \mu_{i|j}(k-1|k-1) \tag{25}$$
The covariance corresponding to the above is
$$\begin{aligned} P^{0j}(k-1|k-1) = \sum_{i=1}^{r} \mu_{i|j}(k-1|k-1) \{ & P^i(k-1|k-1) \\ & + [\hat{X}^i(k-1|k-1) - \hat{X}^{0i}(k-1|k-1)] \\ & \cdot [\hat{X}^i(k-1|k-1) - \hat{X}^{0j}(k-1|k-1)]' \} \end{aligned} \tag{26}$$
3. Model-matched filtering $(j = 1, \ldots, r)$. The estimate Eq. (25) and covariance Eq. (26) are used as the inputs to the filter matched to $\Phi_j(k)$, which uses the measurement $z(k)$ to yield $\hat{X}^j(k|k)$ and $P^j(k|k)$. The likelihood function corresponding to the r filters
$$\Lambda_j(k) = p[z(k) | \Phi_j(k), Z^{k-1}] \tag{27}$$
is computed using the mixed initial condition Eq. (25) and associated covariance Eq. (26) as:
$$\Lambda_j(k) = p[z(k) | \Phi_j(k), \ \hat{X}^{0j}(k-1|k-1), \ P^{0j}(k-1|k-1)] \tag{28}$$
4. Model probability update $(j = 1, \ldots, r)$. It is done as follows:
$$\mu_j(k) = \tfrac{1}{c} \Lambda_j(k) \bar{c}_j, \quad c = \sum_{j=1}^{r} \Lambda_j(k) \bar{c}_j \tag{29}$$
where $\bar{c}_j$ is the expression from Eq. (23).
5. Estimate and covariance combination. Combination of the model-conditioned estimates and covariances is done according to the mixture equations
$$\hat{X}(k|k) = \sum_{j=1}^{r} \hat{X}^j(k|k) \mu_j(k) \tag{30}$$
$$\begin{aligned} P(k|k) = \sum_{j=1}^{r} \mu_j(k) \{ & P^j(k|k) + [\hat{X}^j(k|k) - \hat{X}(k|k)] \\ & \times [\hat{X}^j(k|k) - \hat{X}(k|k)]' \} \end{aligned} \tag{31}$$
Note that the combination is only for output purposes-it is not a part of the algorithm recursions.

5.3. Interacting multiple models for extended Kalman filter estimation

Recall the basic model equations for the linear Extended Kalman Filter Eq. (32).

$$\begin{aligned} X(k) &= \Phi(k-1) * X(k-1) + v(k-1) \\ Z(k) &= C(k) * X(k) + w(k) \end{aligned} \tag{32}$$

where Φ is the state transition matrix, X is the state vector, v is the process noise, Z is the measurement value, C is the Jacobian matrix of the measurement matrix which is used for the Extended Kalman Filter updating, and w is the measurement noise. The actual state estimation and prediction equations are given as Eq. (33):

$$\begin{aligned} &X(k|k-1) = \Phi(k-1) * X(k-1|k-1) \\ &X(k|k) = X(k|k-1) + G(k) * \text{residue}(k) \\ &\text{residue}(k) = Z(k) - C(k) * X(k|k-1) \\ &P(k|k-1) = \Phi(k-1) * P(k-1|k-1) * \Phi(k-1) + Q(k-1) \\ &S(k) = C(k)P(k|k-1)C(k)^{\mathrm{T}} + R(k) \end{aligned} \tag{33}$$

where residue(k) is the measurement residue, a Gaussian random variable with the zero mean and covariance $S(k)$, and $R(k)$ is the covariance of the measurement noise w.

The equation for the filter gain, G, and the covariance matrix, P, of the state prediction are then as Eq. (34):

$$\begin{aligned} G(k) &= P(k|k-1) * C(k)^{\mathrm{T}} * (C(k) * P(k|k-1) * C(k)^{\mathrm{T}} + R(k))^{-1} \\ P(k|k) &= (I - G(k) * C(k)) * P(k|k-1) \end{aligned} \tag{34}$$

where $Q(k)$ is the covariance of the process noise v, $C(k)$ is the measurement matrix from Eq. (32) and $\Phi(k)$ is the state transition matrix from Eq. (32).

According to the Interacting Multiple Models (IMM) approach, several simple linear dynamic models Φ introduced in the previous sections are used as the different state transition matrixes (Eq. (15)) for Extended Kalman Filtering. In the IMM algorithm, the likelihood of each mode-conditioned estimates $\Lambda_j(k)$ as shown in Eq. (28) is generated with the Extended Kalman Filter measurement residue $\text{residue}_j(k)$ (Eq. (33)) as Eq. (36).

$$\Lambda_j(k) = N[\text{residue}_j(k); 0, S_j(k)] \tag{35}$$

Now with the different model's Extended Kalman Filter state estimate X^j together with the covariance estimate P^j and based on the above recursive IMM algorithm, the proposed data fusion based visual target tracking system can be achieved.

Based on the Interacting Multiple Models (IMM) approach, the above four simple linear dynamic models are fused for Extended Kalman Filter. In the IMM algorithm, the likelihood of each mode-conditioned estimates $\Lambda_j(k)$ is generated with the Extended Kalman Filter measurement residue $residure_j(k)$ as Eq. (36).

$$\Lambda_j(k) = N[\text{residue}_j(k); 0, S_j(k)] \tag{36}$$

where $\text{residue}_j(k)$ is the measurement residue, a Gaussian random variable with the zero mean and covariance $S_j(k)$. With the different model's Extended Kalman Filter state estimate X^j together with the covariance estimate P^j and based on the recursive IMM algorithm [1], the proposed data fusion based visual target tracking system can be achieved.

6. Experiment results

To evaluate the performance and adjust the parameters of the proposed method, the tracking system has been tested under both the indoor and outdoor situations. In the experiment a moving platform is designed with the motion sensors and CCD cameras. Motion sensors (tachometers) measure the platform's wheels speed, which estimate the cameras' velocity based on the vehicle's kinematic model. Two color CCD cameras (the focus length $f = 5.4$ mm) are connected with the Hitachi IP5005 image processing card for the image capturing and processing (the image size is 256×220, and the capturing rate is $15f/s$).

The processing of the video images takes a lot of computational time. The images are captured in a 15 Hz rate, but the optical flow vectors estimation and stereo disparity calculation can not be calculated at the same time in such a high rate using the current computer hardware. This makes the target tracking system difficult to work in real-time, thus the following experiments are done off-line. However with the rapid development of the hardware especially the image processing cards available, in the future it is sure to make this tracking system work in real time.

The data from different sensors are captured and processed in the fusion center. Later the target's 3D world coordinate position and velocity are estimated and tracked. In actual autonomous vehicle experiments, these dynamic information will be used for the navigation commands of the vehicle. Here the estimated target's 3D position in the world coordinate is just projected into the 2D image to show the tracking performance. If the target region can be well identified and tracked through the image sequence, it means that the proposed tracking system's performance is accurate.

First, some indoor experimental results are presented to show the step-by-step image processing and Extended Kalman Filtering performance. Next, outdoor experimental results are presented to demonstrate the tracking system's performance under outdoor different situations.

As shown in Fig. 5, the platform carrying the cameras is moving towards the moving target, and as a result the size of the object appearing in the scene increases from left to right. As explained earlier, the tracking system, proposed in this paper, can handle the changes in visual depth as well as the varying background conditions.

In this experiment, the object of interest is always in the scene. If the object is not in the image scene, for example when the motion of the camera and object is kept predom-

Fig. 5. Indoor moving target with moving background. A blue color object is moving in the image sequence. It can be seen that the background is changing frame by frame (For interpretation of color mentioned in this figure the reader is referred to the web version of the article).

inantly horizontal, the tracking algorithm will decide that the target exits and will stop the image processing.

In these experiments stereo images are captured as the vision sensor data. The image processing is done only with one of the stereo image sequences and the 2D visual features are estimated in one image sequence. The stereo images are only used later to estimate the 3D disparity depth. In such a way, the double image processing is avoided, which can reduce the total computational cost.

Fig. 6 shows the optical flow vectors fields estimated. The computation of optical flow vectors is based on the algorithm from the book [14]. The optical flow vectors field is 100% dense, which reflects the movements of the target and background. In this paper before the optical flow estimation, some Gaussian filters are used to smooth the raw images captured from the cameras. With the Gaussian filtering, first, the optical flow estimation accuracy can be improved. Secondly, the smoothing can reduce the texture effects of the object and make the following image segmentation and clustering more accurate.

The K-means clustering algorithm is implemented to segment the optical flow vectors fields into several regions as shown in Fig. 7. The clustering algorithm is sensitive to slight movements in the images, such as the movements due to wind. Therefore, it is necessary to eliminate these vectors by thresholding. Fig. 7a shows the movements after thresholding. Here the clusters with smaller areas are removed with the use of a threshold. In this way only the significant movements in the image sequence are extracted. Fig. 7b show the K-means clustering results. Different colors represent different motion regions. As there are five colors in these images, the cluster number is 5. From the results, the region of the moving object has the same color within its body, which reflects that for a single image the object is clustered into one image cluster as a whole and the image segmentation result is satisfactory.

After the image segmentation and clustering, the major motion regions in the whole optical flow vectors field are clustered into several clusters. Next, template matching is carried out between the template and the clustered motion vectors field. A template from the segmented optical flow vectors field is shown in Fig. 8a. It contains only the optical flow vectors of the ROI. The correlated values of the template are shown graphically in Fig. 8b. In the experiment, the template is updated using the previous estimated ROI. This "naive template update" [22] process is: using the estimated 3D position of the target at the time k, the flow vectors region in the 2D flow vectors field (i.e. ROI) is then extracted by using a pinhole camera projection model. The flow vectors region is used as the template for template matching at the time $k+1$.

In this paper the stereo correspondence matching is done only by matching the target region from one image onto the other to estimate the disparity value, which can reduce the computational time by ignoring the whole image matching. Fig. 9b shows the estimated visual depth vs. the frame number. It can be seen that the distance between the target and cameras are decreasing as the platform moves towards the target. The estimated depth results are compared with the measured distances (the ground true data). The estimation results are satisfactory.

While capturing the images, two tachometers measure the dynamic parameters of the moving platform. Fig. 10a shows the moving platform's velocity in the 2D (X,Z) plain. The platform's velocity is estimated using the two tachometers sensors' readings based on the vehicle's kinematic model. Fig. 10b presents the estimated trajectory of the moving platform compared with the ground

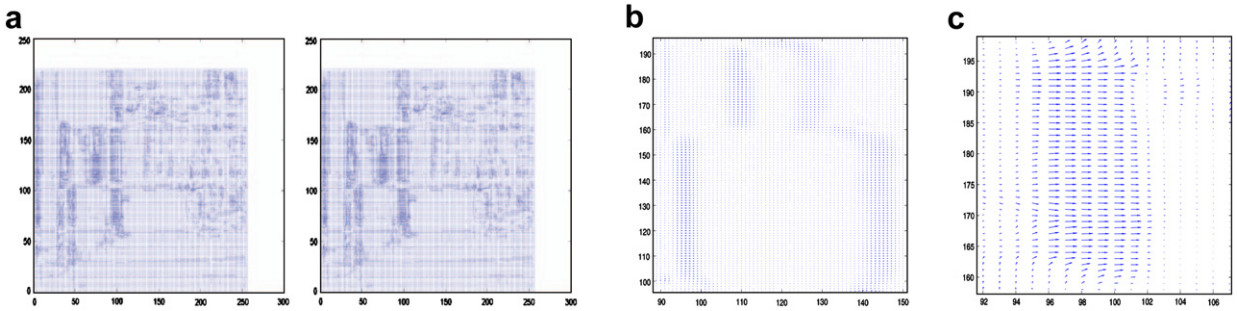

Fig. 6. Optical flow vectors fields estimated from the image sequence with the Horn and Schunck's method. (b and c) The enlarged figures from some of the optical flow vectors fields for clarity. Each vector reflects the image pixel's velocity.

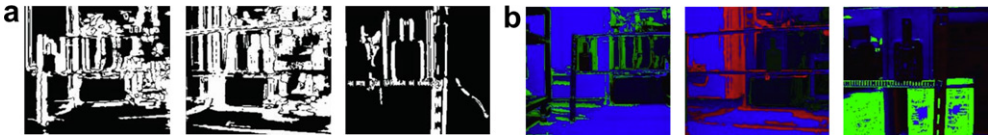

Fig. 7. Motion fields segmentation and clustering results. (a) Shows the motion regions after removing the noise. (b) Shows the K-means clustering results. In these figures, different colors represent different image clusters.

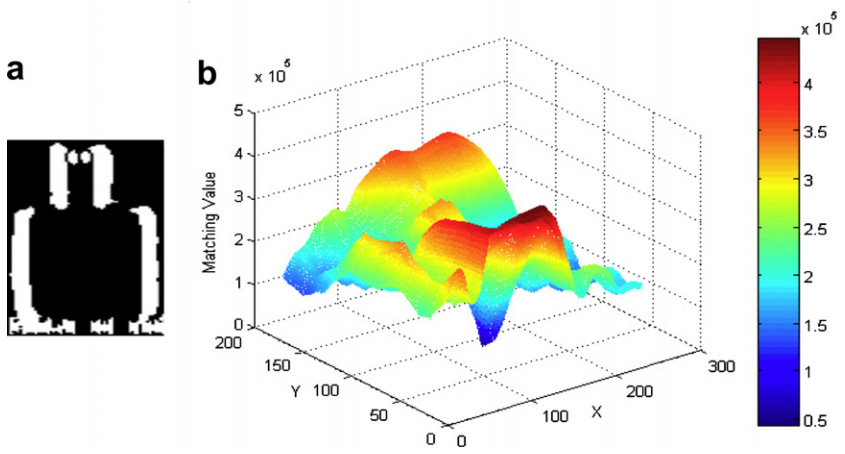

Fig. 8. The matching template and the template matching values. The matching is estimated with the SSD. The smallest value means the best template matched region.

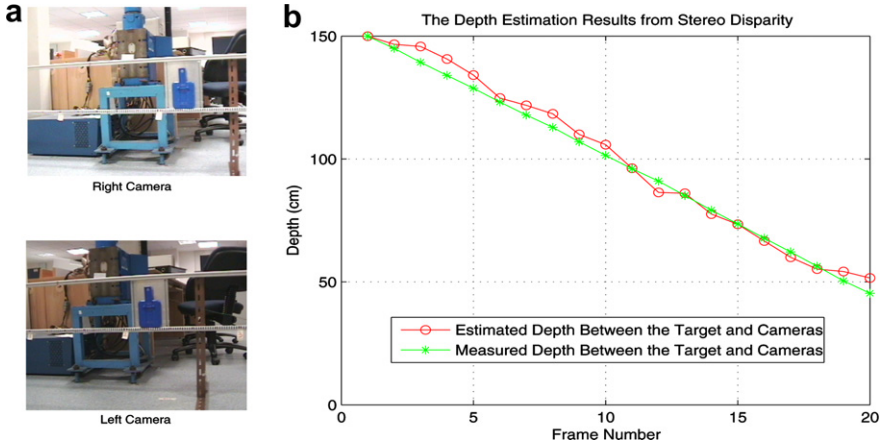

Fig. 9. One pair of stereo images are shown in (a) and the depth estimation result is shown in (b).

truth trajectory in a 2D plain, which also agrees that the cameras are moving towards the target. The difference between the estimated trajectory and the ground truth data is not large (only centimeters compared with the distances in meters and less then 10%). This reflects that the tachometer motion sensor is suitable for the experimental testing.

The moving cameras' velocity is fused with the optical flow vectors and visual disparities for the target's 3D position and velocity estimation (Eq. (12) and Eq. (14)). After the estimation, the Interacting Multiple linear dynamic Models (IMM) algorithm is carried out for the Extended Kalman Filter target tracking.

In this paper for the implementation of IMM based Extended Kalman Filtering, the Markov chain transition matrixes between different linear dynamic models are taken from [1]. The matrix for the second order models (Drifting Points Model, Constant Velocity Model and Periodic Motion Model) is:

$$[p_{ij}] = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix} \tag{37}$$

The final results are not very sensitive to these values (e.g., p_{11} can be set between *0.8* and *0.98*). For the third order model (Constant Acceleration Model), the Markov chain transition matrix can be

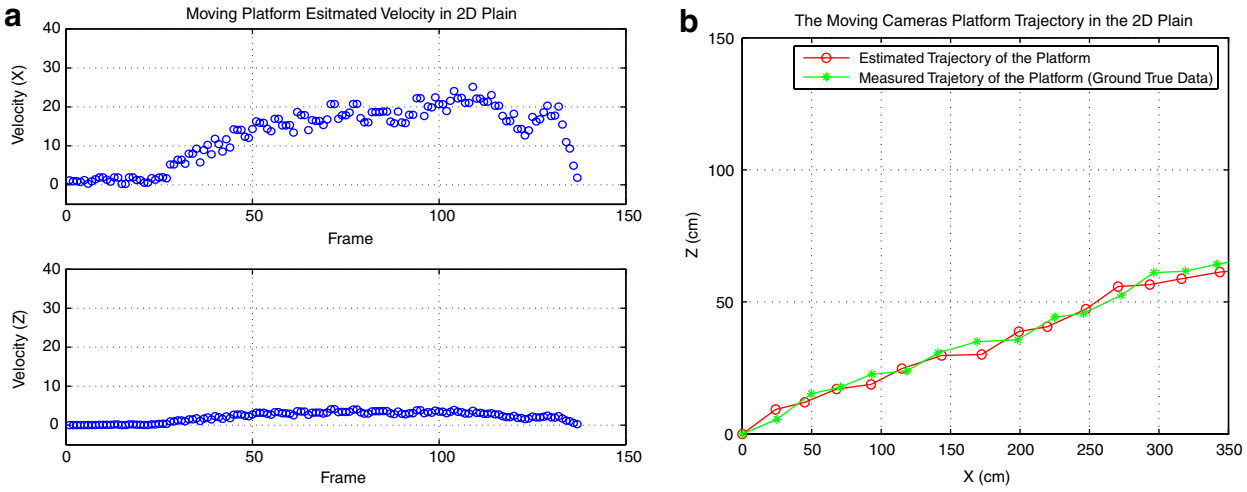

Fig. 10. Motion sensors data and the cameras platform's movement trajectory in the 2D plain. (a) Shows the estimated two wheels velocities. The x axis is the frame number. The y axis is the velocity (cm/s). Based on the estimated velocity of the moving platform, the movement trajectory can be plotted as shown in (b). The x and y axes represent the vehicle's positions in the 2D plain.

$$[p_{ij}] = \begin{bmatrix} 0.95 & 0.05 & 0 \\ 0.33 & 0.34 & 0.33 \\ 0 & 0.05 & 0.95 \end{bmatrix} \tag{38}$$

In the experimental testing, the vector $(\vec{P}, \vec{V})^{\mathrm{T}}$ in Eq. (15) is rewritten as $(p_x,v_x,p_y,v_y,p_z,v_z)^T$ for the easy implementation. $(p_x,p_y,p_z)^T$ represents the target's 3D position and $(v_x,v_y,v_z)^T$ represents the target's 3D velocity. The state estimation noise $v(k,k-1)$ in Eq. (15) can be considered as zero mean, white with covariance, which is obtained based on the noise identification procedures from [1]. The noise in each coordinate axis is assumed independent, then it has:

$$Q = E[v(k,k-1)v(k,k-1)'] = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & Q_1 & 0 \\ 0 & 0 & Q_1 \end{bmatrix}, \quad Q_1 = \begin{bmatrix} \Delta t^3/3 & \Delta t^2/2 \\ \Delta t^2/2 & \Delta t \end{bmatrix} \tilde{q} \tag{39}$$

where Δt is the sampling interval. Q_1 is the process noise covariance matrix component for each direction of the target's 3D movements. The choice of the power spectral density $\tilde{q}$ of the process noise is selected based on the previous experimental testings. Here the process noise variance matrix for Extended Kalman Filter is taken as

$$Q_1 = \begin{pmatrix} 1\mathrm{e}-3 & 0 \\ 0 & 1\mathrm{e}-3 \end{pmatrix}$$

.

The measurement noise $w(k)$ in Eq. (17) is independent of v, and zero mean, white with covariance

$$E[w(k)w(k)^{\mathrm{T}}] = R \tag{40}$$

where R is the measurement noise covariance, which is a 7×7 matrix. R is selected based on the different sensors' characteristics and also the optical flow vectors estimation, stereo disparity calculation and all the other image processing's accuracies. In this paper, the disparity and displacement fields are computed for each time step and superimposed with the additive Gaussian white noise with a variance 1/12.

The estimation starts with the initial covariance $P(1|1)$ (a 6×6 matrix) of the state at $k = 1$ (based on the 2-point differentiation)

$$P(1|1) = \begin{bmatrix} P_{11} & 0 & 0 \\ 0 & P_{11} & 0 \\ 0 & 0 & P_{11} \end{bmatrix}, \quad P_{11} = \begin{bmatrix} r & r/\Delta t \\ r/\Delta t & 2r/\Delta t^2 \end{bmatrix} \tag{41}$$

and proceeds to $k = 2$. P_{11} is the covariance matrix component for each direction of the target's 3D movements. r can be normally set based on the measurement noise $w(k)$ and its covariance R.

In order to show the performance of the tracking algorithm, a white rectangle is plotted to represent the target's regions in the image. The rectangle is obtained by projecting the 3D position of the target in the world coordinate onto the 2D image using a pinhole camera model (Eq. (16)). Here the back-projecting 3D estimates into the original images may not be very accurate, because the systematic errors in the vehicle localization (such as the wheels slippage) are cancelled out in the proposed algorithm. However, in order to show the tracking results in a reasonable and clear way for the off-line testing, the back-projecting is still a better choice despite of some possible errors.

Fig. 11 shows the 3D target tracking results with the IMM algorithm. From Fig. 11a–e, the rectangle can accurately fit the target region, which shows that the tracking system performs well. Fig. 11f shows the projected horizontal position of the target in the image along the x-axis of the plot and the estimated height of the target in the image along the y-axis. As the cameras are moving closer to the target, the target appears bigger and bigger in the images from left to right. In the indoor laboratory experiments,

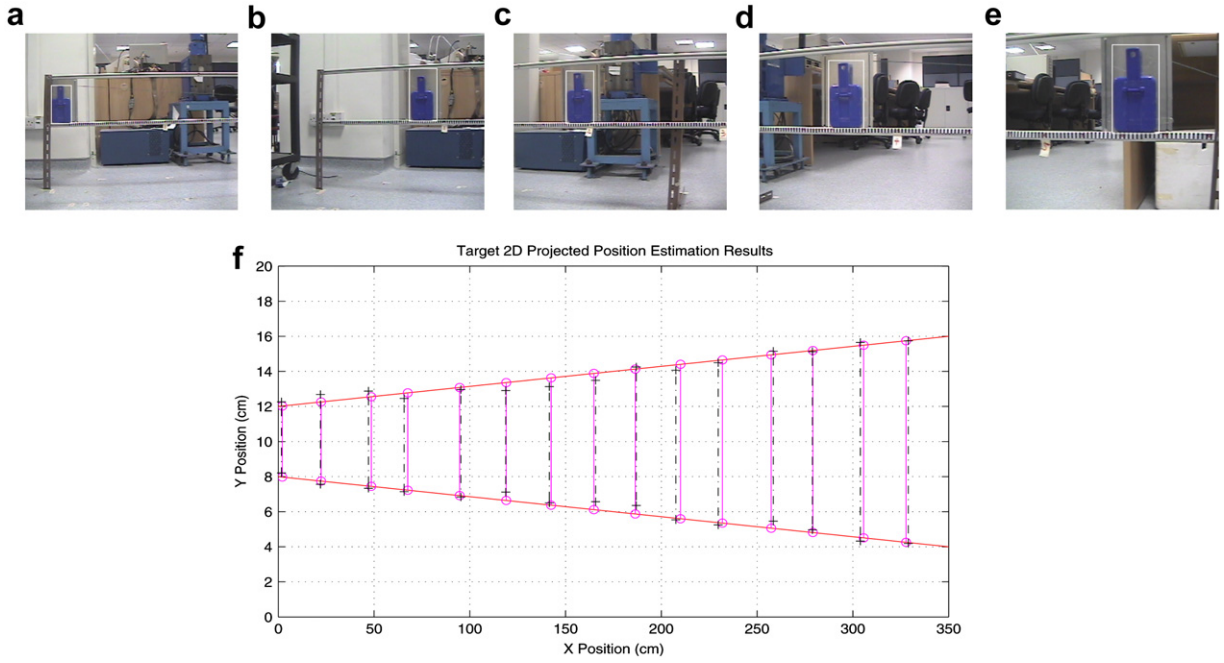

Fig. 11. Target's 3D tracking results with the Interacting multiple linear dynamic models algorithm. The target is correctly tracked (a–e). The size of the template is changing with the distance changes. In (f) the measured target position (ground truth data) and estimated target position along the trail are compared. The x and y axes are the 2D positions along the trail (cm).

the target's movements along the trail are marked to obtain the ground truth measurements (the "$o-$" lines). The "$+-$" lines are the target's estimated positions from the proposed algorithm. There is no much difference between them (only several centimeters less than 10% of the whole position). When the estimated 3D positions are projected into the 2D images as shown in Fig. 11a, the target is precisely localized and these differences can be just neglected. It is clear from the figures that the proposed scheme tracks the 3D position of the target accurately in the world coordinate.

The off-line results of the target's first 20 states are shown to demonstrate the estimation performance with the IMM algorithm. From Fig. 12 the target has quite different dynamic motion properties. In the target velocity estimation figure (Fig. 12b), the target's velocity is hardly following any simple linear dynamic model, which is

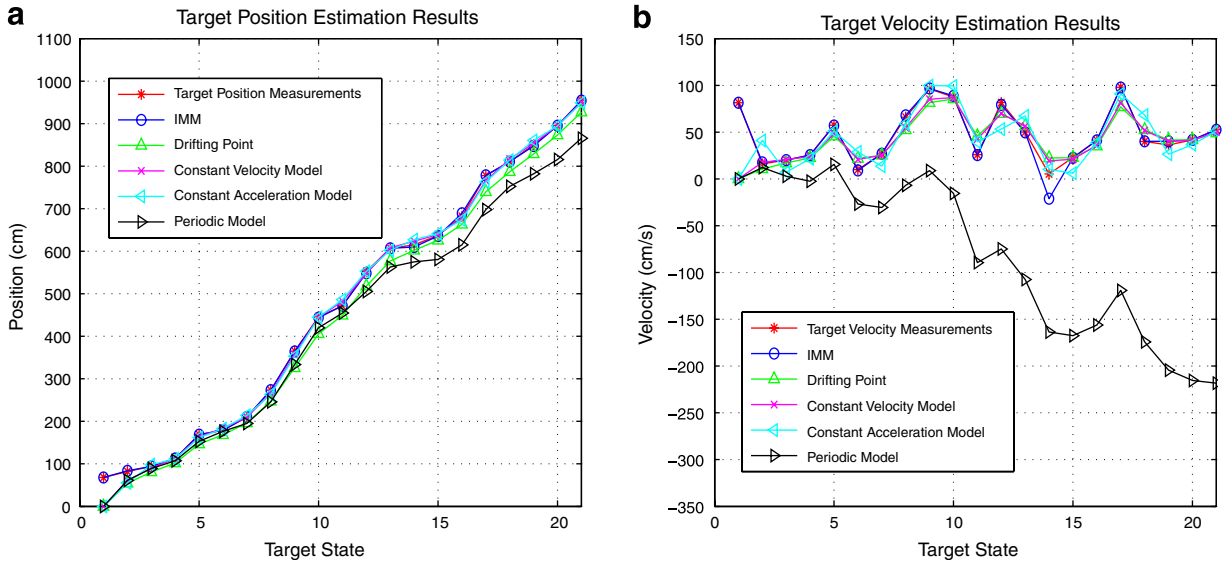

Fig. 12. Target's position (a) and velocity (b) estimation results from different models. The estimates along the x dimension of the target's 3D world coordinate position and velocity are shown here. The estimation results with the IMM algorithm are relatively more accurate than those from other models. In these figures the target position measurements are the ground truth values.

changing continuously. In these figures the estimates from the four simple linear dynamic models and IMM are compared together with the measurements (the ground truth data). It is clear that the IMM estimates are the best among all the other linear dynamic models. Some models like the periodic model even diverge very fast.

The model estimation performance can be evaluated by the root-mean-square error (RMS).

$$\mathrm{RMS}_m = \sqrt{\frac{1}{N}\sum_{i=0}^{N}[\hat{F}_m(z_i) - F_m(z_i)]^2} \tag{42}$$

where the subscript m indicates the mth profile. When a group of profiles are considered, the mean value can be computed,

$$\mathrm{RMS} = \frac{1}{M}\sum_{m=1}^{M} RMS_m \tag{43}$$

where M is the total number of profiles. If the model is perfect, $(\hat{F}_m(z_i) - F_m(z_i))$, RMS_m should be zero. Usually, the model is not perfect. The smaller the RME is, the better the performance of the model is.

The RMS errors comparison of the estimation results among different models can be seen in Fig. 13. The RMS errors are based on the multiple repetitions of experiments (10–20 runs), and using the RMS errors is a better way to demonstrate the tracking performance comparisons. Compared with other models, the RMS errors from the IMM estimation are relatively smaller with the smallest peak error.

Fig. 14 shows the estimation results when the target moves with a direction turn. In such a situation, other linear dynamic models poorly track the target since the velocity changes too much. There are large estimation lag errors at the motion direction turn point. The periodic model even diverges very fast. However the IMM algorithm still gives quite accurate position estimates. Without the maneuver detection decision, the IMM algorithm is able to accurately track the target motion with the direction turn. See Fig. 15

6.1. Outdoor experiments

Some outdoor experiments are done with the moving platform to test the tracking system with the IMM algorithm. The proposed target tracking system can be used to track any object moving in front of the vehicle. In all the outdoor experiments the moving objects are people (real moving objects). Tracking moving people is easier for the experimental testing. In fact the moving people's shape is irregular, which makes the tracking more difficult than tracking regular shape objects like cars. The moving persons can also be considered as the textured objects which have only the uniform motion properties but quite different color features within their bodies. If the moving person can be accurately tracked, then the tracking system is robust to track any other normal moving object.

In the outdoor experiments, the target's position and velocity are estimated in the 3D world coordinate. In order to show the tracking results, the same method used in the indoor situation is applied here. The target's 3D position in the world coordinate is projected into the 2D image through the pinhole camera projection model.

In Fig. 16 if the dynamic model of the tracked object is not considered, there is a chance that the algorithm tracks the wrong object. From Fig. 16, the lost target may result if the algorithm uses only the motion vectors field for the target identification. At the occlusion region, the two persons

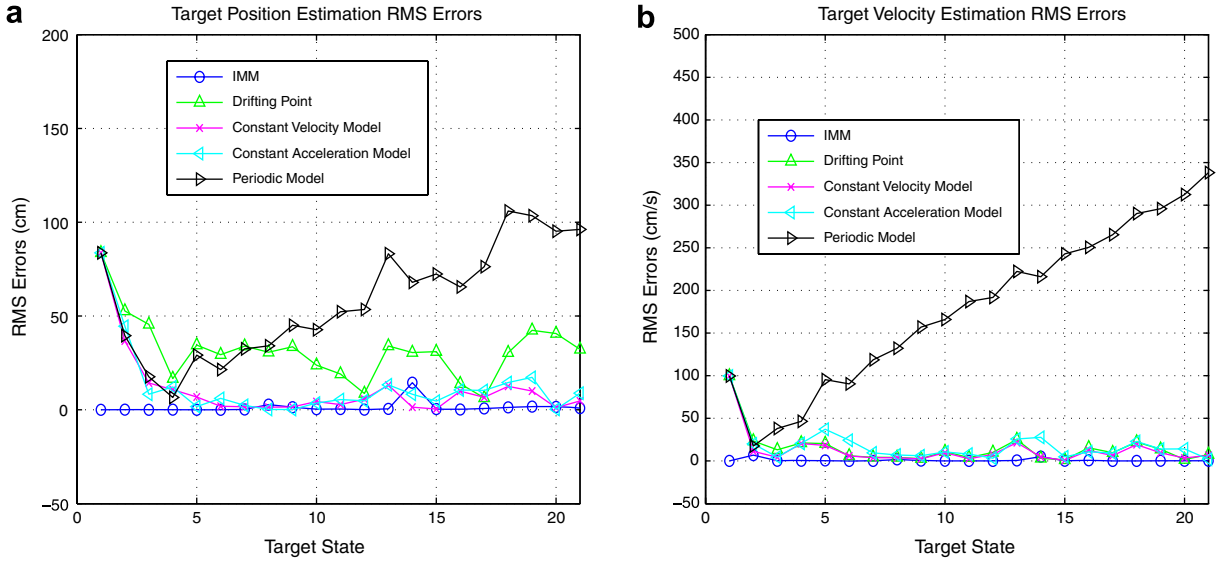

Fig. 13. Multiple repetitions of the experiments are performed and the RMS errors are compared. The estimates along the x dimension of the target's 3D world coordinate position (a) and velocity (b) are shown here. Here the RMS errors from the estimation results with the IMM algorithm are relatively smaller.

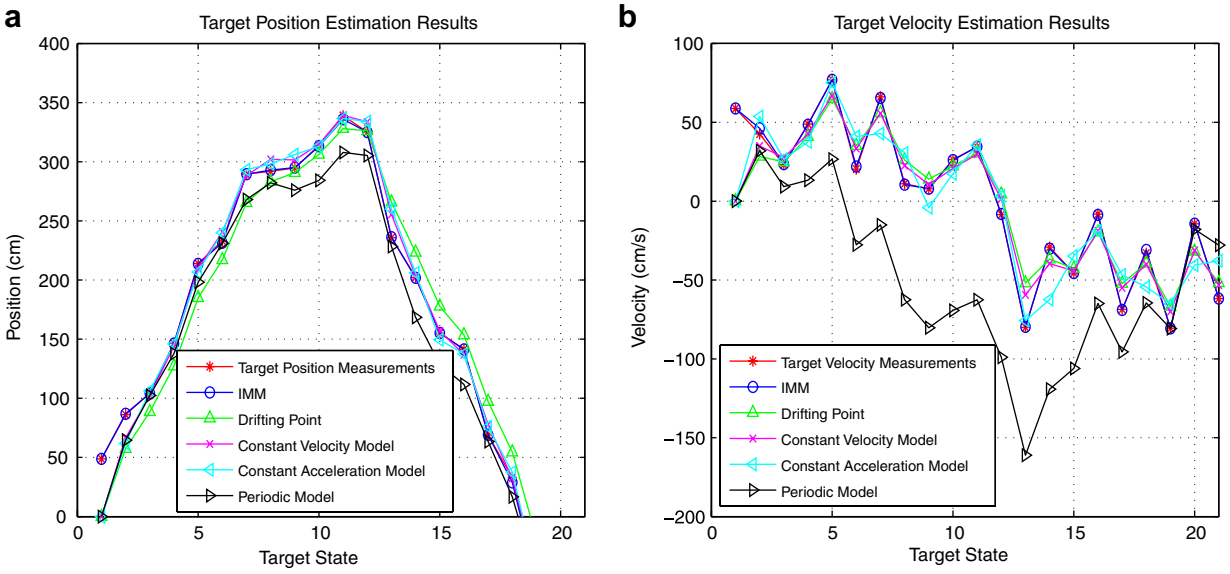

Fig. 14. Target dynamic information estimation results from different models with the target moving direction turn. The estimates along the x dimension of the target's 3D world coordinate position (a) and velocity (b) are shown here. In such a situation, the estimation results with the IMM algorithm are still quite accurate. In these figures the target position measurements are the ground truth values.

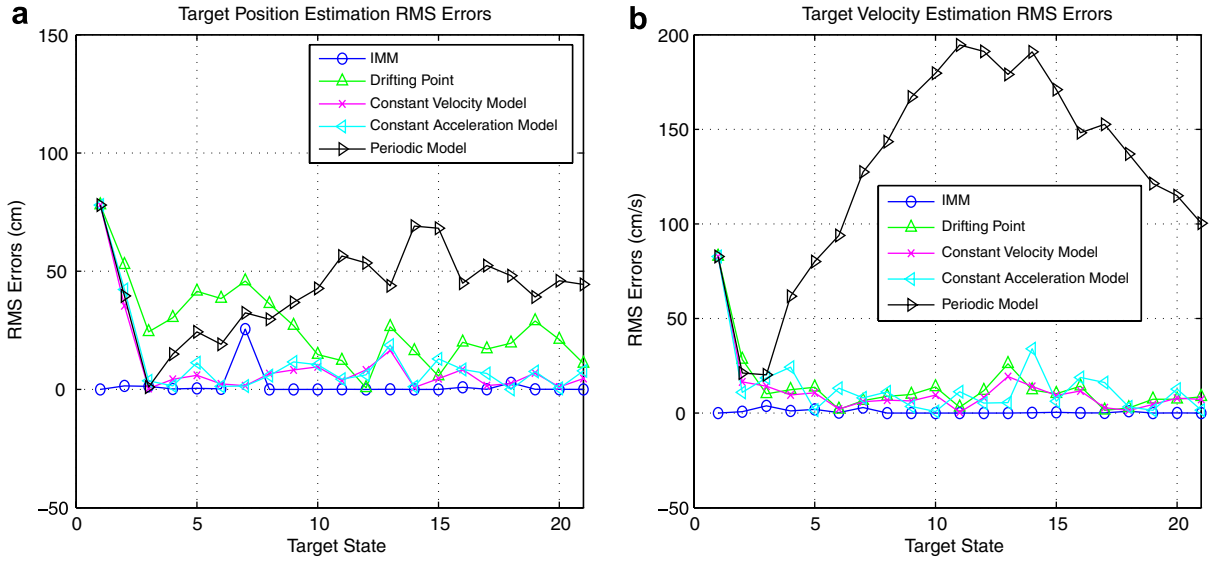

Fig. 15. Multiple repetitions of the experiments are performed and the RMS errors are compared. The estimates along the x dimension of the target's 3D world coordinate position (a) and velocity (b) are shown here. Here the RMS errors from the estimation results with the IMM algorithm are relatively smaller.

Fig. 16. First, the person on the right is tracked. When these two people meet, due to occlusion the tracker will lose the target (a–e). The person moving from left to right is then tracked. (f) is the image segmentation result with occlusion. The tracking system can not recognize the previous tracked person when they are occluded.

have similar motion regions (Fig. 16f). The tracking algorithm can not distinguish the previous tracked one from the other.

In this paper, the above problem is solved by using a suitable dynamic model for tracking, which can estimate and predict the expected direction of the movement of the target. As explained earlier, the Interacting Multiple linear dynamic Models algorithm is used to predict the new possible position of the target in the image sequence. The predicted position of the target with its current position is used to identify the direction of its movement (Fig. 17). With the prediction, the above mentioned problem is avoided as shown in Fig. 18.

In Fig. 19 two different persons are moving together in the same direction and they have different velocities. Using the Extended Kalman Filter estimates, the tracked object's velocity and motion direction can be estimated and then only in the predicted regions, the person is identified and tracked. In such a way, the previous tracked person is still being correctly tracked.

Figs. 18, 20, 21 and 22 show the proposed algorithm's performance in different outdoor situations. First, the moving object can be seen as the textured object in the outdoor cluttered environment. The moving person with different colors within its body can be accurately segmented out and identified from the images using the fusion based K-means clustering algorithm. Second, as the cameras and object are moving towards each other, the cameras have not only the pure translational but also the 3D motions. Since the person's moving region is required to be in image scene for tracking, the target's 3D velocity and position are estimated with the cameras' rotational and z-translational motions based on the moving cameras' kinematic model (R from Eq. (10) for the rotational motion and T from Eq. (11) for the translational motion). Third, in Fig. 20 the motion of the cameras looks large while in Fig. 21 the cameras look to be moving with a slow velocity. When the moving cameras are moving close to the moving object and their motion looks large, the changes between two consecutive images are relatively larger than those with the cameras' slow motion. In this paper,

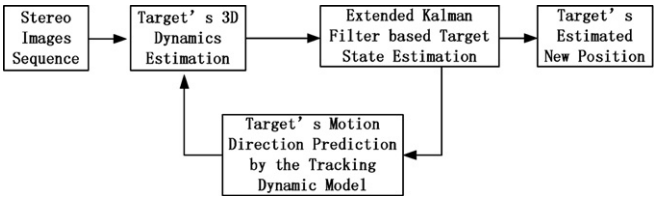

Fig. 17. Tracking with the direction prediction based on the dynamic model. The direction prediction can be used to solve the occlusion problem.

Fig. 18. Improved tracking performance with the modification to solve the two persons' occlusion problem (a–e).

Fig. 19. Two persons are moving with a moving camera in a cluttered outdoor environment (a–g). Here when the first person is fully occluded by the second person (d and e). The target is lost. Here the occlusion problem is solved with the Extended Kalman Filter prediction (Fig. 17) and the first person can still be tracked.

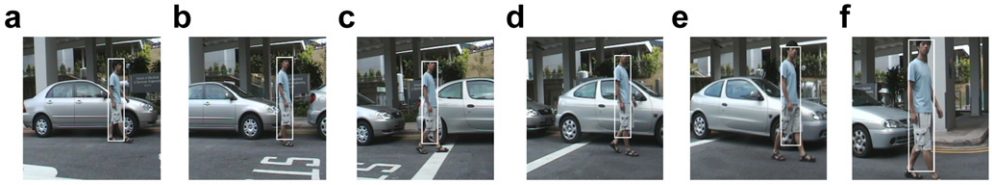

Fig. 20. Target's 3D tracking result when a person is interacting with the moving vehicle. The background is changing frame by frame (a–f). The person is moving towards the moving cameras. The person is correctly tracked with the proposed system. The tracked region's size is scaled due to the distance change between the person and the cameras.

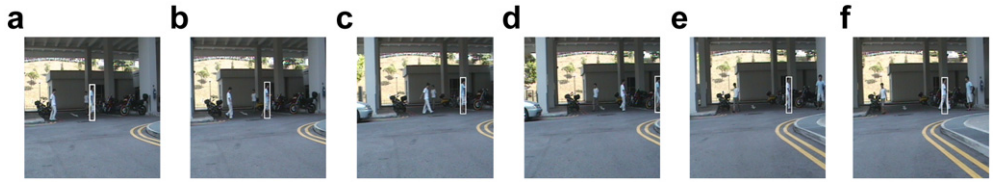

Fig. 21. Target's 3D tracking result when 3 persons are moving in the cluttered outdoor environment. The person is correctly tracked (a–f). However when the tracked person leaves the image scene, the target is lost. Since the proposed method is based on the motion feature tracking, the algorithm tracks the person with the most similar motion instead (e and f).

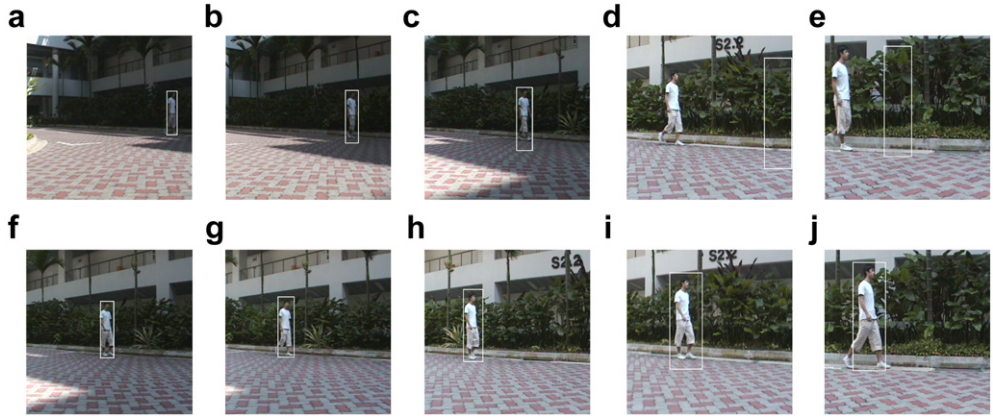

Fig. 22. The target's 3D tracking results when the person is moving in the cluttered outdoor environment. The person is moving from the dark side to the bright side (a–e and f–j). In such a situation the visual features of the person are changing. The person looks much brighter frame by frame. If the tracking is done without the motion based algorithm developed in this paper, the tracking results are not good and the target is lost (d and e). Here the performance is improved with the proposed motion proposed tracking algorithm as shown in (f, g, h, i and j). The person is accurately tracked with its changing visual features.

Fig. 23. The target tracking results when several different people enter the image scene (a–j). The cameras are moving in a cluttered environment. Here the tracker is initialized when one person is entering the image scene and the tracker is terminated when this person is leaving the image scene.

all the data processing are done off-line, so even with a large amount of data to be processed the tracking system can well deal with the large motion of the cameras. Fourth, in Fig. 22, the environmental lighting condition changes greatly. The target's visual features change a lot through the image sequence. The proposed tracking system is designed based on the object's motion properties, which is independent of other visual features like color. From the experimental results, in all these four cases the target tracking is accurate.

Several other image sequences (Figs. 23 and 24) are shown to demonstrate the experimental performance of the target tracker's identification, initialization, termination and multiple objects tracking. These sequences are with a number of moving people to proof the efficiency of the proposed method in a general case.

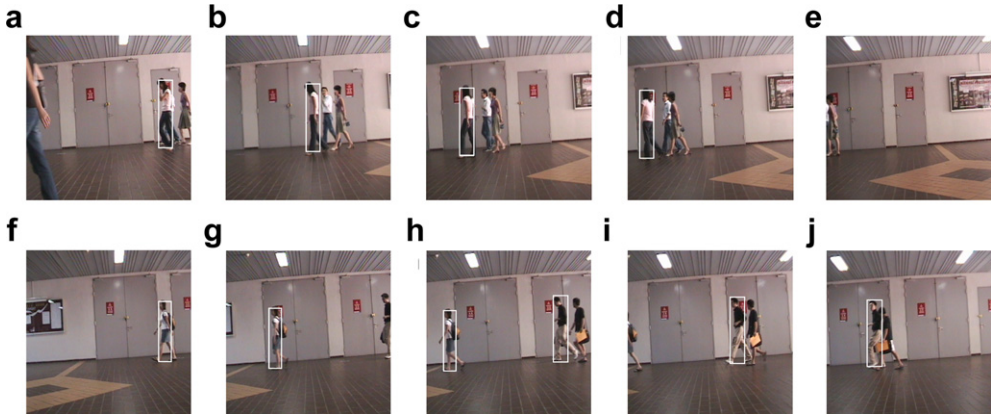

Fig. 24. The target tracking results when several people are moving in the same environment (a–j). The cameras are also moving. As same as Fig. 23, the tracker can be automatically initialized and terminated for each person. The target tracking can be accurately done when several people move in the same environment. Even if two persons have quite similar visual properties (h, i and j), the tracking can still be accurately done.

If a new person is entering the image scene, the tracker can be automatically identified and initialized for this person. When the person is leaving the image scene, the tracker can also be automatically terminated. The target tracking results are good as shown in Fig. 23.

In Figs. 23 and 24, the target tracking algorithm is tested when several different persons enter and leave the image scene. In such a situation, the target tracking system can track different people at the same time in the same environment. If several persons are moving in the environment and a new person is entering the image scene, the target tracking is set so that the appearance of the pervious tracked person is memorized as the image template, and then a new tracker different from the previous tracked one is initialized for the newly entered motion region in the images. From the experimental results the system can accurately initialize or terminate a tracker for each person. This reflects that the proposed target tracking algorithm has the ability to track different objects with similar visual features at the same time.

7. Conclusion

In this paper, a novel data fusion based target-tracking scheme is proposed. Optical flow vectors, stereo disparity and cameras' motion parameters are combined together through the image processing algorithm to obtain the moving target's 3D dynamic features. Several simple and basic linear dynamic models [13] are combined using the IMM algorithm to provide an accurate approximate model for the target's dynamics. Through extensive experiments, the proposed tracking scheme demonstrates the advantages of combining different data fusion strategies to get reliable target tracking results under different situations.

The main contribution of this paper is not any individual technique used or developed but the whole new tracking system formulated. In conclusion the proposed system has the following advantages over the previous works such as [7,10,17,19, 20,26,33], 1. it is only based on the vehicle's onboard sensors to detect and track the object; 2. the object is detected mainly based on its motion properties independent of its specific visual properties, which thus can deal with the situations of the unexpected or textured objects tracking or tracking different objects at the same environment. 3. the kinematic model of the vehicle is included in the tracking system, which can deal with the cameras' complex motions. 4. fusion of different tracking models using IMM make target tracking more reliable. In conclusion it can be stated that the proposed tracking scheme is suitable for the target identification and tracking by autonomous vehicles.

The experiments in this paper are done off-line. Currently, a moving platform is designed with all the sensors and embedded pc available to simulate an autonomous vehicle. Building a real vehicle with fully autonomous functions is really difficult. But from the off-line outdoor experiments, the results are good under different situations and the theories of this paper are also sound. So after new hardwares are purchased and a whole new autonomous vehicle is built, the proposed algorithms can be successfully implemented to get the satisfactory real-time performance. Another reason is that the real-time part of the experiments is only that after the target has been tracked, the target's dynamic information is just sent to the vehicle's motion controller and it will generate commands to control the vehicle's movement to track or follow the target. The vehicle's kinematic control is really out of the scope of this paper. This paper is to design a computer vision and sensor data fusion based target tracking system, and for this part it has successfully fulfilled the task.

The major computation complexity of the proposed scheme is from the image processing parts (optical flow vectors and stereo image processing). The 3D velocity estimation, the Extended Kalman Filter and the IMM algorithm do not add any more computational burden. Thus in order to simplify the whole tracking system, the image processing parts should be faster. However with

the rapid development of the hardware especially the image processing cards (FPGA) available, in the future it is quite reasonable to make this tracking system perform well in real-time.

As for future work, each part of the system can be modified to make them work in cooperation under difficult situations to get reliable results. These can be that: A better image segmentation algorithm and template initializing and updating algorithm can be developed to make the image 2D features extractions more accurate. In order to obtain the best Extended Kalman Filter estimation results, the IMM algorithm has to be properly improved to meet the following requirements: 1. design of the target motion models for all the modes of movements considering both the quality and complexity of the model; 2. selection of the model parameters, such as the noise level. 3. determination of the parameters of the underlying Markov chain, that is, the transition probabilities. Further experiments can be done on autonomous vehicles to conduct the real-time target tracking tasks.

References

[1] Y. Bar-Shalom, X.R. Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation, Wiley, New York, USA, 2001.

[2] J.L. Barron, D.J. Fleet, S.S. Beauchemin, Performance of optical flow techniques, International Journal of Computer Vision 12 (1) (1994) 43–77.

[3] C. Bregler. Learning and recognizing human dynamics in video sequences. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, June 1997, pp. 568–574.

[4] T. Brox, A. Bruhn, N. Papenberg, J. Weickert. High accuracy optical flow estimation based on a theory for warping. In Proceedings of 8th European Conference on Computer Vision, volume 3024 of Lecture Notes in Computer Science, Prague, Czech Republic, May 2004, Springer, pp. 25–36.

[5] J.Y. Chang, W.F. Hu, M.H. Cheng, B.S. Chang, Digital image translational and rotational motion stabilization using optical flow technique, IEEE Transactions on Consumer Electronics 48 (1) (2002) 108–115.

[6] D. Cremers, A variational framework for image segmentation combining motion estimation and shape regularization. In Proceedings of 2003 IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, Madison, Wisconsin, USA, June 2003, pp. 53–58.

[7] T. Dang, C. Hoffmann, C. Stiller. Fusing optical flow and stereo disparity for object tracking. In Proceedings of the 5th IEEE International Conference On Intelligent Transportation Systems, Singapore, September 2002, pp. 112–117.

[8] G.N. DeSouza, A.C. Kak, Vision for mobile robot navigation: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2) (2002) 237–267.

[9] J.S. Evans, R.J. Evans, Image-enhanced multiple model tracking, Automatica 35 (11) (1999) 1769–1786.

[10] Y.J. Fang, I. Masaki, B. Horn, Depth-based target segmentation for intelligent vehicles: Fusion of radar and binocular stereo, IEEE Transactions on Intelligent Transportation Systems 3 (2002) 196–202.

[11] M.E. Farmer, R.-L. Hsu, A.K. Jain, Interacting multiple model (imm) Kalman filters for robust high speed human motion tracking. In Proceedings of the 16th International Conference on Pattern Recognition, vol. 2, pp. 20–23, Quebec City, Canada, August 2002.

[12] G. Farneback. Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. In Proceedings of the Eighth IEEE International Conference on Computer Vision, vol. I, pp. 171–177, Vancouver, Canada, July 2001.

[13] D.A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Prentice Hall, USA, 2002.

[14] B.K.P. Horn, Robot Vision, MIT Press, USA, 1986.

[15] W.M. Hu, T.N. Tan, L. Wang, S. Maybank, A survey on visual surveillance of object motion and behaviors, IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews 34 (3) (2004) 334–352.

[16] M.H. Jeong, Y. Kuno, N. Shimada, Y. Shirai. Two-hand gesture recognition using coupled switching linear model. In Proceedings of the 16th International Conference on Pattern Recognition, vol. 3, pp. 529–532, Quebec City, Canada, August 2002.

[17] Zhen Jia, A. Balasuriya, S. Challa. Motion based 3d target tracking with interacting multiple linear dynamic models. In Proceedings of the 15th British Machine Vision Conference, United Kindom, September 2004.

[18] Zhen Jia, A. Balasuriya, and S. Challa. Visual information and camera motion fusion for 3d target tracking. In Proceedings of The Eighth International Conference on Control, Automation, Robotics and Vision, pp. 2296–2301, Kunming, China, December 2004.

[19] Zhen Jia, A. Balasuriya, S. Challa. Sensor fusion based 3d target visual tracking for autonomous vehicles with imm. In Proceedings of 2005 IEEE International Conference on Robotics and Automation (ICRA 2005), Barcelona, Spain, April 2005, pp. 1841–1846.

[20] A. Kosaka, M. Meng, and A.C. Kak. Vision-guided mobile robot navigation using retroactive updating of position uncertainty. In Proceedings of 1993 IEEE International Conference on Robotics and Automation, vol. 2, Atlanta, Georgia, USA, May 1993, pp. 1–7.

[21] F. Liu, X.Y. Lin, S.Z. Li, Y. Shi. Multi-modal face tracking using Bayesian network. In Proceedings of IEEE International Workshop on Analysis and Modelling of Faces and Gestures, Nice, France, October 2003, pp. 135–142.

[22] I. Matthews, T. Ishikawaa, S. Baker, The template update problem, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (6) (2004) 810–815.

[23] K. Morioka, J.H. Lee, H. Hashimoto, Human-following mobile robot in a distributed intelligent sensor network, IEEE Transactions on Industrial Electronics 51 (1) (2004) 229–237.

[24] K. Muhlmann, D. Maier, R. Hesser, R. Manner. Calculating dense disparity maps from color stereo images, an efficient implementation. In Proceedings of the IEEE Workshop on Stereo and Multi-Baseline Vision, Kauai, Hawaii, USA, December 2001, pp. 30–36.

[25] S. Nadimi, B. Bhanu. Multistrategy fusion using mixture model for moving object detection. In Proceedings of International Conference on Multisensor Fusion and Integration for Intelligent Systems, Baden-Baden, German, August 2001, pp. 317–322.

[26] R. Okada, Y. Shirai, and J. Miura. Tracking a person with 3-d motion by integrating optical flow and depth. In Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, France, March 2000, pp. 336–341.

[27] V. Pavlovic, J.M. Rehg, T.J. Cham, and K.P. Murphy. A dynamic bayesian network approach to figure tracking using learned dynamical models. In Proceedings of International Conference on Computer Vision, Kerkyra, Corfu, Greece, September 1999, pp. 94–101.

[28] N. Peterfreund, Robust tracking of position and velocity with Kalman snakes, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (6) (1999) 564–569.

[29] S. Ray, R. H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. In Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques (ICAPRDT'99), Calcutta, India, December 1999, pp. 137–143.

[30] M. Ridley, E. Nettleton, S. Sukkarieh, H. Durrant-Whyte. Tracking in decentralised air-ground sensing networks. In Proceedings of the Fifth International Conference on Information Fusion, vol. 1, Annapolis, USA, July 2002, pp. 616–623,.

[31] D. Wettergreen, C. Gaskett, and A. Zelinsky. Development of a visually-guided autonomous underwater vehicle. In Proceedings of Conference on OCEANS, vol. 2, September 1998, pp. 1200–1204.

[32] T. Yamane, Y. Shirai, J. Miura. Person tracking by integrating optical flow and uniform brightness regions. In Proceedings of IEEE International Conference on Robotics and Automation, vol. 4, Leuven, Belgium, May 1998, pp. 3267–3272.

[33] F. Yang and A. Balasuriya. Optical flow based speed estimation in auv target tracking. In Proceedings of MTS/IEEE Conference and Exhibition on OCEANS, vol. 4, Hawaii, USA, November, pp. 2377–2382.