2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Hybrid Vision-based SLAM Coupled with Moving Object Tracking

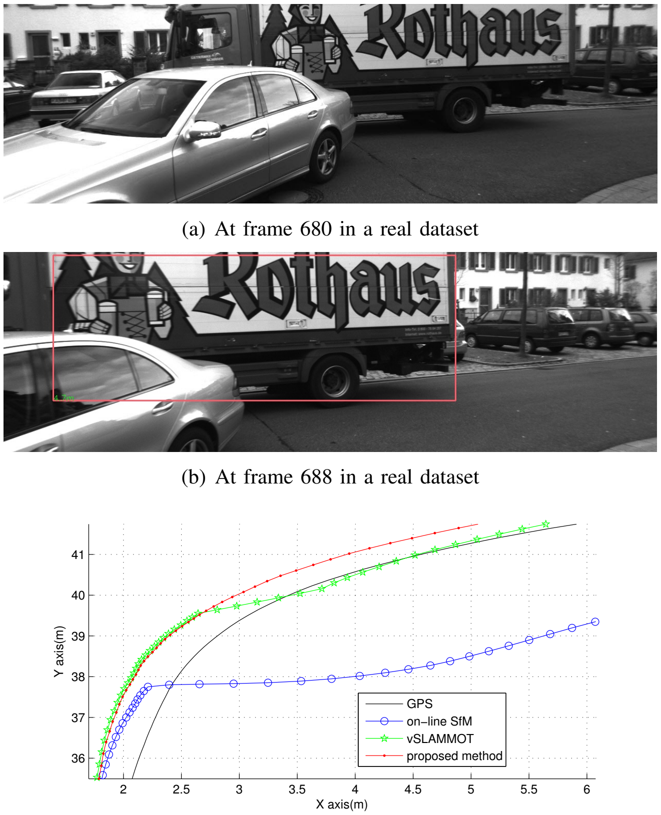
Jihong Min[1], Jungho Kim[2], Hyeongwoo Kim[3], Kiho Kwak[1] and In So Kweon[3]

***Abstract*— In this paper we propose a hybrid vision-based SLAM and moving objects tracking (vSLAMMOT) approach. This approach tightly combines two key methods: a superpixel-based segmentation to detect moving objects and a Rao-Blackwellized Particle Filter to estimate a stereo-vision-based SLAM posterior. Most successful methods perform vision-based SLAM (vSLAM) and track moving objects independently. However, we pose both vSLAM and moving object tracking as a single correlated problem to leverage the performance. Our approach estimates the relative camera motion using the previous tracking result, and then detects moving objects from the estimated camera motion recursively. Moving superpixels are detected by a Markov Random Field (MRF) model which uses spatial and temporal information of the moving objects. We demonstrate the performance of the proposed approach for vSLAMMOT using both synthetic and real datasets and compare the performance with other methods.**

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is one of the most widely researched areas by the robotics and computer vision communities. The goal of SLAM is to build up and update a map within the operating environment of a robot, and localize the robot within the map minimizing the localization and mapping error simultaneously. For the last two decades, many researchers have addressed SLAM algorithms using several different types of sensors. One of the cases is vision-based SLAM using primarily visual(camera) sensors. The vision-based SLAM allows the development of mobile autonomous systems that restrict costs and size.

Many vision-based SLAM (vSLAM) approaches have shown remarkable performance in both indoor and outdoor environments. Most of vSLAM approaches assume that the unknown environment is static, containing non-moving objects. This assumption is reasonable in some cases, but moving objects and dynamic environments should be considered to improve the performance of vSLAM. For example, some moving objects are large in number or dominant part of the scene as shown in Fig. 1(a), most vSLAM approaches degrade their performance.



(a) At frame 680 in a real dataset

(b) At frame 688 in a real dataset

(c) Localization results (Top view)

Fig. 1. (a) shows that moving objects are dominant at frame 680. In (b), a red rectangle represents the moving object detected by the proposed method. (c) shows the localization results of on-line SfM [1], vSLAMMOT and the proposed method.

The vSLAM approaches with detection and tracking of moving objects improve the SLAM accuracy and detection and tracking performance simultaneously [2] [3] [4]. Sola [2] proposed a framework to estimate the camera pose, the static map and the trajectories of the moving objects by separating the SLAM algorithm from tracking one. Lin and Wang [3] proposed a stereo vision-based SLAM with moving object tracking (SLAMMOT) approach to overcome the problems of their previous approach using a single camera. Migliore et al. [4] proposed a monocular SLAM and moving object tracking in which moving objects are detected by a simple statistic test and tracked by separated bearing only trackers. However, since these approaches do not consider each tracking result in the recursive vSLAM process, the image features or landmarks in moving objects increase the localization and mapping errors.

The image features and landmarks of moving objects to estimate visual odometry are able to increase errors in SLAM. Generally, the performance of vSLAM using a motion model with a visual odometry prior is superior to using a constant linear and angular velocity motion model

[1] Min and Kwak are with the Agency for Defense Development, Republic of Korea
{happymin77@gmail.com, kkwak.add@gmail.com}
[2] Kim is with Korea Electronics Technology Institute
kjhforce@gmail.com
[3] Kim and Kweon with the Department of Electrical Engineering, KAIST
{hyeongwoo.kim@rcv.kaist.ac.kr,
iskweon@kaist.ac.kr}

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

in recent researches [5] [6]. However, Wangsiripitak and Ess [7] [8] touched the problem of the vSLAM using a motion model with a visual odometry prior. Wangsiripitak and Murray [7] proposed SLAM for tracking a predefined moving object using a monocular camera. Ess et al. [8] presented a mobile robot equipped with a stereo camera which estimates sequential visual odometry with tracking objects by detection. They did not solve the problem in general cases because they assume that the moving objects in their approach are known previously.

In the vSLAMMOT method using visual odometry, estimating the camera pose and detecting moving objects are tightly coupled, that is the accuracy of camera pose estimation depends on the performance of moving object detection, and vice-versa. Therefore, more static features are detected by excluding moving features, more stable performance of SLAM is achieved.

In this paper, we address a hybrid vision-based SLAM and moving objects tracking (vSLAMMOT) approach. To solve the coupled problem between SLAM and moving object tracking, we introduce a hybrid pose representation. The basic concept of our approach is that visual odometry is estimated by excluding tracked image regions from previous images where moving objects exist and moving objects are detected using outlier regions from a SLAM framework. In the hybrid pose, the distribution of the camera poses is modeled by a data-driven proposal distribution and estimated by a sampling approach. Moving regions are detected using the proposed graph-cut-based method which uses spatial and temporal information of the objects. With a simple data-association method, moving objects are classified from the estimated moving regions.

Similar to successful works on SLAMMOT [2][9], we use Bayesian filtering to provide a probabilistic estimation which can exploit the uncertainties involved in measurements, a hybrid pose, a static map, and the moving objects. We employed a Rao-Blackwellized Particle Filter (RBPF) [10] to estimate the proposed vSLAMMOT posterior, and make the features conditionally independent by estimating the hybrid path rather than the hybrid pose.

This paper is organized as follows. In Section II, the overall procedure and our hybrid vision-based SLAM approach are briefly described. In Section III, the proposed superpixel-based moving object detection method and data-association method are described. In Section IV, we show some experimental results and comparisons with other methods. We then discuss future works and conclude our paper in Section V.

II. Theoretical Foundation

First we summarize the list of notations used for the vSLAMMOT formulation as below:

- A_t, A^t: the subscript t representing at time t and the superscript t representing up to time t.
- $A^{[i]}$: the superscript i in brackets indicating the i^{th} particle.
- n_t: a hybrid pose vector which is composed of a camera pose and label of image pixels.

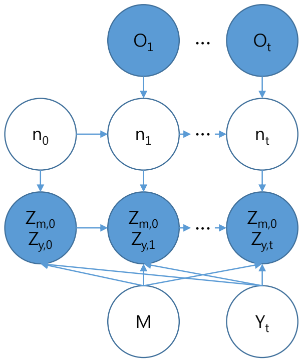


Fig. 2. Graphical model for proposed vSLAMMOT framework. Blue circles and white circles represent observed and hidden variables, respectively.

- x_t: a 6D camera pose vector describing the 3D position and 3D orientation.
- l_t: a pixel label vector of moving objects and static regions in a left image.
- o_t: features matched between four images, the left and right images of two consecutive frames at time $t-1$ and t. In this paper, we call this match a quadmatch and use the source code in [1], This match is used to estimate relative motion and track superpixels between two consecutive frames.
- z_s: the observations of static landmarks.
- z_{y_k}: the observations of the k^{th} moving objects. A set of superpixels [11] is used for a moving object assuming that all pixels in a superpixel belong to the same object.
- M, Y: a set of landmarks consisting of the global map, and a set of moving objects, respectively.
- m_n, y_k: the 3D position vector of the n^{th} landmark and k^{th} moving object, respectively.

A. Overall procedure

Before explaining the proposed method, we summarize the overall procedure as follows:

1) Generating superpixels in a left image.
2) Performing a quadmatch using two consecutive stereo images.
3) Generating particles from a given proposal distribution.
4) Detecting moving object by solving proposed spatio-temporal MRF.
5) Performing data-association to track moving objects.
6) Finding new features and updating landmarks.
7) Adding new landmarks.
8) Re-sampling particles.

B. A Proposed RBPF formulation for vSLAMMOT

Since the estimation of the camera pose and moving object detection are highly related, we introduce a hybrid pose including a camera pose, and labels of moving objects, and static regions in the image, as follows:

$$n_t = < x_t, l_t > . \tag{1}$$

We can now state the problem of vSLAMMOT as the estimation of the joint posterior of the hybrid camera path

n^t and the map Θ with given sequential observations z^t consisting of stationary features z_s^t and moving features $z_{y_k}^t$, and feature matches o^t as follows:

$$p\left(n^t, \Theta | o^t, z^t\right), \tag{2}$$

and the graphical model of proposed vSLAMMOT is shown in Fig. 2.

To solve this problem, we employ the RBPF which can estimate the distribution of the hybrid path with given stationary and moving features. We assume that measurements can be decomposed into measurements of stationary and moving objects.

By Bayes' rule and the independence assumption between stationary and moving features, this joint distribution can be factorized into three distributions, the posterior distribution of the hybrid path, the multiple independent distributions for the landmarks and moving objects that are conditioned on the hybrid path estimation as follows:

$$\begin{aligned}
&p\left(n^t, \Theta | o^t, z^t\right) \\
&= p\left(n^t | o^t, z^t\right) p\left(\Theta | n^t, o^t, z^t\right) \\
&= p\left(n^t | o^t, z^t\right) p\left(M | n^t, z^t\right) p\left(Y_t | n^t, z^t\right) \\
&= \underbrace{p\left(n^t | o^t, z^t\right)}_{hybrid\ path\ posterior} \underbrace{\prod_{n=1}^{N} p\left(m_n | n^t, z_s^t\right)}_{landmark\ estimators} \underbrace{\prod_{k=1}^{L} p\left(y_{k,t} | n^t, z_{y_k}^t\right)}_{moving\ object\ estimators},
\end{aligned} \tag{3}$$

where N is the number of landmarks and L is the number of moving objects.

C. Hybrid Path Posterior

We now address the estimation of the hybrid path using a set of weighted particles. The particles are generated from a given proposal distribution as follows:

$$n_t^{[i]} \sim p(n_t | o^t, z^t, n^{t-1,[i]}). \tag{4}$$

Since the distribution of a camera path is not a Gaussian distribution and a labeling problem is NP-hard, we approximately estimate the distribution of the hybrid path using a sampling approach by two steps: 1) drawing samples for the camera pose using quadmatches in predicted static regions as Eq. 5, and then 2) detecting moving regions and labeling objects regions based on the sampled camera pose in the first step as

$$\begin{aligned}
&1^{st}\ step\ :\ x_t^{[i]} \sim p(x_t | o^t, z^t, n^{t-1,[i]}) \\
&2^{nd}\ step\ :\ l_t^{[i]} = \arg\max_{l_t} p(l_t | x_t^{[i]}, o^t, z^t, n^{t-1,[i]}).
\end{aligned} \tag{5}$$

Relative pose samples between consecutive frames are generated using the three-point algorithm with RANSAC — the image points of three known world points (2D-3D corresponding points) provide the possible camera poses. In RANSAC, three 2D-3D corresponding points are randomly selected and the estimated camera poses are directly used to generate particles.

Generally, features from moving objects are determined as outliers. However, when moving objects are dominant in an image as shown in Fig. 1, incorrect camera motion is estimated and moving features are classified as inliers. To reduce this effect, we exclude moving features using predicted static regions when estimating camera motion. Here, in order to predict static regions, we simply track superpixels of moving objects using the majority of quadmatches and set all image regions except the tracked moving superpixels as static regions.

Given the camera pose, we estimate moving objects by solving proposed spatio-temporal Markov Random Field (MRF) and performing simple data-association. This process will be explained in section III.

D. Landmark Estimators

The conditional distribution of the landmark, $p\left(m_n | n^t, z_s^t\right)$, is represented by a three-dimensional Gaussian distribution. The mean and covariance of a landmark are estimated by an unscented transform [12] through the N-view triangulation function [13] — a minimal set of sample points (called sigma points) around the measured image point is propagated through the non-linear N-view triangulation functions (in our case, first, middle and last camera poses at which the feature is observed), and then the mean and covariance of the three-dimensional position of a landmark are estimated.

E. Moving Object Estimators

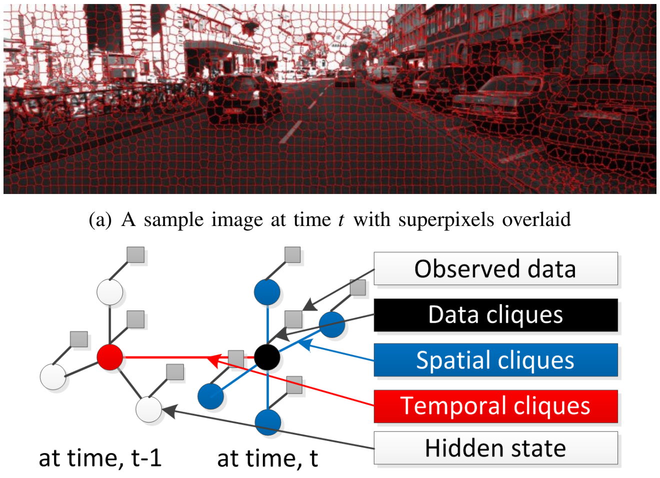
We assume that the motion model of moving objects is a constant velocity model with Gaussian noise. Similar to estimating the distribution of the landmarks, moving object estimates are conditioned on the camera path. Thus L Extended Kalman Filters (EKFs) are attached to each particle, as follows:

$$\begin{aligned}
&p\left(y_{k,t} | n^t, z_{y_k}^t\right) \\
&= \eta\, p\left(z_{y_k}^t | y_{k,t}, x^t, z_{y_k}^{t-1}\right) p\left(y_{k,t} | x^t, z_{y_k}^{t-1}\right) \\
&= \eta\, p\left(z_{y_k}^t | y_{k,t}, x^t\right) \int p\left(y_{k,t} | y_{k,t-1}\right) p\left(y_{k,t-1} | x^{t-1}, z_{y_k}^{t-1}\right) dy_{k,t-1},
\end{aligned} \tag{6}$$

F. Calculating Importance Weights

By Bayes' rule and the Markov assumption, the importance weight of the i^{th} particle is computed based on the current observation z_t with the hybrid pose as follows:

$$\begin{aligned}
w_t^{[i]} &= \frac{p\left(n^{t,[i]} | o^t, z^t\right)}{p\left(n^{t,[i]} | o^t, z^{t-1}\right)} \\
&= \eta \frac{p\left(z_t | n^{t,[i]}, o^t, z^{t-1}\right) p\left(n^{t,[i]} | o^t, z^{t-1}\right)}{p\left(n^{t,[i]} | o^t, z^{t-1}\right)} \\
&= \eta\, p\left(z_{s,t} | n^{t,[i]}, o^t, z^{t-1}\right) p\left(z_{y_k,t} | n^{t,[i]}, o^t, z^{t-1}\right) \\
&= |2\pi\Sigma_{s,t}|^{-\frac{1}{2}} exp\{-\frac{1}{2}(z_{s,t} - \hat{z}_{s,t})^T \Sigma_{s,t}^{-1} (z_{s,t} - \hat{z}_{s,t})\} \times \\
&\quad |2\pi\Sigma_{y_k,t}|^{-\frac{1}{2}} exp\{-\frac{1}{2}(z_{y_k,t} - \hat{z}_{y_k,t})^T \Sigma_t^{-1} (z_{y_k,t} - \hat{z}_{y_k,t})\},
\end{aligned} \tag{7}$$



(a) A sample image at time t with superpixels overlaid

(b) The proposed spatio-temporal MRF model defining the neighbors of a superpixel.

Fig. 3. Each superpixel in (a) corresponds to the nodes of the spatio-temporal MRF model in (b). The spatial cliques favors the spatial smoothness of the neighboring superpixels at time t while the temporal cliques favors the smoothness of the corresponding superpixels between time t and $t-1$.

where $\Sigma_{s,t}$ and $\Sigma_{y_k,t}$ represent the observation uncertainties of the static landmark and the moving object, respectively. $\hat{z}_{s,t}$ and $\hat{z}_{y_k,t}$ are predicted image coordinates of the static landmark and the moving object, respectively.

III. MOVING OBJECT DETECTION

In this section, we address the problem of detecting moving objects and determining data-association using the given camera pose, superpixels, sparse quadmatches and previous labels of moving objects. Since each hybrid path has its own label as shown in Eq. 5, the computational complexity of labeling is proportion to the number of particles. To reduce complexity, we perform a superpixel-level labeling by assuming that all pixels within a superpixel belong to the same object. Fig. 3(a) shows an example of superpixels [11]. By detecting moving objects in the superpixel accuracy, the computational complexity is drastically decreased compared to the single pixel-wise model.

Since a moving object covers a volume in a spatial and temporal space, we use spatio-temporal MRF to detect moving objects. After detecting moving objects, we performed data-association to distinguish a moving object from other adjacent moving objects.

A. Spatio-temporal Markov Random Field for Detecting Moving Object

In this study, the problem of detecting moving objects in the current image is equivalent to determining whether each superpixel belongs to moving objects or not based on the given camera path and observations.

Using the spatio-temporal MRF model in Fig. 3(b), a superpixel at time t has a hidden state node (black node) and a corresponding data node (gray square). The network consists of nodes connected by edges that indicate conditional dependency. The state node has spatial neighbors (blue node) and temporal neighbors (red node) only when quadmatches exist between two superpixels. This spatio-temporal MRF model satisfies the first order Markov characteristics, so a state node only interacts with its neighboring nodes.

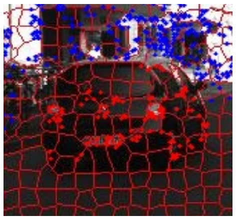


Fig. 4. Blue and red dots represent inliers and outliers, respectively. Since the car is moving, superpixels in image region of the car have many outliers. On the other hand, superpixels in image region of the building have many inliers.

The spatio-temporal MRF is defined in terms of the the energy function which is composed of a data function $D(\cdot)$, a temporal coherence function $D_t(\cdot)$ and a spatial coherence function $V_s(\cdot)$ as follows:

$$E(L) = \sum_{p\in P} D(L_p) + \lambda_1 \sum_{p\in P, p'\in P'} D_t(L_p, L_{p'}) + \lambda_2 \sum_{(p,q)\in N} V_s(L_p, L_q), \tag{8}$$

where $L = \{L_p \in \{0,1\} | p \in P\}$ is a binary labeling of an image P, p and q are superpixels, p' is the temporally matched superpixel of p, and N is a set of all pairs of neighboring superpixels. Here, λ_1 and λ_2 are the parameters to balance the temporal and spatial coherence terms respectively.

When the camera path and quadmatches are given, we can estimate the difference between measured and projected image points of a feature which is matched between four images. If the difference is bigger than a pre-defined value, we call this point an outlier otherwise an inlier. The data function is determined by the ratio of the number of the inliers and outliers within the superpixel p as follows:

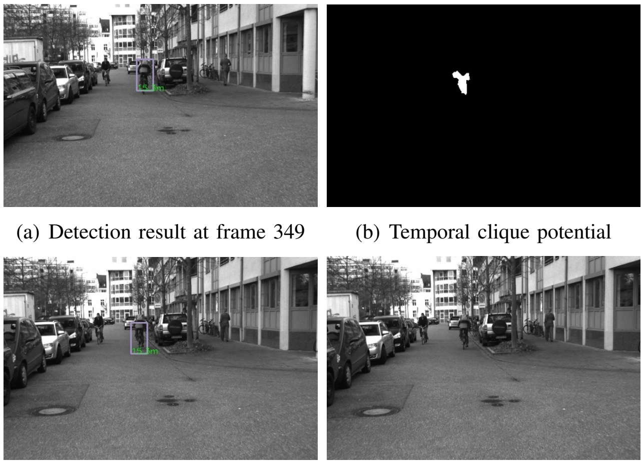
$$D(L_p) = \begin{cases} p(L_p = 0|z) = n_i/(n_i + n_o) \\ p(L_p = 1|z) = n_o/(n_i + n_o) \end{cases}, \tag{9}$$

where n_i and n_o are the number of inliers and outliers, respectively. Fig. 4 shows an example of inliers (blue dots) and outliers (red dots) in superpixels, and many outliers exist in superpixels of image region of the moving car.

Some moving objects which have the forward motion along the camera ray are unexpectedly detected as static objects because reprojection difference caused by forward motion is very small not enough to detect as moving objects. To seamlessly detect these moving objects, the temporal coherence function maintains a previous label when the velocities of two superpixels are similar as follows:

$$D_t(L_p, L_{p'}) = \exp(-\left\|v(L_p) - v(L_{p'})\right\|^2 / 2\sigma_v^2), \tag{10}$$

where $v(L_p)$ is the 3D mean velocity value of features within the superpixel p and σ_v is a 3D covariance of the velocity between two superpixels. Fig. 5 shows comparison results with and without the temporal coherence function.



(a) Detection result at frame 349

(b) Temporal clique potential

(c) Detection result at frame 353 with temporal coherence function

(d) Detection result at frame 353 without temporal coherence function

Fig. 5. Results of moving object detection with and without the temporal coherence function. In (a), the rectangle in the image represents a detected rider at frame 349. In (b), the temporal clique potential is represented and white superpixels represent high value. (c) and (d) show that the rider is continuously detected using the temporal coherence function.

The spatial coherence function between adjacent nodes is computed as follows:

$$V_s(L_p, L_q) = \exp(-\left\|c(L_p) - c(L_q)\right\|^2 / 2\sigma_c^2), \tag{11}$$

where $c(L_p)$ is the mean color value of pixels in superpixel p, and σ_c is a color covariance between two superpixels.

B. Data-association

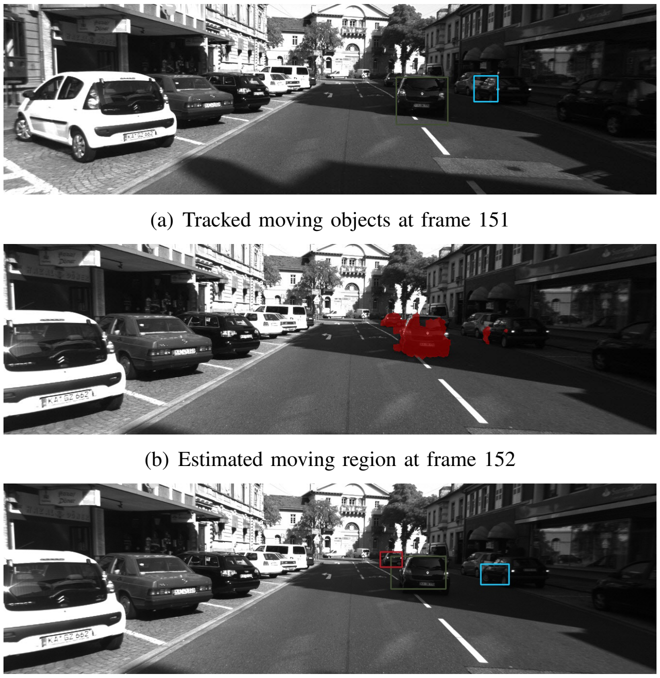
For data association of moving objects, we simply match consecutive sets of superpixels using the majority of quad-matches like superpixel matching. However, sometimes the two moving objects close to each other cannot be separated by detecting a set of moving superpixels, as shown in Fig. 6(b).

To split different moving objects in a set of superpixels, we applied a simple agglomerative clustering method [14] in which superpixels with similar depth (in our case, within ten meters) are merged as shown in Fig. 6(c).

IV. EXPERIMENTAL RESULTS

We have tested the approach with both a synthetic stereo dataset [15] and real stereo datasets [16]. All images are rectified, and the intrinsic and extrinsic parameters of cameras are known. The fundamental drawback of a limited-range 3D sensing is the impossibility to consider moving objects beyond the 3D observability region bounds. Therefore, we limit our scope to those objects situated within these bounds.

The proposed method was fully implemented in C++ and tested on a single PC with a 2.8GHz Quad-Core CPU. Processing time depends on the number of particles, superpixels and quadmatches. For example, processing time for one frame (1344×391 pixels) with 100 particles is about three seconds.

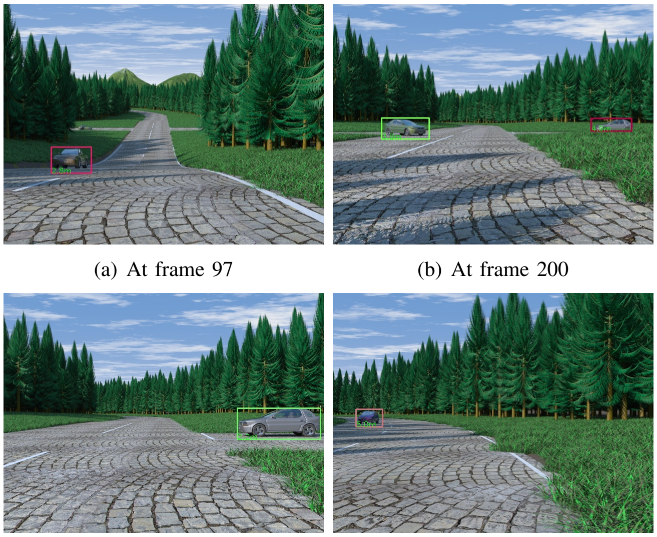


(a) Tracked moving objects at frame 151

(b) Estimated moving region at frame 152

(c) Data association result at frame 152

Fig. 6. Data association results. Two moving objects are tracked at frame 151 as shown in (a). In the next frame, big segments are detected and divided into two objects as shown in (b) and (c).



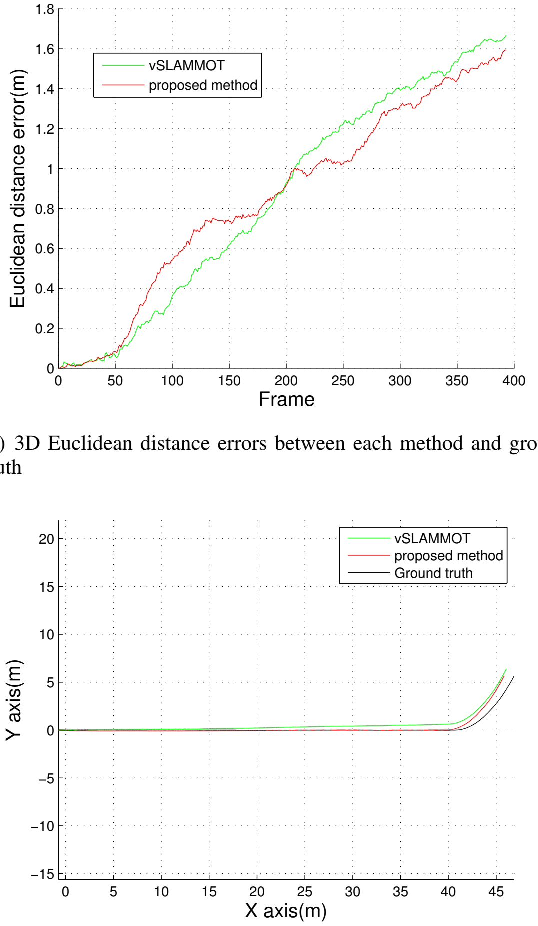
(a) At frame 97

(b) At frame 200

(c) At frame 230

(d) At frame 356

Fig. 7. Some detection results obtained from the synthetic dataset. Different colors of rectangles represent different objects.



(a) 3D Euclidean distance errors between each method and ground truth

(b) Localization results (Top-view) of all methods

Fig. 8. Localization results on synthetic dataset.

TABLE I

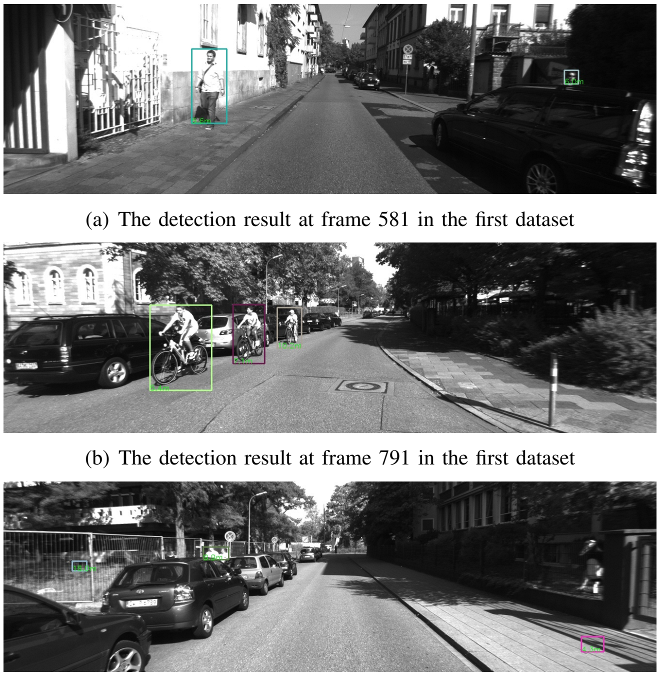
COMPARISON RESULTS OF MOVING OBJECT DETECTION FOR A SYNTHETIC DATASET

Method	true positive	false positive
Ground truth	110	0
vSLAMMOT	**110**	44
proposed method	**110**	**40**

A. Performance Comparison : Synthetic Dataset

We compared our proposed method (vSLAMMOT) with ground truth and vSLAMMOT using visual odometry without considering moving objects. Fig. 7 shows some detection results of a synthetic sequence and Fig. 8 shows localization results of two methods. In this experiment, classifying moving features does not significantly improve the localization performance because the regions of moving objects are small in the image sequence. Thus all methods show the similar localization result.

We compared the detection performance of the proposed method and vSLAMMOT which uses visual odometry but does not consider moving objects by comparing with ground truth. Due to the limited detection-range of the stereo camera, we count moving objects within the pre-defined range (in this dataset, six meters) to make ground truth data.



(a) The detection result at frame 581 in the first dataset

(b) The detection result at frame 791 in the first dataset

(c) The detection result at frame 921 in the first dataset

Fig. 9. Detection results on the first dataset. In (a) and (b), colored rectangles represent detected moving objects. In (c), some false positives exist due to the false quadmatches.

TABLE II

COMPARISON RESULTS OF MOVING OBJECT DETECTION FOR THE FIRST REAL DATASET

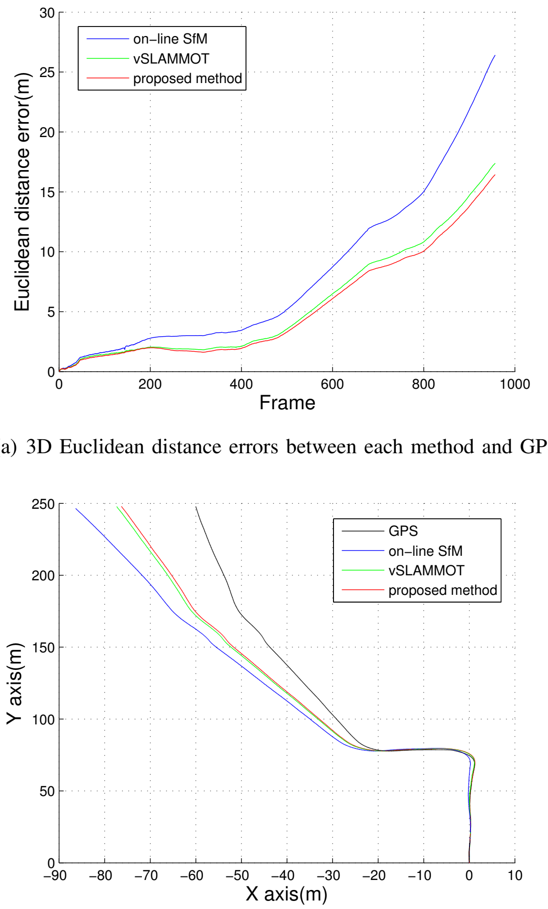
Method	true positive	false positive
Ground truth	1192	0
vSLAMMOT	1130	112
proposed method	**1134**	**84**

In this study, we define the true positive when the overlap region between the detected moving object and the true object is bigger than fifty percentage of the region for the true object. Table. I shows the comparison results of detecting moving objects. Since localization results of all methods are similar, the detection results are also similar. False positive occurs when false quadmatches exist.

B. Performance Comparison : Real Dataset

We compared the performance of the proposed method with those of the on-line structure-from-motion (SfM) method [1] and vSLAMMOT using odometry without considering moving objects.

Using the first real dataset, detection and localization results of the proposed method are shown in Figs. 9 and 10, respectively. As shown in Fig. 10(a), the proposed method considering moving objects shows the better localization performance. Based on more accurate localization, the proposed method also shows the better performance of moving



(a) 3D Euclidean distance errors between each method and GPS

(b) Localization results (Top-view) of all methods

Fig. 10. Localization results on the first dataset.

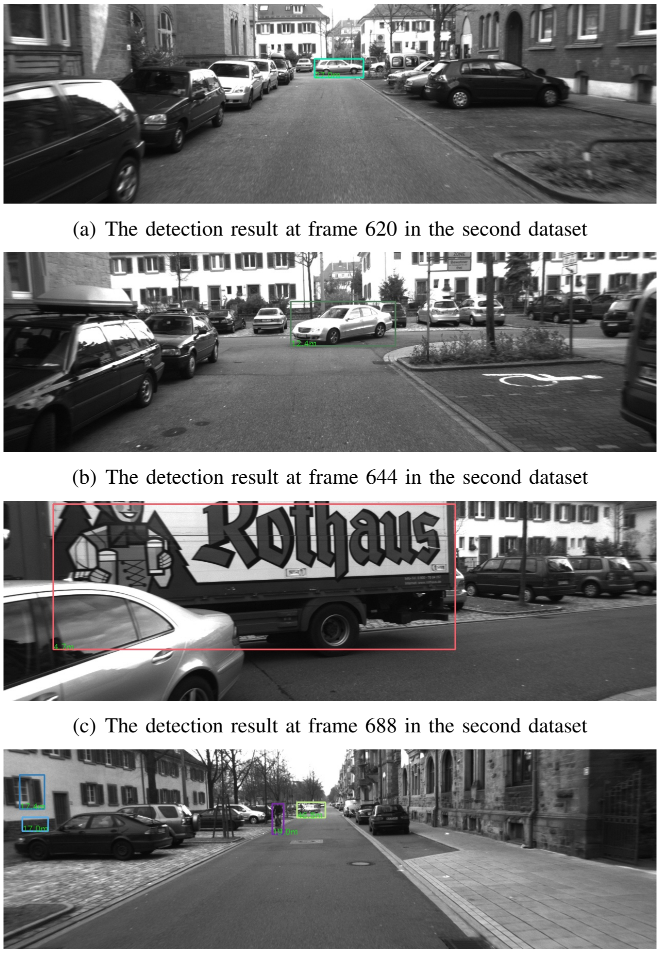
TABLE III

COMPARISON RESULTS OF MOVING OBJECT DETECTION FOR THE SECOND REAL DATASET

Method	true positive	false positive
Ground truth	116	0
vSLAMMOT	109	72
proposed method	**116**	**46**

objects detection as shown in Table. II. In contrast to the synthetic dataset, real images contain motion blur and severe measurement noise, which can yield more false matches and the distances of the objects are inaccurately estimated. For these reasons, static objects are wrongly classified as moving objects.

From frame 650 to frame 700 in the second dataset, a moving object gradually appears and occupies the greater part of the image as shown in Fig. 11. The proposed method sequentially detects the moving objects and immediately classifies features in the moving objects as moving features. Thus we can improve the localization performance by estimating motion only using the static features. Fig. 12 and Table. III show the localization and detection results of the proposed method for the second real dataset, respectively.



(a) The detection result at frame 620 in the second dataset

(b) The detection result at frame 644 in the second dataset

(c) The detection result at frame 688 in the second dataset

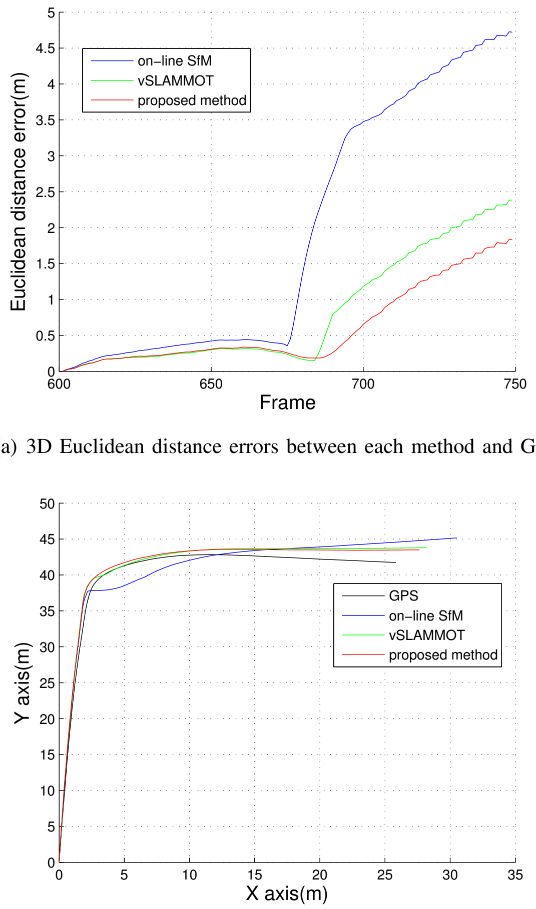
(d) The detection result at frame 746 in the second dataset

Fig. 11. Detection results on the second dataset. The rectangle in the center of each image represents a detected moving car in (a) and (b). A truck that occupies a big region in an image is detected in (c). In (d), some false positives exist due to the false quadmatches.

V. SUMMARY AND FUTURE WORK

In this paper, we have presented a probabilistic solution to stereo-based vSLAMMOT with a sequential Bayesian filtering framework. To achieve the goal, we estimate the hybrid camera path consisting of the camera path and labels of moving objects. The samples for the camera path are generated by using the data-driven proposal distribution. Moving objects are detected by using the proposed MRF energy function. The proposed superpixel-wise MRF framework ensures the spatio-temporal consistency of the detection results through the image sequence. We demonstrated that the proposed method outperforms other vSLAMMOT approaches in terms of the localization accuracy and the performance on moving object detection.

In future work, we hope to address two issues. First, we plan to use region descriptors of objects in images to increase the robustness of tracking because tracking an object is apt to be fail when there is no quadmatches in superpixles belong to the object. Second, we would like to apply parallel processing methods to reduce the computational complexity.



(a) 3D Euclidean distance errors between each method and GPS

(b) Localization results (Top-view) of all methods

Fig. 12. Localization results in the second dataset.

VI. ACKNOWLEDGMENTS

We would like to express our sincere gratitude to the reviewers of this paper for their valuable comments and recommendations. This work was partially supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2010-0028680).

REFERENCES

[1] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germany, June 2011.

[2] J. S. Ortega, "Towards visual localization, mapping and moving objects tracking by a mobile robot: a geometric and probabilistic approach," Feb. 02 2007.

[3] K.-H. Lin and C.-C. Wang, "Stereo-based simultaneous localization, mapping and moving object tracking," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 3975–3980.

[4] D. Migliore, R. Rigamonti, D. Marzorati, M. Matteucci, and D. Sorrenti, "Use a single camera for simultaneous localization and mapping with mobile object tracking in dynamic environments," in *ICRA Workshop on Safe navigation in open and dynamic environments: Application to autonomous vehicles*, 2009.

[5] B. P. Williams and I. Reid, "On combining visual SLAM and visual odometry," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 3494–3500.

[6] P. F. Alcantarilla, L. M. Bergasa, and F. Dellaert, "Visual odometry priors for robust EKF-SLAM," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 3501–3506.

[7] S. Wangsiripitak and D. W. Murray, "Avoiding moving outliers in visual SLAM by tracking moving objects," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2009, pp. 375–380.

[8] A. Ess, B. Leibe, K. Schindler, and L. J. V. Gool, "Robust multiperson tracking from a mobile platform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1831–1846, Oct. 2009.

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

[10] A. Doucet, N. de Freitas, K. Murphy, and S. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI-00)*, 2000, pp. 176–183.

[11] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels," *EPFL Technical Report no.*, no. 149300, 2010.

[12] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *American Control Conference, 1995. Proceedings of the*, vol. 3. IEEE, 1995, pp. 1628–1632.

[13] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge Univ Press, 2000, vol. 2.

[14] T. Hsatie, R. Tibshirani, and J. J. H. Friedman, *The elements of statistical learning*. Springer New York, 2001, vol. 1.

[15] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn, "Differences between stereo and motion behavior on synthetic and real-world stereo sequences," in *23rd International Conference of Image and Vision Computing New Zealand (IVCNZ '08)*, 2008, pp. 1–6.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.