Robust 3-D Visual SLAM in a Large-scale Environment

Jungho Kim, Kuk-Jin Yoon and In So Kweon

Abstract Motion estimation approaches enable the robust prediction of successive camera poses when a camera undergoes erratic motion. It is especially difficult to make robust predictions under such conditions when using a constant-velocity model. However, motion estimation itself inevitably involves pose errors that result in the production of an inconsistent map. To solve this problem, we propose a novel 3D visual SLAM approach in which both motion estimation and stochastic filtering are performed; in the proposed method, visual odometry and Rao-blackwellized particle filtering are combined. First, to ensure that the process and the measurement noise are independent (they are actually dependent in the case of a single sensor), we simply divide observations (i.e., image features) into two categories, common features observed in the consecutive key-frame images and new features detected in the current key-frame image. In addition, we propose a key-frame SLAM to reduce error accumulation with a data-driven proposal distribution. We demonstrate the accuracy of the proposed method in terms of the consistency of the global map.

1 Introduction

Simultaneous localization and mapping (SLAM) is the technique of building up a map of the unknown environment while simultaneously keeping track of the current position of the cameras or robots in the environment. This problem has attracted immense attention in the robotics and computer vision communities. To solve the SLAM problem, many methods based on the recursive estimation of the posterior distribution have been introduced. Davison [1] successfully performed monocular

Jungho Kim and In So Kweon
KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon, Korea, e-mail: jhkim@rcv.kaist.ac.kr, iskweon@kaist.ac.kr

Kuk-Jin Yoon
GIST, 261 Cheomdan-gwagiro, Buk-gu, Gwangju, Korea e-mail: kjyoon@gist.ac.kr

SLAM by employing an extended Kalman filter (EKF) and adopting an initialization process to determine the depth of the 3D landmarks by using the particle filter-type approach. Eade et al. [2] utilized the FastSLAM-type particle filter in single-camera SLAM to manage a greater number of landmarks because the computational requirements of EKF-based SLAM approaches rapidly grow with the number of landmarks. In [3], the authors described a visual SLAM algorithm that is robust to erratic camera motion and visual occlusion by using efficient scale prediction and examplar-based feature representations in conjunction with the use of an unscented Kalman filter (UFK). Recently, Eade et al. [4] proposed a monocular SLAM system in which map inconsistencies can be prevented by coalescing observations into independent local coordinate frames, building a graph of the local frames, and optimizing the resulting graph. Paz et al. [5] presented a 6-degree-of-freedom (DOF) visual SLAM system based on conditionally independent local maps by using a stereo camera as the only sensor. Here, it is worth noting that in most 3D visual SLAM approaches, a constant-velocity model is employed to achieve the independence between the process and the measurement noise. However, in the constant velocity model, when cameras undergo sudden motion, these SLAM approaches are highly prone to failure, resulting in inconsistencies in the global map. In [6], the authors combined the particle filter-based localization with the UKF-based SLAM problem to cope with erratic camera motion while maintaining a small number of landmarks.

On the other hand, in the vision community, structure-from-motion (SFM) approaches have been studied independent of SLAM to estimate camera trajectories by using only a sequence of images. For example, Nister et al. introduced 'visual odometry' that estimates the relative movements of the stereo head in the Euclidean space [7]. Recently, Zhu et al. [8] developed a helmet-based visual odometry system that consists of two pairs of stereo-cameras mounted on a helmet; one pair faces forward while the other faces backward. By utilizing the multi-stereo fusion algorithm, they improved the overall accuracy in pose estimation. Here, we should note that in many previous studies, optimization techniques such as bundle adjustment [9, 10] have been adopted to avoid inconsistencies in the global map. However, it is not feasible to perform conventional bundle adjustment in on-line approaches because the computational cost rapidly grows with the number of 3D landmarks and their observations (the image coordinates over the sequence).

2 Motivation

SLAM approaches are described by two probabilistic models — the process and measurement models [11]. The current state, x_t, is governed by the probabilistic function of the previous state x_{t-1}, the control input u_t, and the additive process noise w_t as follows :

$$x_t = f\left(x_{t-1}, u_t, w_t\right). \tag{1}$$

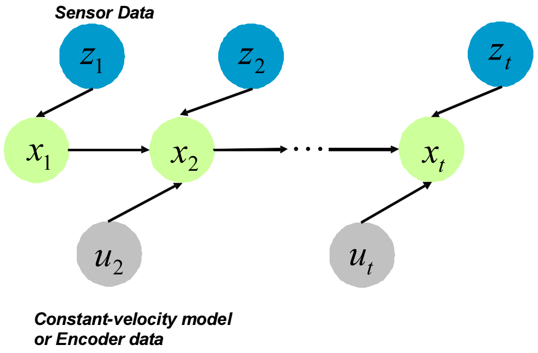


Fig. 1 Bayesian network that describes the relations between process and measurement models

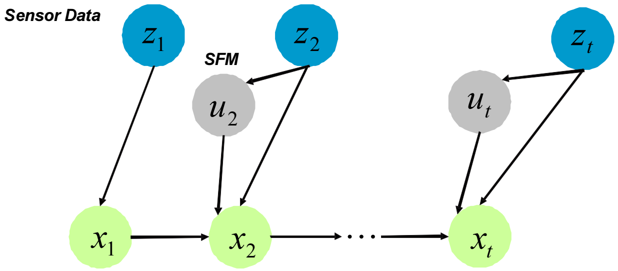


Fig. 2 Bayesian network that describes dependency between process and measurement noise

The measurements z_t are also governed by the probabilistic function of the current state and the additive measurement noise v_t. We re-estimate the posterior distribution of the state produced from the process model by using the measurement model as follows :

$$z_t = h(x_t, v_t). \tag{2}$$

where w_t and v_t represent the process and measurement noise, respectively; they are assumed to be independent, as shown in Fig. 1.

In many visual SLAM approaches, a constant-velocity model that is independent of sensor data is employed [1, 2, 3] or another sensor such as a wheel encoder [12], or IMU [13] is used for the process model. However, if a camera undergoes sudden motion, a constant-velocity model is not valid, and wheel encoders cannot be used to obtain the 6-DOF poses of the camera. In contrast, motion estimation approaches can be adopted to obtain good estimates for the 6-DOF pose under erratic camera motion.

However, on the other hand, if we use motion estimation methods for the process model, the control input, u_t, directly depends on measurements, and the independence assumption is no longer valid, as shown in Fig. 2. Moreover, SLAM approaches involve some problems related to dimensionality when estimating the

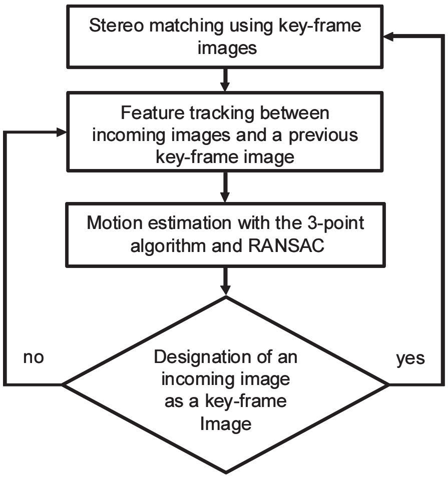

Stereo matching using key-frame images
Feature tracking between incoming images and a previous key-frame image
Motion estimation with the 3-point algorithm and RANSAC
Designation of an incoming image as a key-frame Image
no
yes

Fig. 3 Overall procedure followed in our visual odometry system

6-DOF camera pose and 3D landmarks, and scalability caused by the large number of landmarks obtained during long-distance navigation.

The contributions of the proposed approach compared to the previous methods can be summarized as follows:

(i) We achieve the independence between the process and measurement noise when using conventional motion estimation approaches for the process model in SLAM. (Section 4.2).

(ii) We use the key-frame SLAM approach to reduce the number of camera poses to be estimated in the path by generating the poses only at key-frame locations. In addition, we propose a method that effectively updates the posterior distribution of the camera path by using many observations obtained at non key-frame locations. (Section 4.3).

(iii) We use a data-driven proposal distribution computed using the RANSAC to efficiently represent the posterior distribution of the camera pose by a limited number of particles. (Section 4.5)

(iv) We develop a novel SLAM approach by integrating the above contributions to reduce the error accumulation involved in conventional motion estimation approaches.

3 The Visual Odometry System

Our visual odometry system consists of a few sub-components. Fig. 3 shows the overall procedure followed in our visual odometry system.

3.1 Feature extraction and stereo matching

For each stereo pair, we first extract corner points in the left image and then apply the 1D KLT feature tracker [14] to stereo images to obtain correspondences. The 3D coordinates of the matched corner points are used for map building and for motion estimation.

3.2 Motion Estimation

As mentioned previously, Nister et al. [7] introduced a method called 'visual odometry' for the real-time estimation of the movement of a stereo head or a single camera. In this approach, the 3-point algorithm [15] is employed: the images of three known world points are used to obtain up to four possible camera poses and more than three points are required to automatically obtain one solution. Here, we also employ the RANSAC [16] where a set of 3 world points and their image coordinates are randomly selected to compute the relative camera pose. The estimated pose is evaluated from other correspondences.

3.3 Key-frame Designation

The number of tracked corners between an incoming image and a previous key image is a measure that is used to determine the key-frame locations. If the number of points tracked by the KLT tracker [17] is smaller than a pre-defined threshold, we designate an incoming image as a key-frame image and estimate the relative pose w.r.t the previous key-frame. This strategy can partially prevent error accumulation because we determine the relative pose w.r.t the previous key-frame pose.

4 Stochastic Filtering

4.1 Rao-Blackwellized Particle Filter

We initially employ a Rao-Blackwellized particle filtering technique [18, 19], by which a SLAM problem is decomposed into a localization problem and a collection of landmark estimation problems that are conditioned on pose estimates as follows:

$$\begin{aligned} p\left(x_{1:t}, M | z_{1:t}, d_{1:t}\right) &= \frac{p\left(x_{1:t}, M, z_{1:t}, d_{1:t}\right)}{p\left(z_{1:t}, d_{1:t}\right)} \\ &= p\left(x_{1:t} | z_{1:t}, d_{1:t}\right) p\left(M | x_{1:t}, z_{1:t}, d_{1:t}\right). \end{aligned} \tag{3}$$

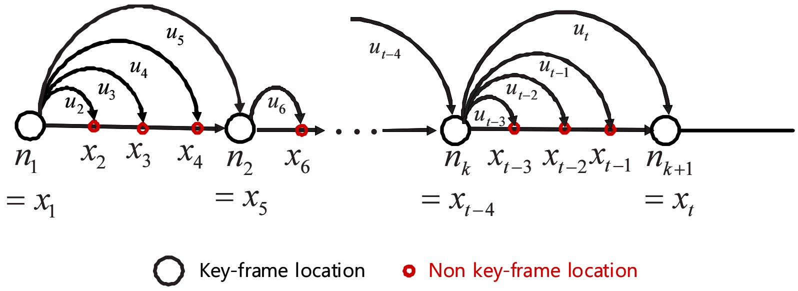


Fig. 4 A sequence of frames divided according to key-frame and non key-frame locations

where $x_{1:t}$ and M represent the camera path and a collection of 3D landmarks, respectively. $z_{1:t}$ and $d_{1:t}$ indicate the observations and data association up to t.

In FastSLAM [19], the path estimator is implemented using the particle filter. The landmark estimator is implemented using a Kalman filter, with a separate filter for different landmarks, because all the 3D landmarks are independent of each other with a given path as shown in Eq. (4).

$$p\left(M|x_{1:t}, z_{1:t}, d_{1:t}\right) = \prod_{l=1}^{L} p\left(m_l|x_{1:t}, z_{1:t}, d_{1:t}\right) \tag{4}$$

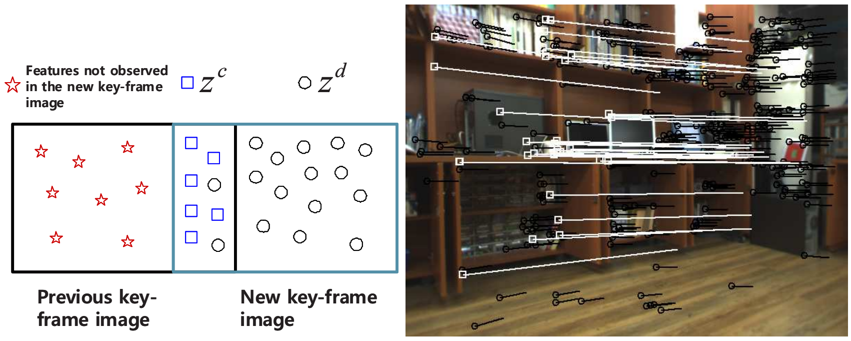
where m_l represents each landmark in M.

4.2 Independence Between Process and Measurement Noise

We estimate the camera path that consists of a sequence of camera poses at only the key-frame locations instead of all the frames, as mentioned in Section 3.3. In other words, we estimate the posterior distribution $p\left(n_{1:k}|z_{1:t}, d_{1:t}\right)$ instead of $p\left(x_{1:t}|z_{1:t}, d_{1:t}\right)$. When designating an incoming image as a key-frame image, we elongate the path, $n_{1:k+1}$, by adding the relative pose, u_t, w.r.t the last key-frame pose, n_k, to the previous path, $n_{1:k}$, as shown in Fig. 4. We first divide the observations into two categories: observed features common to two key-frame images z^c and newly detected features in the current key-frame image z^d $(= z - z^c)$, as shown in Fig. 5. z^c are used to evaluate the posterior distribution of the path $n_{1:k}$, and z^d are used to estimate the relative pose, u_t. Thus, we have

$$p\left(n_{1:k}|z_{1:t}, d_{1:t}\right) = p\left(n_{1:k}|z^c_{1:t}, z^d_{1:t}, d_{1:t}\right) \tag{5}$$

We then achieve the independence between the process and measurement noise by simply dividing the observations instead of using another sensor, as shown in Fig. 6.



(a) Consecutive key-frame images

(b) Black circles and white rectangles represent features belonging to z_t^d and z_t^c, respectively

Fig. 5 Achieving independence between process and measurement noise by dividing observations according to purpose

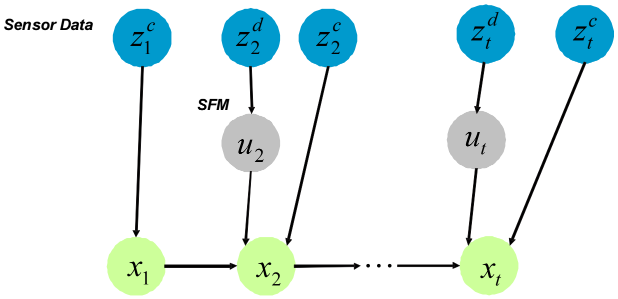


Fig. 6 Bayesian network that describes the independence between process and measurement noise in the case of divided observations

4.3 Path Estimation for Key-Frame SLAM

The posterior distribution of $n_{1:k}$ with given $z_{1:t}$ and $d_{1:t}$ (correspondences between the landmarks, M, and observations, $z_{1:t}^c$) is represented as a weighted set of particles:

$$p\left(n_{1:k}|z_{1:t}^c, z_{1:t}^d, d_{1:t}\right) = \sum_i p\left(n_{1:k}^i|z_{1:t}^c, z_{1:t}^d, d_{1:t}\right)\delta\left(n_{1:k} - n_{1:k}^i\right) \tag{6}$$

where $\delta(x)$ represents the Dirac delta function that returns 1 if x is zero, and 0 otherwise.

For non key-frame locations, we re-estimate the posterior distribution of the camera path $n_{1:k}$ by marginalizing the relative pose u_t using Eq. (7).

$$\begin{aligned} p\left(n^i_{1:k}|z^c_{1:t}, z^d_{1:t}, d_{1:t}\right) &= \int p\left(n^i_{1:k}, u_t|z^c_{1:t}, z^d_{1:t}, d_{1:t}\right) du_t \\ &= \sum_j p\left(n^i_{1:k}, u^j_t|z^c_{1:t}, z^d_{1:t}, d_{1:t}\right) \end{aligned} \tag{7}$$

The posterior distribution of the relative pose is represented by a set of particles coming from the RANSAC, where relative poses are estimated by selecting multiple sets of minimal correspondences. Each pair of the minimal set provides a single hypothesis on the relative pose, and its weight is computed according to the number of inliers among all correspondences. This will be described in Section 4.6. We compute the joint probability of a camera path, $n^i_{1:k}$, and a relative pose, u^j_t, using Eq. (8).

$$\begin{aligned} p\left(n^i_{1:k}, u^j_t|z^c_{1:t}, z^d_{1:t}, d_{1:t}\right) &= \frac{p\left(n^i_{1:k}, u^j_t, z^c_t, z^d_t, d_t, z^c_{1:t-1}, z^d_{1:t-1}, d_{1:t-1}\right)}{p\left(z_{1:t-1}, d_{1:t-1}, z_t, d_t\right)} \\ &= \eta\, p\left(z^c_t|n^i_{1:k}, u^j_t, d_t\right) p\left(u^j_t|z^d_t\right) p\left(n^i_{1:k}|z^c_{1:t-1}, z^d_{1:t-1}, d_{1:t-1}\right) \end{aligned} \tag{8}$$

where $p(z^c_t|n^i_{1:k}, u^j_t, d_t)$ is a likelihood and $p(u^j_t|z^d_t)$ is the posterior distribution of the relative pose, as defined by Eq. (11). $p(n^i_{1:k}|z_{1:t-1}, d_{1:t-1})$ is the previous posterior distribution up to $t-1$, and η is a normalization term that makes the sum of all probabilities 1.

In our key-frame SLAM approach, we can reduce the number of camera poses to be estimated in the path and update the particles of the camera path whose poses are generated at the key-frame locations by using many observations obtained at the non key-frame locations.

4.4 Likelihood Estimation

The likelihood estimation is based on the number of inliers for each particle of the camera pose. It is computed by examining how many scene points m_l are projected close to relevant measurements z^l_t, as defined in Eq. (9).

$$\begin{aligned} p\left(z^c_t|n^i_{1:k}, u^j_t, d_t\right) &= \sum_{l=1}^{L} d\left(z^l_t, m_l, n^i_{1:k}, u^j_t\right)/L \\ d\left(z^l_t, m_l, n^i_{1:k}, u^j_t\right) &= \begin{cases} 1 & \text{if } \left\|z^l_t - z\left(m_l, \left(n^i_{1:k} \oplus u^j_t\right)\right)\right\| < e_l \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

where $(n^i_{1:k} \oplus u^j_t)$ indicates the global pose of the camera computed from the path $n^i_{1:k}$ and the relative pose u^j_t. $d(z^l_t, m_l, n^i_{1:k}, u^j_t)$ indicates whether the point m_l is an inlier or outlier with respect to the observation z^l_t and the camera pose $(n^i_{1:k} \oplus u^j_t)$, and $z(m_l, (n^i_{1:k} \oplus u^j_t))$ is the projection of a scene point m_l for a particular camera pose $(n^i_{1:k} \oplus u^j_t)$. L is the number of scene points that are associated with the current measurements, as defined by d_t, and e_l represents the uncertainty in the projection of the 3D scene point (see Section 4.8).

4.5 Outlier Rejection

We eliminate the outliers z^o_t among z^l_t that are not supported by any particles in the computation of the likelihood values as shown in Eq. (10).

$$z^o_t = \left\{ z^l_t | \sum_i \sum_j d\left(z^l_t, m_l, n^i_{1:k}, u^j_t\right) = 0 \right\} \tag{10}$$

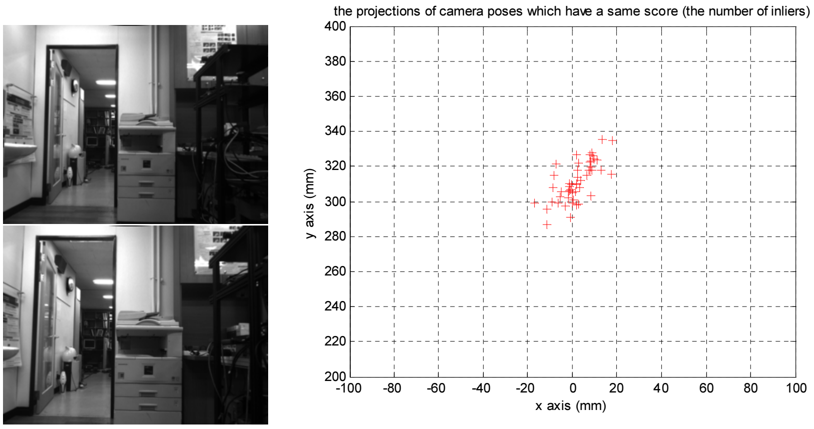
Thus, we eliminate the outliers in z^d_t using the RANSAC when estimating the relative pose, u_t, and the outliers in z^c_t when computing the likelihood values.

4.6 Data-Driven Proposal Distribution

It is especially insufficient to represent the posterior distribution using the limited number of particles in the 6-dimensional space. In our approach, we use multiple hypotheses that are generated in the RANSAC step. The RANSAC is an efficient technique for determining a good hypothesis, but unfortunately the hypothesis selected with the best score (the number of inliers) does not always correspond to the correct estimate. Therefore, in our approach, instead of selecting an unique hypothesis, we propagate multiple reasonable hypotheses to the subsequent frames to re-estimate the posterior distribution by using more observations. Fig. 7 shows the projections of 3D camera poses that have the best score, i.e., the maximum number of inliers computed using the RANSAC. We represent the posterior distribution of the relative pose using these hypotheses and their weights according to the number of inliers, as shown in Eq. (11).

$$p\left(u^j_t | z^d_t\right) \propto \frac{N^j_{inlier}}{N_{total}}, \quad \sum_j p\left(u^j_t | z^d_t\right) = 1 \tag{11}$$

where N^j_{inlier} is the number of inliers for u^j_t, and N_{total} is the total number of correspondences in z^d_t. This means that the multiple hypotheses on the camera pose generated by the RANSAC are probabilistically evaluated by using more incoming observations than just two views.



(a) Two consecutive images (b) The top-down view of the 3D camera poses which have the same score (the number of inliers)

Fig. 7 Hypotheses of the camera pose that have the same score when using the RANSAC

4.7 Path Generation

Whenever we have a new key-frame image, we elongate the path using the previous posterior distribution of the camera trajectory and the relative pose as follows:

$$n_{1:k+1}^{N_j \times i+j} \leftarrow \left\{ n_{1:k}^{i}, \left(n_{1:k}^{i} \oplus u_t^j \right) \right\},$$
$$p\left(n_{1:k+1}^{N_j \times i+j} | z_{1:t}, d_{1:t} \right) \propto p\left(u_t^j | z_t^d \right) p\left(n_{1:k}^{i} | z_{1:t}, d_{1:t} \right), \tag{12}$$

where N_j is the number of particles for the relative pose. Here, before adding the relative pose to the particles of the camera path, we prune some hypotheses on the camera path on the basis of their weights. In our implementation, only the 10 best particles remain.

4.8 Landmark Estimation

When designating an incoming image as a key-frame image, we update the posterior distributions of the landmarks. We model the posterior distribution of each landmark $p(m_l | n_{1:k}, z_{1:k}^l, d_{1:k})$ defined in Eq. (4) using a optimized 3D landmark location, $\hat{m}_l$, and its uncertainty ,e_l, in the image space; we re-triangulate 3 observations (first two stereo views and the last view) corresponding to each landmark for each particle of the camera path by using SVD [9] to compute $\hat{m}_l$, as shown in Fig. 8, and e_l is determined by the projection error of $\hat{m}_l$ for the pose of the last key-frame image n_N, as shown in Eq. (13).

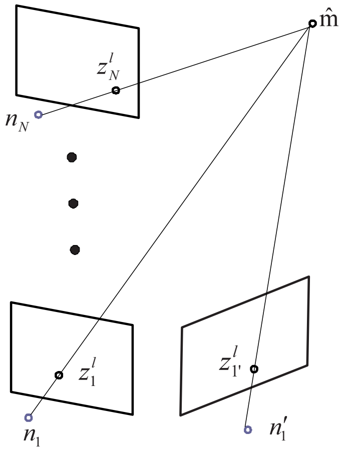


Fig. 8 Landmark update by re-triangulating observations with a given camera path

$$e_l = \left\| z_N^l - z\left(\hat{m}_l, n_N\right) \right\| + e_0 \tag{13}$$

where N is the number of observations at the key-frame locations for each landmark, and e_0 is a pre-defined initial projection uncertainty of the landmark.

5 Experimental Results

For experiments, we used a stereo camera with a 12*cm* baseline and a 6*mm* lens, which provide a narrow field of view. The resolution of the images is 320×240 pixels. Fig. 9 shows the path obtained by using visual odometry (red line) and the hypotheses on the trajectory computed by using the proposed SLAM method (blue lines). For this experiment, we captured 552 images by driving a robot over a distance of 10*m* in an indoor environment and to evaluate the performance, we added the first image to the end of the image sequence so that the initial and final locations are identical. At the final location, we choose the path $n_{1:k}^m$ and the relative pose u_t^n that maximize the posterior distributions as follows:

$$\begin{aligned} m &= \operatorname*{argmax}_i p\left(n_{1:k}^i | z_{1:t}, d_{1:t}\right), \text{ where } p\left(n_{1:k}^i | z_{1:t}, d_{1:t}\right) = \sum_j p\left(n_{1:k}^i, u_t^j | z_{1:t}, d_{1:t}\right) \\ n &= \operatorname*{argmax}_j p\left(u_t^j | z_{1:t}, d_{1:t}\right), \text{ where } p\left(u_t^j | z_{1:t}, d_{1:t}\right) = \sum_i p\left(n_{1:k}^i, u_t^j | z_{1:t}, d_{1:t}\right) \end{aligned} \tag{14}$$

Table 1 lists the errors of the final camera pose for visual odometry and the proposed method. Fig. 11(a) shows the path computed by visual odometry and the corresponding global map obtained by integrating the structures computed by stereo

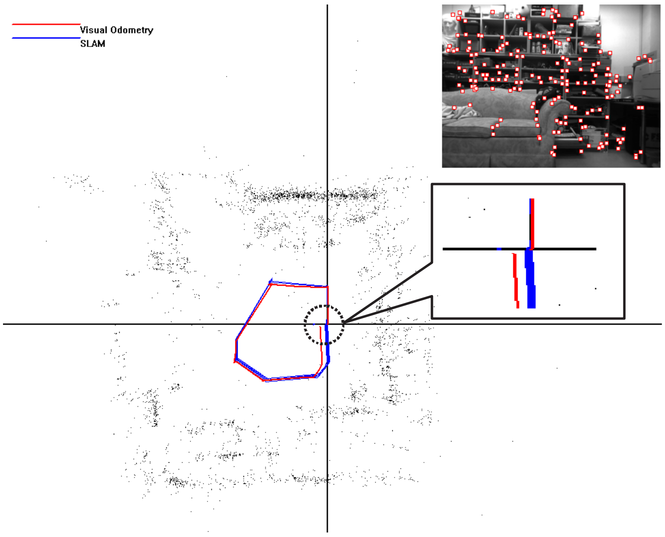


Fig. 9 Camera trajectories estimated by visual odometry only (red line) and by using the proposed method (blue lines)

Table 1 Pose errors for visual odometry and the proposed method (*mm*)

	x	y	z	total
visual odometry only	-182.08	-147.70	-50.50	239.83
proposed	-53.67	5.33	30.57	61.99

matching over time. In this experiment, many images (102 from the 312 images) were influenced by motion blur because we captured images by using a hand-held stereo camera that underwent erratic motion; the images are shown in Fig. 10. In addition, a camera re-visited the scene where it first observed. We can easily find the inconsistency in the global map caused by error accumulation of the path. However, the map produced by using the proposed method is more consistent than that obtained by visual odometry, as shown in Fig. 11(b). Here, we randomly selected 500 sets of 3 correspondences to obtain 500 hypotheses of which we only chose a maximum of 50 hypotheses on the basis of their weights (the number of inliers).
Fig. 12(a) shows the map computed by visual odometry using the same sequence of images. In this experiment, we generated only 300 hypotheses of which we retained a maximum of 50 hypotheses on the basis of their weights. Because the number of hypotheses is small, this map has a larger error than the previous result. However, we can compensate for error accumulation by using the proposed method, as shown in Fig. 12(b). Moreover, we can observe that the results obtained by using the proposed method are more consistent than those obtained by visual odometry because the proposed method is not strongly affected by randomness. Fig. 13 (a) and (b) show the global maps and the camera paths computed by visual odometry and by using the proposed method, respectively. For this experiment, we captured more

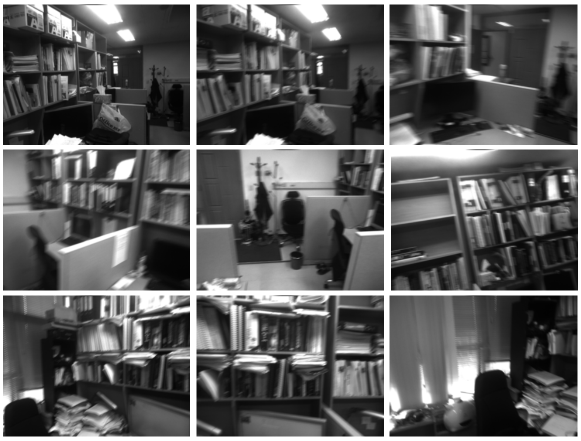
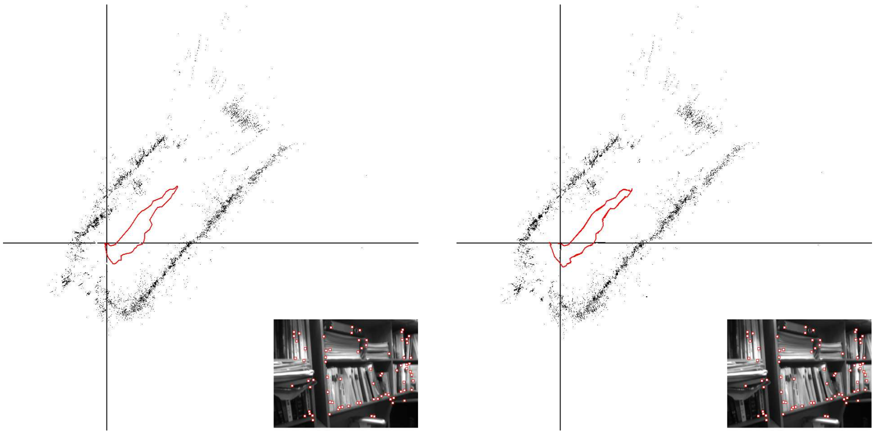


Fig. 10 Some images affected by motion blur

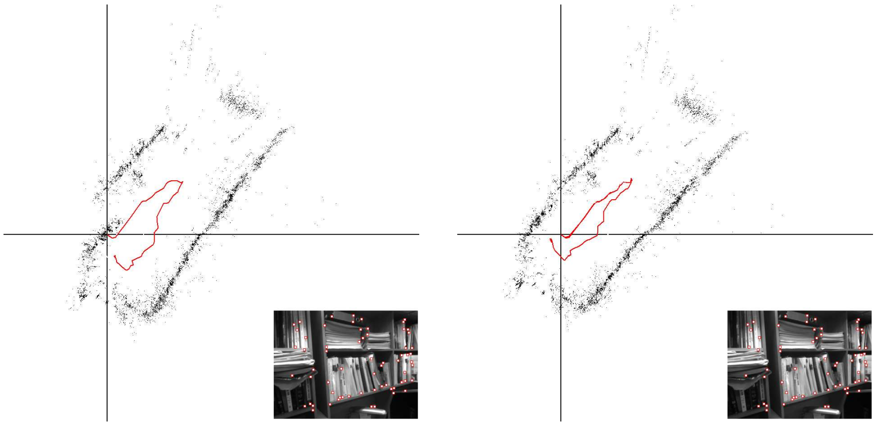


(a) The top-down view of the 3D map and camera trajectory estimated by visual odometry

(b) The top-down view of the 3D map and camera trajectory estimated by the proposed method

Fig. 11 The top-down view of the 3D map and camera trajectories obtained for 500 hypotheses in the RANSAC step

than 3000 images while walking more than 200 *m* in the outdoor environment with the stereo camera in hand. We can see that the results obtained with the proposed visual SLAM approach, in which the visual odometry and stochastic filtering are combined, are much better than those obtained by only visual odometry. To evaluate the consistency, we overlap the maps and paths with the corresponding google map as shown in Fig 13(c) and (d). The proposed SLAM algorithm can process approximately 10 frames per second when using a 2.4 GHz CPU. Table 2 lists the computational complexities for visual odometry and stochastic filtering.



(a) The top-down view of the 3D map and camera trajectory estimated by visual odometry

(b) The top-down view of the 3D map and camera trajectory estimated by the proposed method

Fig. 12 The top-down view of the 3D map and camera trajectories obtained for 300 hypotheses in the RANSAC step

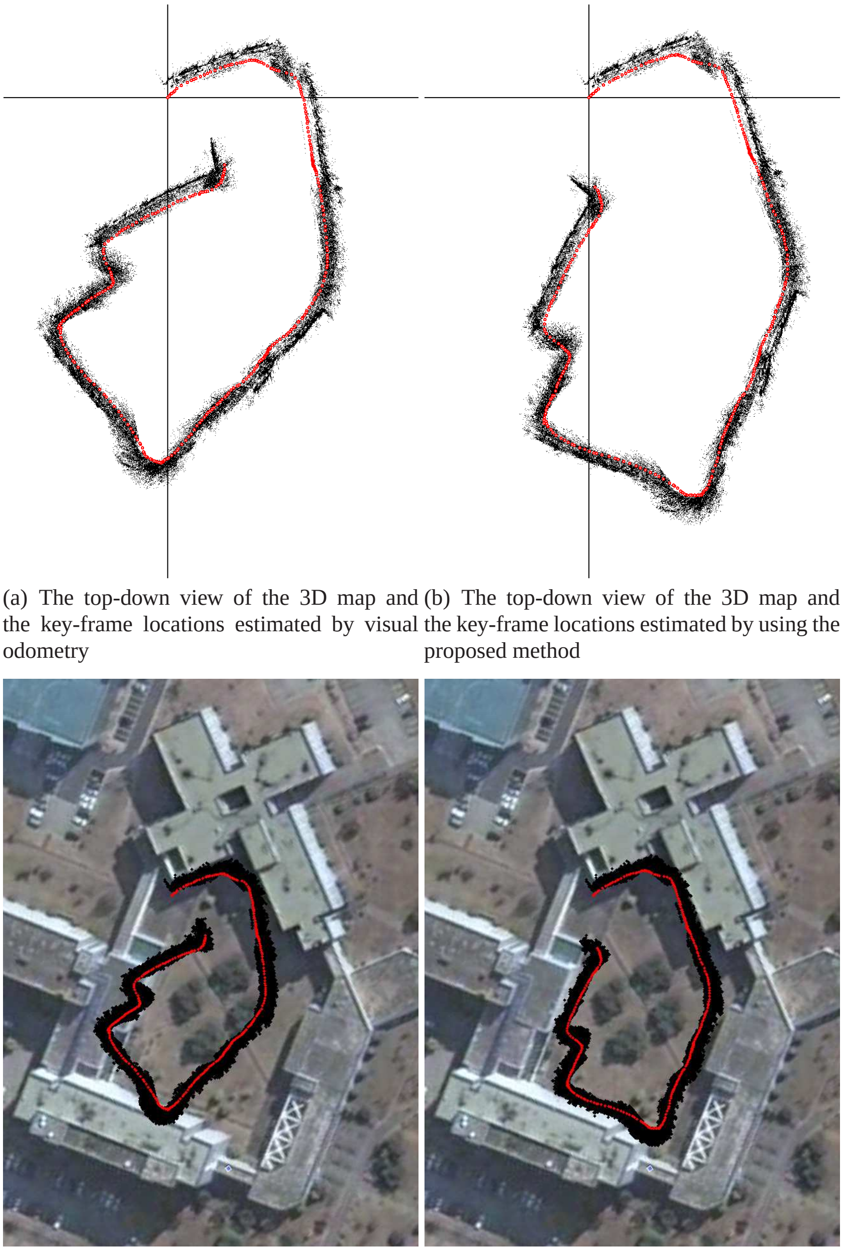
Table 2 Average processing time for visual odometry and proposed stochastic filtering when running 500 frames (ms)

operation	processing time
visual odometry (partially implemented by MMX programming)	27.655
stochastic filtering	75.733
total	103.388

6 Conclusion

We have presented a novel 3D visual SLAM method in which visual odometry and a stochastic filtering approach are combined to cope with sudden camera motion and to obtain consistent maps. To ensure that the process and the measurement noise are independent, we simply divide observations into two categories: common features observed in the consecutive key-frame images and new features detected in the current key-frame image. The proposed stochastic filtering technique can be adopted in existing motion estimation approaches to avoid error accumulation. In addition, our approach is especially efficient in the following sense:

- Dimensionality — we use a data-driven proposal distribution computed by the RANSAC approach with the 3-point algorithm to efficiently represent the posterior distribution of the camera pose.
- Scalability — we reduce the number of possible camera poses in the path by formulating the key-frame SLAM, and our SLAM approach is based on the Rao-



(a) The top-down view of the 3D map and the key-frame locations estimated by visual odometry

(b) The top-down view of the 3D map and the key-frame locations estimated by using the proposed method

(c) The results obtained by visual odometry overlapped with the corresponding google map

(d) The results obtained by using the proposed method overlapped with the corresponding google map

Fig. 13 The global map and the key-frame locations for the outdoor environment

Blackwellized particle filter that can manage more landmarks than the EKF and particle filter-based approaches.

Acknowledgements This work was supported by National Strategic R&D Program for Industrial Technology and a project for Mega City Modeling, Korea.

References

1. A. J. Davison, Real-Time Simultaneous Localisation and Mapping with a Single Camera, ICCV 2003.
2. E. Eade and T. Drummond, Scalable Monocular SLAM, CVPR 2006.
3. D. Chekhlov, M. Pupilli, W. Mayol and A. Calway, Robust Real-Time Visual SLAM Using Scale Prediction and Exemplar Based Feature Description, CVPR 2007.
4. E. Eade and T. Drummond, Monocular SLAM as a Graph of Coalesced Observations, ICCV 2007.
5. L. M. Paz, P. Pinié, J. D. Tardós, J. Neira, Large-Scale 6-DOF SLAM With Stereo-in-Hand, IEEE Trans. on Robotics, vol. 24, no. 5, 946–957, 2008.
6. M. Pupilli and A. Calway, Real-Time Visual SLAM with Resilience to Erratic Motion, CVPR 2006.
7. D. Nister, O. Naroditsky and J. Bergen, Visual Odometry, CVPR 2004.
8. Z. Zhu, T. Oskiper, S. Samarasekera, R. Kumar and H. S. Sawhney, Real-Time Global Localization wtih A Pre-Built Visual Landmarks Database, CVPR 2008.
9. R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000.
10. B. Triggs, P. McLauchlan, R. Hartley, A. Fitzgibbon, Bundle adjustment - a modern synthesis, LNCS (Vision Algorithms: Theory and Practice), 2000.
11. M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte and M. Csorba, A Solution to the Simultaneous Localization and Map Building (SLAM) Problem, IEEE Trans. on Robotics and Automation, vol. 17, no. 3, 229–241, 2001.
12. M. Kaess and F. Dellaert, Visual SLAM with a Multi-Camera Rig, GVU Technical Report, GIT-GVU-06-06, 2006.
13. T. K. Marks, A. Howard, M. Bajracharya, G. W. Cottrell and L. Matthies, Gamma-SLAM: Using Stereo Vision and Variance Grid Maps for SLAM in Unstructured Environments, ICRA 2008.
14. J. Kim, Y. Bok and I. S. Kweon, Robust Vision-based Autonomous Navigation against Environment Changes, IROS 2008.
15. R. Haralick, C. Lee, K. Ottenberg and M. Nölle, Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem, IJCV, vol. 13, no. 3, 331–356, 1994.
16. M. Fischler and R. Bolles, Random Sample Consensus : a Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography, Communications ACM, 1981.
17. J. Shi and C. Tomasi, Good Features to Track, CVPR 1994.
18. K. Murphy and S. Russell, Rao-blackwellized particle filtering for dynamic bayesian networks, Sequential Monte Carlo Methods in Practice, Springer, 2001.
19. M. Montemerlo, W. Whittaker and S. Thrun, FastSLAM: A factored solution to the simultaneous localization and mapping problem, AAAI National Conference on Artificial Intelligence, 2002.