Detection and Tracking of Moving Objects in SLAM using Vision Sensors

Yin-Tien Wang, Ying-Chieh Feng, and Duen-Yan Hung
Department of Mechanical and Electro-Mechanical Engineering
Tamkang University
Tamsui, Taipei Hsien, TAIWAN 25137
{ ytwang@mail; 696372076@s96; 697370400@s97}.tku.edu.tw

Abstract—**This paper presents algorithms for improving the detection of moving objects in robot visual simultaneous localization and mapping (SLAM). The method of speeded-up robust feature (SURF) is employed in the algorithm to provide a robust detection for image features as well as a better description of landmarks in the map of a visual SLAM system. Meanwhile, a moving object detection (MOD) is designed based on the correspondence constraint of the essential matrix for the feature points on image plane. Experiments are carried out on a hand-held camera sensor to verify the performances of the proposed algorithms. The results show that the integration of SURF and MOD is efficient to improve the robustness of robot SLAM.**

Keywords- Simultaneous Localization and Mapping (SLAM), Speeded Up Robust Features (SURF), Moving Object Detection (MOD), Vision Sensor

I. INTRODUCTION

In recent years, simultaneous localization and mapping (SLAM) have been successfully implemented and validated by many researchers [1-8]. In particular, Davison et al. [4] and Paz et al. [7] performed visual SLAM by using hand-held cameras as the sensors, and estimated the states of the free-moving camera systems which have unknown inputs. Nevertheless, the research in this paper aims at providing algorithms for improving the detection of moving objects as well as the extraction of image features in visual SLAM systems with hand-held vision sensors.

For visual SLAM systems, the features in the environment are detected and extracted by analyzing the image taken by the robot vision, and then the data association between the extracted features and the landmarks in the map is investigated. Many researchers [4,7] employed the concept by Harris and Stephens [9] to extract apparent corner features from one image and tracked these point features in the consecutive image. The descriptors of the Harris corner features are rectangle image patches. When the camera translates and rotates, the scale and orientation of the image patches will be changed. The detection of Harris corner will fail in this case, unless the variances in scale and orientation of the image patches are recovered. Instead of detecting corner features, some works [2-3] detect the features by using the scale-invariant feature transform (SIFT) method [10] which provides a robust image feature detector. The unique properties of image features extracted by SIFT method are further described by using a high-dimensional description vector [10]. However, the feature extraction by SIFT requires more computational cost than that by Harris's method [9]. To improve the computational speed, Bay et al. [11] introduced the concept of integral images and box filter to detect and extract the scale-invariant features, which they dubbed Speeded-Up Robust Features (SURF). The extracted SURF must be matched with the landmarks in the map of a SLAM system. The nearest-neighbor (NN) searching method [12] can be utilized to match high-dimensional data sets of description vectors.

In the literature, more and more researchers in the last decade solve the SLAM and the moving object detection (MOD) problems concurrently taking account the moving object information. Wang et al. [6] developed a consistency-based moving object detector and provided a framework to solve the SLAM and the detection and tracking of moving object (DATMO) problems simultaneously. Bibby and Reid [13] proposed a method that combines sliding window optimization and least-squares together with expectation maximization to do reversible model selection and data association that allows dynamic objects to be included directly into the SLAM estimate. Zhao et al. [14] uses GPS data and control inputs to achieve global consistency in dynamic environments. As a result, establishing the spatial and temporal relationship among the robot, stationary and moving objects in the environment serves as the basic for scene understanding.

We focus on the problems of robot state estimation and MOD in dynamic environments. An online SLAM algorithm with a moving object detector is developed based on the correspondence constraint for the essential matrix of image features. The essential matrix is calculated from the correspondences of the image features selected using the SURF method [11]. Moving object information is extracted from image plane and integrated into the MOD process such that the robustness of SLAM algorithm can be considerably improved, particularly in highly dynamic environments where surroundings of robots are dominated by non-stationary objects.

The contributions in this paper are two-fold. First, we develop algorithms to solve the problems for detection of image features as well as of moving objects, and then integrate it with the robot SLAM to improve the robustness of state estimation and mapping. Second, the improved SLAM is implemented on a hand-held camera system.

This paper was partially supported by the National Science Council in Taiwan under grant no. NSC99-2221-E-032-064 to Y.T. Wang.

978-1-4244-7935-1/11/$26.00 ©2011 IEEE

II. SLAM WITH A FREE-MOVING VISION SENSOR

SLAM is a target tracking problem for the robot system during navigating in the environment [1]. The targets to be tracked include the state of the robot itself and of the landmarks in the environment. The state sequence of the SLAM system at time step k can be expressed as

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, w_{k-1}) \tag{1}$$

where $\mathbf{x}_k$ is the state vector; $\mathbf{u}_k$ is the input; w_k is the process noise. The objective of the tracking problem is to recursively estimate the state $\mathbf{x}_k$ of the target according to the measurement $\mathbf{z}_k$ at k,

$$\mathbf{z}_k = g(\mathbf{x}_k, v_k) \tag{2}$$

where v_k is the measurement noise. In this paper, a hand-held vision sensor is utilized as the only sensing device for the measurement in SLAM system. We treat this hand-held camera as a free-moving robot system with unknown inputs. The states of the robot system are estimated by solving the recursive SLAM problem using the extended Kalman filter (EKF) [1],

$$\begin{aligned}
\mathbf{x}_{k|k-1} &= f(\mathbf{x}_{k-1|k-1}, \mathbf{u}_{k-1}, 0) \\
\boldsymbol{P}_{k|k-1} &= \boldsymbol{A}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{A}_k^T + \boldsymbol{W}_k \boldsymbol{Q}_{k-1} \boldsymbol{W}_k^T \\
\boldsymbol{K}_k &= \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T + \boldsymbol{V}_k \boldsymbol{R}_k \boldsymbol{V}_k^T)^{-1} \\
\mathbf{x}_{k|k} &= \mathbf{x}_{k|k-1} + \boldsymbol{K}_k (\mathbf{z}_k - g(\mathbf{x}_{k|k-1}, 0)) \\
\boldsymbol{P}_{k|k} &= (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_{k|k-1}
\end{aligned} \tag{3}$$

where $\mathbf{x}_{k|k-1}$ and $\mathbf{x}_{k|k}$ represent predicted and estimated state vectors, respectively; $\boldsymbol{K}_k$ is Kalman gain matrix; $\boldsymbol{P}$ denotes the covariance matrix, respectively; $\boldsymbol{A}_k$ and $\boldsymbol{W}_k$ are the Jacobian matrices of the state equation f with respect to the state vector $\mathbf{x}_k$ and the noise variable w_k, respectively; $\boldsymbol{H}_k$ and $\boldsymbol{V}_k$ are the Jacobian matrices of the measurement g with respect to the state vector $\mathbf{x}_k$ and the noise variable v_k, respectively.

A. Motion Model

The motion of the hand-held camera is presumed to be at constant velocity (CV), and the acceleration is caused by an impulse noise from the external force. The state vector of the SLAM system in Eqn. (1) is chosen as:

$$\mathbf{x} = [\mathbf{x}_C \ \mathbf{m}_1 \ \mathbf{m}_2 \cdots \mathbf{m}_n]^T \tag{4}$$

$\mathbf{x}_C$ is a 12×1 state vector of the camera including the three-dimensional vectors of position $\boldsymbol{r}$, rotational angle $\boldsymbol{\phi}$, linear velocity $\mathbf{v}$, and angular velocity $\boldsymbol{\omega}$, all in the world frame; $\mathbf{m}_i$ is the three-dimensional (3D) coordinates of i^{th} image feature or landmark in the world frame; n is the number of the image features. Therefore, the state vector $\mathbf{x}_C$ of the camera with a CV motion model at time step k is expressed as:

$$\mathbf{x}_{Ck} = \begin{bmatrix} \boldsymbol{r}_k \\ \boldsymbol{\phi}_k \\ \mathbf{v}_k \\ \boldsymbol{\omega}_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_{k-1} + (\mathbf{v}_{k-1} + \boldsymbol{w}_{\mathbf{v}k-1})\Delta t \\ \boldsymbol{\phi}_{k-1} + (\boldsymbol{\omega}_{k-1} + \boldsymbol{w}_{\omega k-1})\Delta t \\ \mathbf{v}_{k-1} + \boldsymbol{w}_{\mathbf{v}k-1} \\ \boldsymbol{\omega}_{k-1} + \boldsymbol{w}_{\omega k-1} \end{bmatrix}$$

where $\boldsymbol{w}_{\mathbf{v}}$ and $\boldsymbol{w}_{\omega}$ are linear and angular velocity noise caused by acceleration, respectively.

B. Vision Sensor Model

The measurement vector of the vision system is expressed as

$$\mathbf{z}_k = [\mathbf{z}_{1k} \ \mathbf{z}_{2k} \cdots \mathbf{z}_{mk}]^T$$

m is the number of the observed image features in current measurement. The perspective projection method [15] is employed to model the transformation from 2D image plane to 3D space coordinate system. For one observed image feature, the measurement is

$$\mathbf{z}_{ik} = \begin{bmatrix} I_{ix} \\ I_{iy} \end{bmatrix} = \begin{bmatrix} u_0 + f_C k_u \dfrac{h_{ix}^C}{h_{iz}^C} \\ v_0 + f_C k_v \dfrac{h_{iy}^C}{h_{iz}^C} \end{bmatrix} \quad for \ i = 1, 2, \cdots, m \tag{5}$$

where f_C is the focal length of the camera denoting the distance from the camera center to the image plane; (u_0, v_0) is the offset pixel vector of the pixel image plane; k_u and k_v are the image pixel correctional parameters. Assuming that there is no distortion phenomenon on the image plane and we make k_u and k_v as 1; $\boldsymbol{h}_i^C = [h_{ix}^C \ \ h_{iy}^C \ \ h_{iz}^C]^T$ is the ray vector of the image features in the camera frame. The 3D coordinates of an image feature or landmark in world frame, as shown in Figure 1, is given as

$$\mathbf{m}_i = [X_i \ \ Y_i \ \ Z_i]^T = \boldsymbol{r} + \boldsymbol{R}_C^W \boldsymbol{h}_i^C \tag{6}$$

$\boldsymbol{R}_C^W$ is the rotational matrix from the world frame {W} to the camera frame {C}, represented by using the elementary rotations [16],

$$\boldsymbol{R}_C^W = \begin{bmatrix} c\phi_y c\phi_z & s\phi_x s\phi_y c\phi_z - c\phi_x s\phi_z & c\phi_x s\phi_y c\phi_z + s\phi_x s\phi_z \\ c\phi_y s\phi_z & s\phi_x s\phi_y s\phi_z + c\phi_x c\phi_z & c\phi_x s\phi_y s\phi_z - s\phi_x c\phi_z \\ -s\phi_y & s\phi_x c\phi_y & c\phi_x c\phi_y \end{bmatrix} \tag{7}$$

where $c\phi$=cosϕ and $s\phi$=sinϕ; ϕ_x, ϕ_y and ϕ_z are the corresponding rotational angles in world frame. We can utilize Eqn. (6) to calculate the ray vector of an image feature in the camera frame.

The coordinates of the feature in the image plane are obtained by substituting Eqns. (6)-(7) into Eqn. (5),

$$I_{ix} = u_0 + f_C \frac{c\phi_y c\phi_z (X_i - r_x) + c\phi_y s\phi_z (Y_i - r_y) - s\phi_y (Z_i - r_z)}{(c\phi_x s\phi_y c\phi_z + s\phi_x s\phi_z)(X_i - r_x) + (c\phi_x s\phi_y s\phi_z - s\phi_x c\phi_z)(Y_i - r_y) + c\phi_x c\phi_y (Z_i - r_z)}$$

$$I_{iy} = v_0 + f_C \frac{(s\phi_x s\phi_y c\phi_z - c\phi_x s\phi_z)(X_i - r_x) + (s\phi_x s\phi_y s\phi_z + c\phi_x c\phi_z)(Y_i - r_y) + s\phi_x c\phi_y (Z_i - r_z)}{(c\phi_x s\phi_y c\phi_z + s\phi_x s\phi_z)(X_i - r_x) + (c\phi_x s\phi_y s\phi_z - s\phi_x c\phi_z)(Y_i - r_y) + c\phi_x c\phi_y (Z_i - r_z)}$$

Moreover, the elements of the Jacobian matrices $\boldsymbol{H}_k$ and $\boldsymbol{V}_k$ are determined by taking the derivative of $\mathbf{z}_i$ with respect to the state $\mathbf{x}_k$ and the measurement noise v_k. The Jacobian matrices are obtained for the purpose of calculating the matrix of the innovation covariance [17].

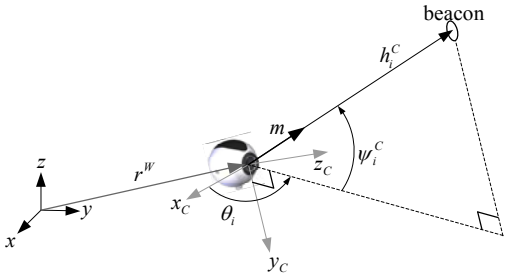

Figure 1. Coordinate setting

C. *Speeded Up Robust Features (SURF)*

The basic concept of a scale-invariant method is to detect image features by investigating the determinant of Hessian matrix $\boldsymbol{H}$ in scale space [18]. In order to speed up the detection of image features, Bay et al. [11] utilize integral images and box filters to process on the image instead of calculating the Hessian matrix, and then the determinant of Hessian matrix is approximated by

$$\mathbf{det}(\boldsymbol{H})_{\text{approx.}} = D_{xx} D_{yy} - (wD_{xy})^2$$

where D_{ij} are the images filtered by the corresponding box filters; w is a weight constant. The interest points or features are extracted by examining the extreme value of determinant of Hessian matrix. Furthermore, the unique properties of the extracted SURF are described by using a 64-dimensional description vector as shown in Fig. 2 [10].

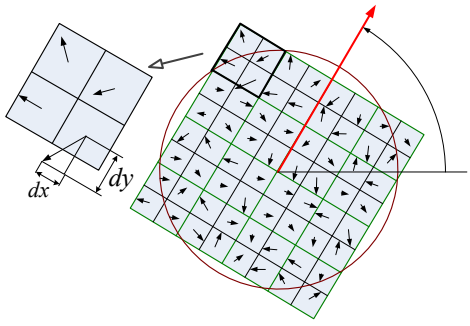

Figure 2. 64-dimensional description vector for SURF

D. *Implementation of EKF SLAM*

The EKF SLAM is implemented on the free-moving vision system by integrating the motion and sensor models, as well as the extraction of SURF. A flowchart for the developed SLAM system is depicted in Fig. 3. The images are captured by the camera and features are extracted using SURF method. In the SLAM flowchart, data association in between the landmarks in the database and the image features of the extracted SURF is carried out using the nearest neighbor ratio matching strategy [10]. A tactic is designed to manage the newly extracted features and the bad features in SLAM system. The properties of the newly extracted features are investigated and the moving objects will be discriminated from the stationary objects using the proposed detection algorithm. Those features which are not continuously detected at each time step will be treated as bad features and erased from the state vector in Eqn. (4).

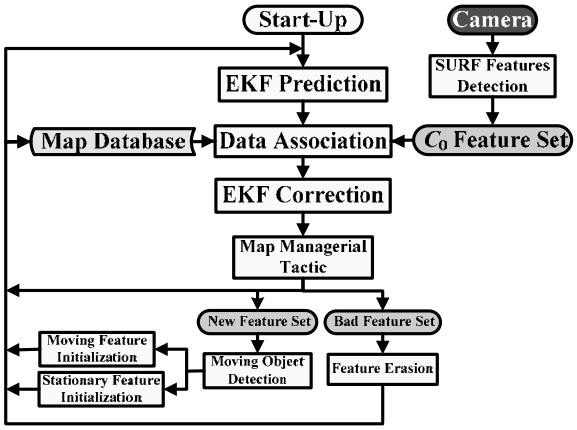

Figure 3. Flowchart of EKF-based visual SLAM

III. MOVING OBJECT DETECTION AND TRACKING

The function block of moving object detection (MOD) in Fig. 3 is designed based on the concept of the correspondence constraint for the essential matrix. The essential matrix is defined as [19]

$$\boldsymbol{E} = [\boldsymbol{t}]_\times \boldsymbol{R} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \boldsymbol{R}$$

where $\boldsymbol{R}$ is the rotation matrix and $\boldsymbol{t}$ is the translation vector; $[\boldsymbol{t}]_\times$ is the matrix representation of the cross product with $\boldsymbol{t}$. The essential matrix $\boldsymbol{E}$ is calculated by using the EKF estimator. The correspondence constraint for the same feature in two sequential images can be expressed as

$$h_d^{C\,T} \boldsymbol{E} h_d'^{C} = 0 \qquad (8)$$

where h_d^C and $h_d'^C$ are the homogenous normalized image coordinates of the same feature abstracted from image 1 and 2, respectively.

In the moving object tracking (MOT) problem, the targets to be tracked include the state and motion mode of the moving object and of the landmarks in the environment. The state of the MOT system at time step k can be expressed as

$$\mathbf{x}_k=[\boldsymbol{o}_k\ \boldsymbol{s}_k]^T$$

where $\boldsymbol{o}_k$ is state of the moving object; $\boldsymbol{s}_k$ is object motion mode. The MOT problem can be expressed as a probability density function (pdf) in Bayesian probability

$$p(\boldsymbol{o}_k, \boldsymbol{s}_k|\mathbf{z}_{1:k}) = p(\boldsymbol{o}_k|\boldsymbol{s}_k, \mathbf{z}_{1:k}) \cdot p(\boldsymbol{s}_k|\mathbf{z}_{1:k}) \qquad (9)$$

where $p(\boldsymbol{o}_k|\boldsymbol{s}_k,\mathbf{z}_{1:k})$ is state inference; $\mathbf{z}_{1:k}$ is the set of measurements for time t=1 to k; $p(\boldsymbol{s}_k|\mathbf{z}_{1:k})$ is the mode learning. The EKF-based interacting multiple model (IMM) estimator [20] is utilized in the paper to estimate the motion mode of a moving object. The state is computed at time k under each possible current model using r filters, with each filter using a different combination of the previous model-conditioned estimates. The mode $\mathcal{M}^i$ at time k is assumed to be among the possible r modes

$$\mathcal{M}^i \in \boldsymbol{\mathcal{M}} = \left\{\mathcal{M}^1, \mathcal{M}^2, \cdots, \mathcal{M}^r\right\}$$

Given r motion models, the object state $\boldsymbol{o}_k$ in Eqn. (9) is estimated. On the other hand, the coordinates of ith stationary landmark in 3D space is expressed as

$$\mathbf{m}_i = [X_i \quad Y_i \quad Z_i]^T$$

IV. Experimental Results

The SLAM system with moving object detection and tracking is implemented on a Window-based NB with a free-moving camera sensor. In this experiment, two moving objects are tracked at one corner of our laboratory. Object 1 is carried by a person, as shown in Fig. 4; Object 2 is pulled to move from left- to right-side by a wire. Two motion models, CV and CA, are utilized for the IMM estimator in this example. A rectangle area is drawn in the image for each object to extract the image feature on the corresponding object. One typical tracking result is depicted in Fig. 5 for 35th image frame. In the figure, the captured image, top-view plot of the estimated states and the estimated velocities are shown in the left-, middle- and right-panel of the figure. In the left-panel of Fig.5, image feature #1 is extracted on Object 1. The estimated state of feature #1 is illustrated in a two-dimensional plot and indicated as a circle in the middle-panel. Meanwhile, the estimated velocity components of feature #1, v_x and v_y, are plot in the right-panel. More experimental results are illustrated in Figs. 6-10. As shown in Fig. 6 for 85th image frame, image feature #3 is extracted on Object 2. Then two objects begin to move, as shown in Fig. 7. The solid lines in middle-panel depict the estimated trajectories of the moving objects. In the right-panel, the solid lines indicate the velocity components of feature #1 on Object 1 and the dash lines describe the velocity components of feature #3 on Object 2. In Fig. 8, Objects 1 stops and waits for Object 2 to pass. The estimated velocity components are reduced to zero, as shown in right-panel of the figure. The object 1 moves again as shown in Fig. 9. Two objects stop their motion, as shown in Fig. 10, and their estimated velocity components are reduced to zero.

Figure 4. Two moving objects are tracked

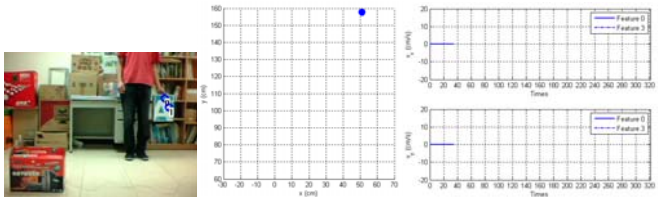

Figure 5. 35th frame: add new feature #1 on object 1

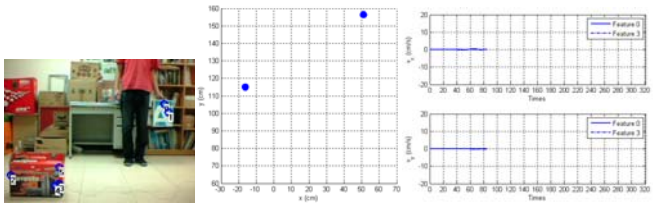

Figure 6. 85th frame: add new feature #2 on object 2

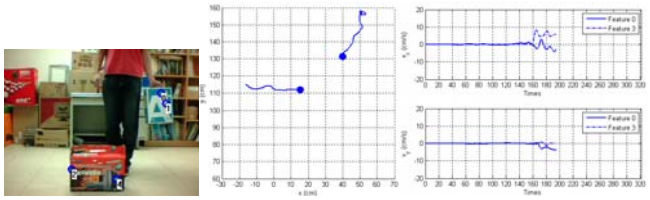

Figure 7. 195th frame: objects 1 and 2 moving

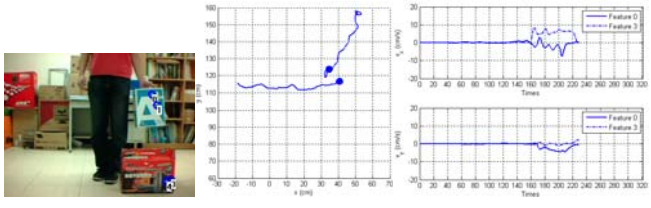

Figure 8. 230th frame: objects 1 stops and waits for object 2 passing

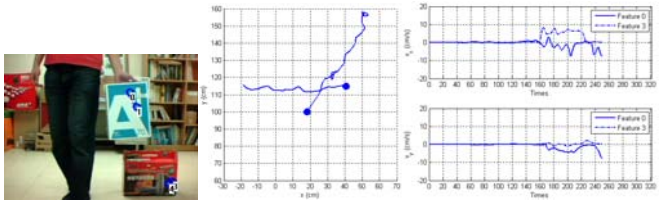

Figure 9. 250th frame: objects 1 moves again and object 2 stops

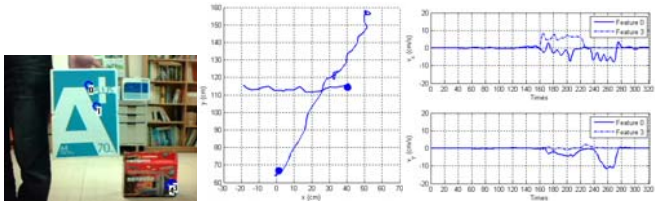

Figure 10. 320th frame: both objects 1 and 2 stop

V. CONCLUSIONS

In this research, we developed an algorithm for detection and tracking of moving objects to improve the robustness of robot visual SLAM with SURF. Experimental works were also carried out in this paper and the results showed that the SLAM with the proposed algorithm has the capability to support the camera system simultaneously navigating and tracking moving objects in dynamic environments.

REFERENCES

[1] R. Smith, M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics," In *Autonomous Robot Vehicles*, I.J. Cox and G.T. Wilfong, Eds., Springer-Verlog, pp.167-193, 1990.

[2] N. Karlsson, E.D. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M.E. Munich, "The vSLAM Algorithm for Robust Localization and Mapping," Proceedings of *IEEE International Conference on Robotics and Automation*, pp.24-29, 2005.

[3] R. Sim, P. Elinas and J.J. Little, "A Study of the Rao-Blackwellised Particle Filter for Efficient and Accurate Vision-Based SLAM," *International Journal of Computer Vision*, vol.74, no.3, pp.303-318, 2007.

[4] A.J. Davison, I.D. Reid, N.D. Molton, and O. Stasse, "Mono SLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.29, no.6, pp.1052-1067, 2007.

[5] M. Montemerlo and S. Thrun, *FastSLAM*, Springer-Verlag, 2007.

[6] C.C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *International Journal of Robotics Research*, vol.26, no.9, pp.889-916, 2007.

[7] L.M. Paz, P. Pinies, J.D. Tardos, and J. Neira, "Large-Scale 6-DOF SLAM with Stereo-in-Hand," *IEEE Transactions on Robotics*, vol.24, no.5, pp.946-957, 2008.

[8] Y.T. Wang, M.C. Lin, and R.C. Ju, "Visual SLAM and Moving Object Detection for a Small-size Humanoid Robot," *International Journal of Advanced Robotic Systems*, vol.7, no.2, pp.133-138, 2010.

[9] C. Harris and M. Stephens, "A combined corner and edge detector," *Proceedings of the 4th Alvey Vision Conference*, pp.147-151, 1988.

[10] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol.60, no.2, pp. 91-110, 2004.

[11] H. Bay, T. Tuytelaars, and L. Van Gool, SURF: "Speeded up robust features." *The ninth European Conference on Computer Vision*, 2006.

[12] G. Shakhnarovich, T. Darrell, and P. Indyk, *Nearest-neighbor methods in learning and vision*, MIT Press, 2005.

[13] C. Bibby and I. Reid, "Simultaneous Localisation and Mapping in Dynamic Environments (SLAMIDE) with Reversible Data Associa," *Proceedings of Robotics: Science and Systems III*, 2007.

[14] H. Zhao, M. Chiba, R. Shibasaki, X. Shao, J. Cui, and H. Zha, "SLAM in a Dynamic Large Outdoor Environment using a Laser Scanner," *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, California, 2008.

[15] S. Hutchinson, G.D. Hager, and P.I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol.12, no.5, pp.651-670, 1996.

[16] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators*, McGraw-Hill, New York, NY, 1996.

[17] Y.T. Wang, M.C. Lin, R.C. Ju, and Y.W. Huang, "Image Feature Initialization for SLAM and Moving Object Detection," *Innovative Computing, Information and Control -- Express Letters*, vol.3, no.3(A), pp.477-482, 2009.

[18] T. Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol.30, no.2, pp79-116, 1998.

[19] H.C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, *Nature*, vol.293, pp.133-135, 1981.

[20] H. Blom, A.P**,** and Y. Bar-Shalom, "The interacting multiple-model algorithm for systems with Markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol.33, pp.780-783, 1988.