The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Detailed 3D Mapping Based on Image Edge-Point ICP and Recovery from Registration Failure

Masahiro Tomono

Abstract— **This paper presents a method of 3D mapping using a binocular stereo camera with automatic recovery from registration failure. We employ edge points as map element, and apply a variant of ICP algorithm to the inter-frame registration. The method detects failure in the registration due to erratic camera motion or moving objects and recovers from the failure by searching a good image to resume the registration using edge-point SIFT descriptors. In experiments, our method successfully coped with registration failure in 3D mapping in indoor and outdoor environments under various conditions.**

I. Introduction

Vision-based 3D Simultaneous Localization and Mapping (SLAM) has been studied intensively because cameras provide a variety of useful information including geometry, texture, and color and also because cameras are more suitable for realtime 3D mapping than laser scanners. Most of the conventional vision-based SLAM systems utilize corner-like features [4], [6], [12] since data association for corner points is relatively easy due to their distinguishability. However, corner points cannot be detected sufficiently in non-textured environments such as a corridor. In the previous work, we proposed a 3D SLAM scheme with a stereo camera using edge points [14]. An edge point is a point on an edge segment detected in the image. A major advantage of the method is that the estimation process is extremely robust due to plenty of edge points, which can be detected from a few edge lines in non-textured environments. Another important advantage is that edge points provide the detailed shape of the objects in the environment.

A problem in vision-based 3D SLAM is failure in feature tracking and pose estimation due to erratic camera motion, occlusion by moving objects, and illumination changes. These kinds of failure are unavoidable since they are caused by external factors. Thus, it is important to recover automatically from failure. There are some studies in automatic recovery. Chekhlov et. al [3] proposed robust monocular SLAM using multi-resolution descriptors like SIFT. Williams et. al [15] presented automatic relocalization scheme based on a pose consensus approach. These methods build sparse maps using corner points. Our system builds detailed maps of large indoor and outdoor environments using edge points, and we need another automatic recovery scheme.

This paper presents a method of 3D mapping using a binocular stereo camera with automatic recovery from registration failure. Our method detects failure in registration due to erratic camera motion or moving objects and recovers from the failure by searching an image similar to the image just before the failure by manipulating the camera or waiting for the moving object to pass by. For this purpose, we define a score for failure detection, which is based on the ratio of matched edge points and all the edge points. Also, we define two scores for failure recovery, which are based on SIFT descriptors and similarity transformation between two images. The system enters the recovery mode when detecting a failure, and resumes localization and mapping when finding a good image for recovery.

M. Tomono is with Future Robotics Technology Center, Chiba Institute of Technology, Narashino, Chiba 275-0016, Japan. tomono@furo.org

II. 3D Mapping Using Image Edge Points

This section briefly reviews the stereo SLAM presented in the previous work [14]. We refer to a pair of left and right images as *stereo frame* (*frame*, for short).

A. Intra-Frame Reconstruction

Edge points are detected using the Canny detector [2]. Note that edge points can be obtained from not only long segments but also fine textures. Thus, our system can build maps from textures on soil and asphalt.

Intra-frame reconstruction (i.e., between the left and right images) is performed based on the epipolar geometry in parallel stereo. We search the matching pair of edge points between left and right images along the scanline since epipolar lines are horizontal for parallel binocular stereo cameras. The matching criterion is the normalized correlation of a small window around the edge point. Also, the orientation of the image gradient at the edge point is optionally used to reduce outliers.

Multiple candidate matches are inevitably obtained especially when the edge direction is nearly parallel with the epipolar line. The DP matching approach [11] is employed to address this problem. After determining the matching pair, the locations of edge points are refined at subpixel level. We employ bilinear interpolation for subpixel matching.

B. Inter-Frame Registration Using Edge-Point ICP

A key to stereo SLAM is the estimation of the camera motion between frames. Unlike intra-frame reconstruction, epipolar geometry is unknown in inter-frame registration. Thus, edge point correspondences are hard to obtain. Edge points are prone to aperture problem, which can cause multiple candidate matches including outliers. Random sampling approaches for robust estimation such as RANSAC [5] are not suitable due to a large number of edge points and multiple candidate matches. To address this problem, we apply the ICP algorithm [1] to the matching of the image edge points and 3D edge points via perspective projection.

1) Camera Motion Estimation: The camera motion from time $t-1$ to t can be estimated by matching the edge points in frame I_{t-1} and those in frame I_t. We employ 3D-2D matching, in which the 3D points reconstructed from I_{t-1} are matched with the 2D points detected in I_t. This method is stable since errors in depth have less influence on the registration accuracy due to perspective projection. The 3D-2D matching has been employed for model-based tracking [8] and visual odometry [10].

The registration is performed by a variant of ICP algorithm on the image plane. Let r_t be the camera pose at t, P^i_{t-1} be the i-th 3D edge point reconstructed at $t-1$, and p^i_{t-1} be the projected point of P^i_{t-1} onto image I_t. Let q^i_t be the image edge point at t which corresponds to p^i_{t-1}. We define cost function $G()$ as follows.

$$G(r_t) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} d(q^i_t, p^i_{t-1}) \qquad (1)$$

Here, $d(q^i_t, p^i_{t-1})$ is the perpendicular distance between p^i_{t-1} and the edge segment which contains q^i_t.

Camera motion r_t and edge point correspondences are searched by minimizing $G(r_t)$ using a gradient descent method. The initial value of r_t is set to r_{t-1}, and the initial corresponding point q^i_t is set to the edge point which is the closest to p^i_{t-1} in terms of Euclidean distance. By repeating the minimization of $G(r_t)$ and edge point matching, the optimal r_t and edge point correspondences are obtained. Here, we assume that the camera motion between frames is small. If the camera motion is large, the ICP may fall into local minima and the registration will fail. In this case, the failure is detected using a measure to be mentioned in Section III-B.

Due to a large number of edge points, the above ICP algorithm is time consuming if all the edge points are used. Thus, we employ a coarse-to-fine approach to improve efficiency. In the early stage, we roughly estimate the camera motion with a small number of edge points, and then we refine the camera motion and edge point correspondences by increasing the number of edge points [14].

2) Map Building: Based on the camera pose r_t obtained, a tentative 3D map is built by transforming the intra-frame 3D points from the camera coordinates to the world coordinates. Let P^i_c be the i-th 3D point in the camera coordinates. The location of 3D point P^i can be written as $P^i = g(P^i_c, r_t)$, where $g()$ is the coordinate transformation based on the rotation and translation components of r_t.

The 3D points created at each frame are fused according to the edge point correspondences between the frames. The 3D points obtained here are just used as a tentative map for the camera motion estimation between keyframes. The final map is created by the keyframe adjustment to be mentioned below.

C. Keyframe Adjustment

The algorithm mentioned above suffers from accumulated errors. To reduce the accumulated errors, the camera motion and the 3D map are re-estimated by keyframe adjustment.

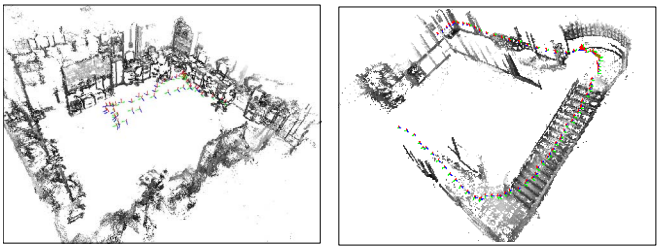


Fig. 1. Our edge-point based SLAM creates detailed 3D maps. Left: a room of $7[m] \times 8[m]$, right: corridors and stairs of about 40[m].

In keyframe adjustment, keyframes are extracted with an interval d from the image sequence, and the camera motion between keyframes is refined using the ICP algorithm in the manner mentioned above. In the process, each 3D point P^i_{t-d} in keyframe I_{t-d} is projected onto keyframe I_t, and matched with image edge point q^i_t. Due to the errors accumulated between keyframes, P^i_{t-d} will not be projected exactly onto the corresponding image edge point. The discrepancy is small in general, and the keyframe ICP successfully converges closely into the global minimum to update the camera motion and the locations of the 3D points.

The keyframe adjustment just refines the camera motion between keyframes. It is unavoidable that small errors accumulate in a long sequence even using the keyframe adjustment. To reduce long-term accumulated errors, the bundle adjustment will be necessary at a loop closing phase, but it has yet to be done.

A 3D map is built by aligning the 3D points reconstructed in the keyframe in the manner mentioned in Section II-B.2. Fig. 1 shows examples of the maps created by the method.

III. Failure Detection and Recovery

A. Types of Failure

This paper treats the following types of failure.

- Erratic camera motion
  Rapid camera motion due to shake, slippage and collision can cause failure in feature tracking, and the ICP can easily fall into local minima. Such motions are often accompanied by motion blur, which drastically reduces edge points and increase the probability of failure.
- Moving objects
  The camera pose estimation requires static features. Moving objects increase the probability of failure since they decrease reliable static features and increases outliers and reprojection errors in the ICP.
- Illumination changes
  Illumination change significantly affects image intensity and can increase outliers.

B. Failure Detection

A measure of failure detection would be the average matching error $G(r_t)$ in the ICP. If $G(r_t)$ is larger than a threshold, the registration process is regarded as failed. However, $G(r_t)$ tends to be large when the image contains many outliers caused by moving objects. This makes it

difficult to determine an appropriate threshold for $G(r_t)$. Therefore, we employ the ratio of matched edge points as a score of failure detection.

$$\begin{aligned} R_t &= \frac{|M_t|}{|E_t|} \\ M_t &= \{q_t^i \mid d(q_t^i, \exists p_{t-1}^i) \leq th_1, q_t^i \in E_t\} \end{aligned} \tag{2}$$

Here, E_t is the set of edge points detected in frame I_t, and M_t is the set of the edge points which are matched with a 3D point in the map with the reprojection error of th_1 or less. If R_t is smaller than a threshold th_2, the registration process is regarded as failed. Intuitively, R_t is the ratio of well-matched edge points, and its threshold th_2 is relatively easy to determine.

If the number of edge points largely changes per frame, R_t may not be an appropriate score. For instance, if the number of edge points detected in frame I_t is half of that in I_{t-1}, the actual matching ratio between I_t and I_{t-1} is half of R_t. This situation is caused by motion blur and illumination changes. To cope with this problem, the following score is employed.

$$S_t = R_t \times \frac{|E_t|}{|E_{t-1}|} = \frac{|M_t|}{|E_{t-1}|} \tag{3}$$

Note that under good camera motion, the number of edge points between consecutive frames do not change significantly since the displacement between consecutive frames is small. The number of edge points largely changes only when some failure occurs.

By examining whether S_t is less than a threshold th_2, we detect the failure mentioned in the previous section. Erratic camera motion and moving objects will decrease S_t due to false matching, and the accompanying motion blur will significantly decrease S_t due to the reduction of detected edge points. Illumination change will decrease S_t mainly due to the reduction of edge points shared between frames. Unfortunately, S_t cannot discriminate failure types. The appropriate camera motion for recovery is different between erratic camera motion and moving objects as will be mentioned below. This must be determined by other information source such as motion sensors.

C. Recovery from Failure

We assume that the system can take an image similar to the image just before the failure by appropriate camera manipulation. For example, in the case of erratic camera motion by small slippage or collision, the system can obtain a similar image by returning the camera to the correct pose. In the case of moving objects and illumination change, the system can obtain a similar image by waiting for the condition to recover.

1) Scores for Recovery: Suppose that the system successfully builds a map up to frame I_{t_0} and failure occurs at frame I_{t_0+1}. Then, the system enters into the recovery mode to find a frame which is similar to I_{t_0}. If the good frame I_t is found, the system resumes the registration process from I_t. The system searches a frame similar to I_{t_0} by manipulating the camera in the case of failure due to erratic camera motion. The system searches a frame similar to I_{t_0} by waiting for the moving objects to pass by in the case of failure due to moving objects. For illumination change, the system waits for the image intensity to be stable. We assume that the camera has automatic exposure. Then, the image intensity changes a moment and soon returns to the almost same level before the illumination change.

We define image similarity scores to search a good frame to recover. The poses and scales of objects in the image change as the camera pose changes. To evaluate the image similarity independently of such changes, we utilize the SIFT descriptor [9]. The SIFT descriptor is an orientation histogram over 4×4 sample regions around the key point. The SIFT descriptor is invariant to image rotation, translation, scale change, and small viewpoint change. Although the conventional SIFT use blobs and corner points as key points, we extend the descriptor to use edge points as key points [13]. This is because blobs and corner points cannot be detected sufficiently from non-textured environments. On the other hand, edge points can be detected from even a few edge lines in non-textured environments. Edge points are prone to aperture problem, which can cause location uncertainty. This is addressed by the similarity transformation estimation using a voting scheme as will be mentioned later.

We define two scores for image similarity. One is the ratio of matched edge points between I_{t_0} and the current frame I_t. The similarity between two edge points q_1 and q_2 is defined as the normalized correlation of the SIFT descriptors of q_1 and q_2. If the correlation exceeds a threshold th_3, q_1 and q_2 are regarded as matched. A similarity score Q_t of frame I_t against I_{t_0} is defined as follows.

$$\begin{aligned} Q_t &= \frac{|m_t|}{|E_t|} \\ m_t &= \{q_t^i \mid D(q_t^i, \exists q_{t_0}^j) \geq th_3, q_t^i \in E_t\} \end{aligned} \tag{4}$$

m_t is the set of the edge points in E_t which are matched with an edge point in E_{t_0} by SIFT descriptors. $D()$ is the normalized correlation function between SIFT descriptors.

The other similarity score is based on the similarity transformation between I_{t_0} and I_t. As the transformation parameters are close to 0 for rotation and translation, and to 1 for scale ratio, the similarity between images is high. To find the transformation parameters, we employ a voting scheme in terms of similarity transformation. For each pair of edge points matched between the two frames, we calculate the similarity transformation parameter, and vote in the parameter space (translation, rotation, and scale ratio).

First, the rotation angle $\Delta\theta$ between each pair of edge points in I_{t_0} and I_t is computed as Eq.(5). Here, θ_i and θ_j are the orientations of an edge point in I_{t_0} and an edge point in I_t, respectively. The orientation of an edge point is obtained by the Canny detector. The scale ratio s between each pair of edge points in I_{t_0} and I_t is also computed as Eq.(6). Here, s_i and s_j are the scales of an edge point in I_{t_0} and an edge point in I_t, respectively. The scale of an edge point is obtained by the scale space analysis presented

in [7]. We calculate $\Delta\theta$ and s for each pair of edge points in I_{t_0} and I_t, and cast a vote in the corresponding bin in the rotation-scale space. Local maxima in the voting space are employed as candidate of rotation angle and scale ratio.

$$\Delta\theta = \theta_i - \theta_j \quad (5)$$

$$s = s_i/s_j \quad (6)$$

Then, the translation Δx and Δy are computed for each candidate of $\Delta\theta$ and s as Eq.(7) and Eq.(8). (x_i, y_i) is the location of an edge point in I_{t_0}, and (x_j, y_j) is the location of an edge point in I_t, and we calculate Δx and Δy for each pair of edge points in I_{t_0} and I_t, and cast a vote in the corresponding bin in the translation space. The maximum in the voting space is employed in this voting.

$$\Delta x = x_i - s\ (x_j cos\Delta\theta - y_j sin\Delta\theta) \quad (7)$$

$$\Delta y = y_i - s\ (x_j sin\Delta\theta + y_j cos\Delta\theta) \quad (8)$$

Now, similarity score T_t is defined using transformation parameters $\tau = (\Delta x, \Delta y, \Delta\theta, s)$.

$$T_t = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (k_1\Delta\theta)^2 + (k_2(s-1))^2} \quad (9)$$

k_1 and k_2 are constants for weighting the parameters. In implementation, $k_1 = 2, k_2 = 100$. In the case that I_{t_0} and I_t are completely different, $Q_t \approx 0$ and T_t is undefined.

RANSAC [5] could be used for estimating the similarity parameters, but is less suitable due to a large number of edge points with multiple candidate matches. The multiple candidate matches increase the actual outlier ratio, which makes RANSAC inefficient.

Q_t and T_t are correlated with the displacement between frames, and they are good measures to evaluate the similarity of the frames. On the other hand, S_t cannot be used as a similarity measure when the displacement is large. This is because the ICP will fall into a local minimum with a high probability and S_t will be randomly set to a small value.

2) Procedure of Failure Recovery: The pseudo code of failure recovery is shown below. First, the scores Q and T and similarity transformation τ are calculated between I_{t_0} and I_t (calSimilarity). If both of Q and T meet their thresholds, the procedure returns I_t. The system resumes the registration between I_{t_0} and I_t. If either of Q and T does not meet its threshold, the robot (or human) capture a new image by manipulating the camera (moveCamera, captureImage) and repeats the evaluation of Q and T. In this step, the robot should control the camera based on τ in order to make the new image similar to the I_{t_0}. Since the ICP easily falls into local minima when the displacement is large, the camera control using τ is essential for efficient recovery. This paper, however, does not consider the camera control, and new images were taken by a human in the experiments.

Algorithm 1 find a good image for failure recovery.

```
(Q, T, τ) = calSimilarity(I_t0, I_t)
while Q < th_4 ∨ T > th_5 do
  t = t + 1
  moveCamera(τ)
  I_t = captureImage()
  (Q, T, τ) = calSimilarity(I_t0, I_t)
end while
return (I_t)
```

IV. Experiments

We conducted experiments in indoor and outdoor environments. Images were captured manually by human with Point Grey Research's binocular camera Bumblebee2. The baseline distance is 120 [mm]. The image size was reduced to 320×240 pixels. No motion sensors were used.

A. Failure Detection

We evaluated the score S_t of failure detection for erratic camera motion, moving objects, and illumination changes. The ground truth, i.e. whether the registration for each frame fails or not, was determined by human.

1) Erratic camera motion: In this experiment, the camera was moved rapidly to make a large displacement between consecutive frames. In many cases, motion blur was accompanied. Fig.2 (a) shows an example. To create examples without motion blur, we skipped several frames from image sequence captured by normal camera motion. Fig.2 (b) shows an example. The left image is I_{t-1} and the right image is I_t in both examples. In the examples, the reprojection of the 3D map are represented by red dots. As can be seen in Fig.2 (b), the 3D map are mismatched with the image. In both examples, failure occurred at I_t, and S_t decreased significantly.

Fig. 3(a) shows the distribution of S_t for 248 frames. The frames consists of 32 scenes with erratic camera motion. The last frame of each scene was captured under rapid camera motion, and each scene has several consecutive frames captured under moderate camera motion before the last frame. Therefore, there were 32 erroneous frames and 216 normal frames. In this experiment, all the successful frames were the normal frames, and all the failed frames were the erroneous frames.

2) Moving Object: Fig.2 (c) and (d) show examples of failure by moving objects. In both examples, failure occurred at I_t (right image), and S_t decreased significantly. As can be seen in I_t of Fig.2 (c) and (d), the 3D map are erroneously matched with the moving object.

Fig.3(b) shows the distribution of S_t for 310 frames. All the frames contains moving objects. In general, the registration fails when the size of the moving objects is large. However, if stable edge points are not sufficiently extracted from the image, the registration can fail due to small moving objects. The right image in Fig.2(c) shows an example of it. The peak of the distribution lies at lower value of S_t than the Fig.3(a). This is because the matched edge points were reduced by moving objects.

3) Illumination Change: We made illumination changes by turning on/off the light in rooms and by moving around the boundary of the sunshine and shade. Fig. 2(e) and (f) show examples. Since the camera used here has automatic exposure, the image intensity changed at the moment of

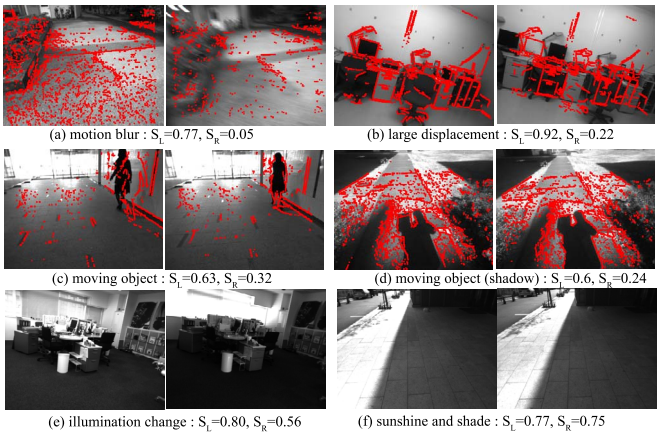


Fig. 2. Examples of score S_t under erratic camera motion, moving objects, and illumination change.

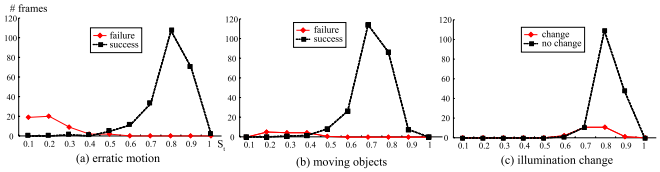


Fig. 3. Distribution of score S_t

illumination change and returned to the almost same level before the illumination change. Thus, the change in the image intensity was not so drastic.

Fig. 3(c) shows the distribution of S_t for 163 frames. The frames consists of 16 scenes with illumination change. The last frame of each scene was captured under illumination change, and each scene has several consecutive frames captured under stable illumination before the last frame. Therefore, there were 16 frames under illumination change and 147 normal frames. In this experiment, illumination change had only slight influence on the registration, and all the frames were successfully processed including illumination change. The edge point detection is robust to illumination changes, and registration failure will not occur unless illumination change is very large.

An important issue is how to determine the threshold th_2. When th_2 is small, the probability of mismatching increases and as a result the 3D map will have large errors. Empirically, when S_t is very small (0.3 or less), the process sometimes fails catastrophically by motion blur, large skips, and large occlusion. When S_t is moderate (0.4 to 0.5), the process has registration errors but is continuable. From these facts, the appropriate value of th_2 would be 0.35 to 0.5 depending on the required efficiency and accuracy. If th_2 is large, the process may frequently enter into the recovery mode even for correct matching, which makes the process inefficient.

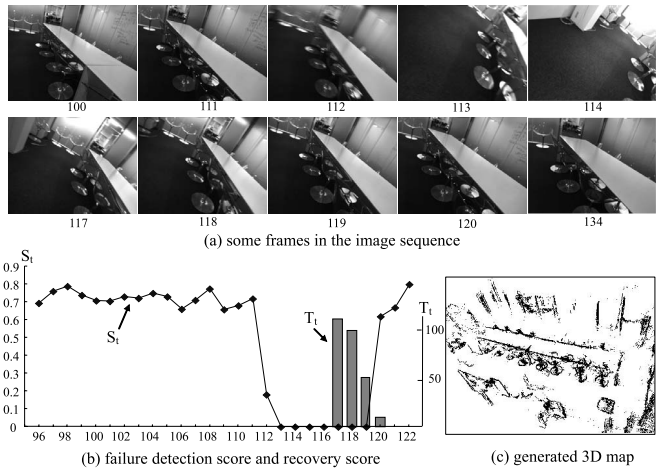


Fig. 4. Recovery from failure caused by rapid camera motion

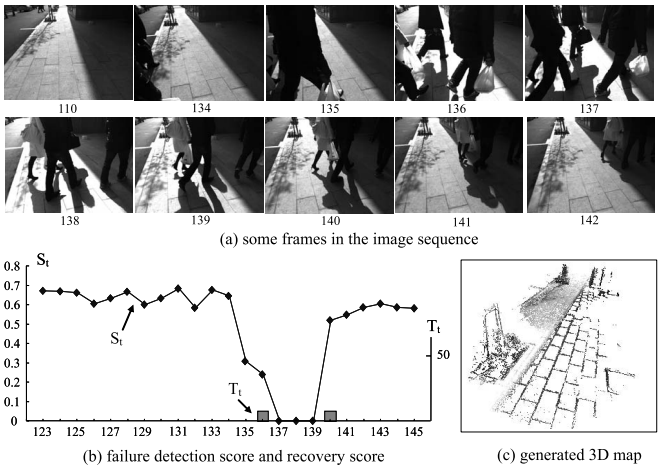
B. Failure Recovery

We conducted experiments of failure recovery. The first experiment examined recovery from failure caused by erratic camera motion in mapping from 134 frames. Fig.4(a) shows some frames of the image sequence. At frame 112, the operator made erratic camera motion by turning the camera away rapidly, and then returned the camera slowly to the pose before the averting. Fig.4(b) shows S_t around frame 112. As can be seen, S_t significantly decreases at frame 112. T_t meets the threshold $th_5 = 20$ at frame 120, and the system resumed the mapping process. T_t is undefined from frame 112 to 116 since Q_t is very small. Fig.4(c) shows the 3D map built from the 134 frames.

The second experiment examined recovery from failure caused by moving objects in mapping from 230 frames. Fig.5(a) shows some frames of the image sequence. At frame 135, passersby occupied a large region in the image and the registration failed. After waiting for the passersby to pass by, the system resumed the mapping process at frame 140. Fig.5(b) shows S_t around frame 135. As can be seen, S_t significantly decreases at frame 135. T_t meets the threshold $th_5 = 20$ at frame 136, and the system resumed the mapping process but failed again. Note that S_t is very small at frame 136. T_t meets the threshold $th_5 = 20$ at frame 140, and the system successfully resumed the mapping process. T_t is undefined from frame 137 to 139 since Q_t is very small. Fig.5(c) shows the 3D map built from the 230 frames.

C. 3D Mapping with Failure Recovery

Fig.6 shows an example of 3D mapping with failure recovery from 990 frames. The travel distance is about 45[m]. At frame 408 in Fig.6, the operator made erratic camera motion by turning the camera away rapidly, and returned the camera slowly to the pose before the averting. The system entered into the recovery mode at frame 408, and recovered from the failure at frame 459. Soon after this, another failure occurred by small shake, and was recovered again. At frame 618, the operator made another erratic camera motion, and the system

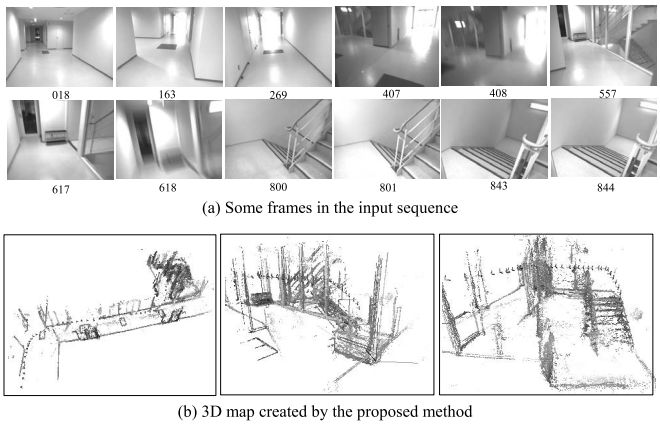


(a) some frames in the image sequence

(b) failure detection score and recovery score

(c) generated 3D map

Fig. 5. Recovery from failure caused by moving objects



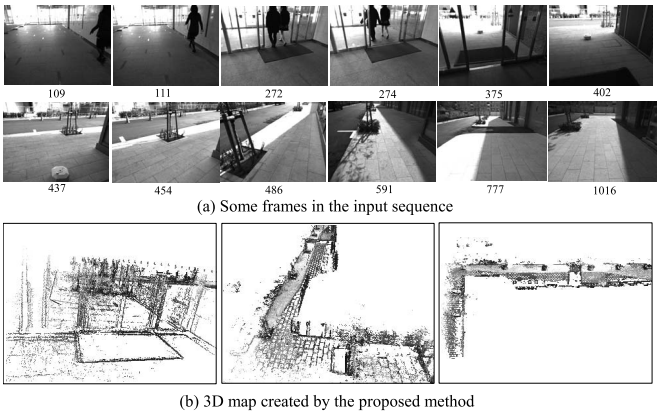
(a) Some frames in the input sequence

(b) 3D map created by the proposed method

Fig. 6. 3D mapping under rapid camera motions and illumination changes.

recovered from the failure at frame 640. Illumination changes occurred at frame 801 and 844 because of automatic lights at the landing, but the illumination changes had no influence on the mapping process.

Fig.7 shows another example of 3D mapping with failure recovery from 1040 frames. The travel distance is about 60[m]. At frame 109, the registration failed due to a passerby. Stable edge points were not sufficiently detected from the background due to the darkness (weak intensity) of the image, and a small moving object was able to affect the registration process. The system recovered from the failure after two frames. At frame 113, the registration failed due to the same passerby again and recovered after two frames. At frame 272, the registration failed due to passersby and an automatic door. The system recovered from the failure after two frames again. The movement of the second automatic door had no influence. Illumination changes occurred when going outside through the door and moving around the boundary of the sunshine and shade, but the illumination changes had no influence on the mapping process.



(a) Some frames in the input sequence

(b) 3D map created by the proposed method

Fig. 7. 3D mapping under moving objects and illumination changes.

V. Conclusions

This paper has presented a method of 3D mapping using a stereo camera with automatic recovery from registration failure. We employ edge points as map element, and apply a variant of ICP algorithm to the inter-frame registration. Our method detects failure in camera pose estimation caused by erratic camera motion or moving objects and recovers from the failure by searching a good image to resume the registration using edge-point SIFT descriptors. In experiments, our method successfully coped with failure in 3D mapping under various conditions. Future work includes the integration of the proposed system and a mobile robot platform.

References

[1] P. J. Besl and N. D. Mckay: "A Method of Registration of 3-D Shapes," *IEEE Trans. on PAMI*, Vol. 14, No. 2, pp. 239-256, 1992.

[2] J. Canny: A Computational Approach to Edge Detection, *IEEE Trans. PAMI*, Vol. 8, No. 6, pp. 679-698 (1986).

[3] D. Chekhlov M. Pupilli, W. Mayol-Cuevas, and A. Calway: "Real-Time and Robust Monocular SLAM Using Predictive Multi-resolution Descriptors," *Proc. of ISVC2006*, 2006.

[4] P. Elinas, R. Sim, and J. J. Little: "σ $SLAM$: Stereo Vision SLAM Using the Rao-Blackwellised Particle Filter and a Novel Mixture Proposal Distribution," *Proc. of ICRA2006*, pp. 1564–1570, 2006.

[5] M. Fischler and R. Bolles: "Random Sample Consensus: a Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography", *Communications ACM*, 24:381-395, 1981.

[6] M. A. Garcia and A. Solanas: "3D Simultaneous Localization and Modeling from Stereo Vision," *Proc. of ICRA2004*, 2004.

[7] T. Lindberg: "Feature Detection with Automatic Scale Selection," *Int. J. of Computer Vision*, 30(2), pp. 79-116 (1998).

[8] D. Lowe: Fitting Parameterized Three-Dimensional Models to Images, *Trans. on IEEE PAMI*, Vol. 13, No. 5, May, pp. 441–450, 1991.

[9] D. Lowe: "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. of Computer Vision*, 60(2), pp. 91-110 (2004).

[10] D. Nistér, O. Naroditsky, and J. Bergen: "Visual Odometry," *Proc. of CVPR2004*, 2004.

[11] Y. Ohta and T. Kanade: Stereo by intra- and inter- scanline search using dynamic programming," *Trans. on IEEE PAMI*, Vol. 7, No. 2, pp. 139–154, 1985.

[12] S. Se, D. Lowe, and J. Little: "Local and Global Localization for Mobile Robots using Visual Landmarks," *Proc. of IROS2001*, 2001.

[13] M. Tomono: "3-D Object Map Building Using Dense Object Models with SIFT-based Recognition Features," *Proc. of IROS2006*, 2006.

[14] M. Tomono: "Robust 3D SLAM with a Stereo Camera Based on an Edge-Point ICP Algorithm," *Proc. of ICRA2009*, 2009.

[15] B. Williams, P. Smith, and I. Reid: "Automatic Relocalisation for a Single-Camera Simultaneous Localisation and Mapping System," *Proc. of ICRA2007*, 2007.