3D Reconstruction of Edge Line by ICP-based Matching with Geometric Constraints

Kojiro Takeyama
Toyota Central Research and Development Labs. Inc.
Nagakute, Japan
takeyama@mosk.tytlabs.co.jp

***Abstract*—This paper presents a novel edge-based 3D reconstruction method using a monocular camera. The edge information is known to be illumination-invariant and to include abundant structural information in a relatively small number of pixels. However, since edge line cannot explicitly determine the pixel-to-pixel correspondence as in the feature point approach, it is difficult to perform accurate matching of pixels in a scene with dense edge lines. In this study, edge-based 3D reconstruction using ICP (iterative closest point algorithm) with geometric constraints has been proposed. In our approach, quasi-rigid body assumption for the edge line deformation and smart search for the matching process are introduced for the improvement of matching robustness in scenes with dense edge lines. Experimental results show that the performance of our method for both motion parallax estimation and depth estimation is greatly improved compared with two recent edge-based methods.**

Keywords—SFM; spatial map, edge line; monocular camera; iterative closest point algorithm; (key words)

I. INTRODUCTION

Map information, including the 3D structure of the environment, plays an important role in a variety of applications such as robot navigation or augmented reality. Structure from motion (SFM)[1] is a well known method for building 3D maps, which can be used in low cost small devices such as smart phones or UAVs. Classical SFM techniques used feature point descriptors to calculate the motion parallax for the reconstruction of 3D maps, but feature points include poor structural information. In recent approaches, dense and semi-dense direct method have shown successful performance in constructing 3D maps with rich structural information, however, computationally expensive for low cost embedded systems. In addition, feature point and direct method are less robust to illumination change because they use pixel intensity as a feature.

Some recent studies have focused on edge information as feature to overcome the problems of feature points and direct methods. Since edge line directly represent the shapes of structures in the environment by thin lines, they can efficiently contain rich structural information in a relatively small number of image pixels. Furthermore, edge line is robust to changes in illumination because they do not directly use the intensity of the pixels. On the other hand, the main challenge with using edge line is in the matching process since edge line cannot explicitly determine the pixel-to-pixel correspondence as in the feature point approach. Iterative closest point algorithm (ICP) based matching[9] generally provides good solutions for edge line matching if the motion parallax between the edge lines is sufficiently small; otherwise, ICP algorithm may easily converge to wrong solutions because pixels that should not correspond to each other may become the nearest neighbors and associated. In particular, in a scene where edge lines are extracted at close intervals on structures, such as in urban areas, even a slight parallax may cause false correspondences since it is difficult to distinguish between adjacent edge line at close intervals.

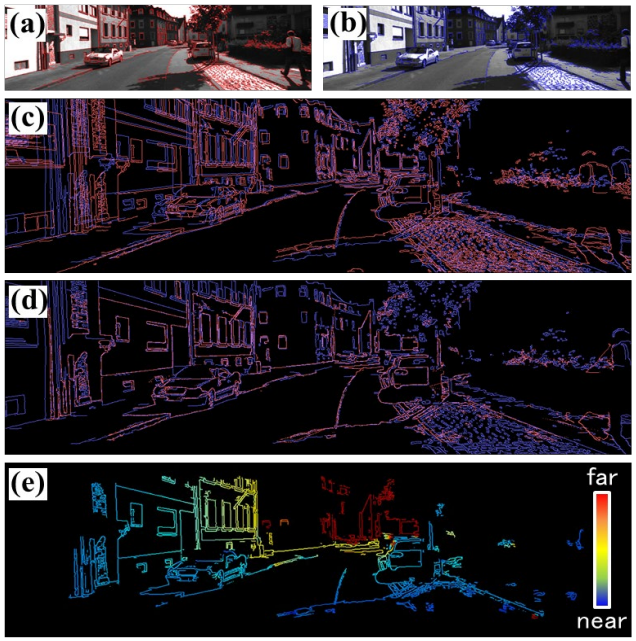

Fig. 1: The outline of our method. (a)-(b)edge images extracted from consecutive monocular images, (c)overlaid edge image of (a) and (b), (d)edge image after matching, and (e)depth estimated from the motion parallax calculated in (d).

In this study, we propose a novel method using ICP with geometric constraints to improve the robustness of the matching in an environment with dense edge lines. In our approach, quasi-rigid body assumption for the edge line is introduced, which allows to preserve the overall edge line geometry while having local flexibility in the matching process. Further, constraints using epipolar geometry and line gradient

are introduced to the closest point search to avoid false correspondences during ICP process. Fig. 1 shows the outline of our method.

II. Related works

Previous approaches for image-based 3D reconstruction are categorized by their feature types, and their characteristics are described below.

A. Feature-point-based method

Feature-point-based 3D reconstruction techniques [2,3] have been widely used because of their low computational cost and simple methodology. However, since most of the algorithm relies on the hypothesis that the pixel intensity is almost constant over the consecutive camera images, the robustness to the illumination change is low. Furthermore, the reconstructed map with sparse 3D points contains less structural information.

B. Direct methods

The direct method is a rapidly growing topic in recent years that performs pixel-to-pixel matching based on the pixel intensities without using complex features. Dense direct method [4,5,17] computes all the pixels in the image with global consistency check. The advantage of this algorithm is that a full dense map can be constructed. However, the computational cost is high due to the computation of the whole image pixels. A semi-dense direct method [6,7] has reduced the computational cost by ignoring non-textured areas, but the number of pixels used in the algorithm is still high for the low cost CPU. Further, the robustness against illumination is low due to the intensity-based matching algorithm.

C. Edge-based methods

The edge-based method uses the edge lines of objects in camera images. The advantages of using the edge line is the robustness to the illumination change and the rich structural information in a small number of pixels. Ramalingam [8] adopted line-sweep technique to match the edge lines, which uses the optical flows of nearby feature points as a clue to track edge pixels. However, the performance depends on the accuracy of optical flows, thus the robustness is not enough. Tarrio [9] worked on ICP-based edge-matching algorithm for consecutive monocular images. ICP determines the correspondences between the edge pixels by closest point search. It works well if edge lines are sufficiently sparse or their motion parallaxes are small enough; otherwise, the correspondences become inaccurate and the overall estimation performance can be easily degraded. Kim [10] and Wang [11] used the depth information provided by RGB-D camera, which gives an accurate initial estimate of motion parallax to improve the accuracy of ICP. However, RGB-D depth information is not robust in sun light, further the extra sensor cost is required.

In this study, we propose a method to improve the robustness of ICP-based approach, which works accurately in areas with dense edge lines, such as urban environments. In the proposed method, geometric constraints are introduced to ICP in order to solve the problems of conventional ICP method.

III. Proposed method

A. Methodology

The conventional ICP has problems with matching stability, which comes from two reasons. One is that the shape of the edge line is not properly used as a matching cue. This is due to the fact that the shape of edge line is unstable during the matching process since each pixel has degree of freedom. Another is that it is difficult to determine the pixel-to-pixel correspondence using edge information. Though the closest point search in ICP process determines the pixel-to-pixel correspondence based on the distance between the edge pixels, false correspondences frequently occur in the dense edge line area.

There are two approaches to addressing the problems of ICP using geometric constraints.

The first approach is to introduce quasi-rigid body assumption to the edge line deformation, which properly reduces overall degree of freedom of edge pixels. In this approach, overall edge line is divided into multiple edge segments at branch points, then each edge segment is assumed to be the rigid body existing in the same plane in 3D space (planar constraint), and further, each branch point is assumed to be a joint that connects the edge segments with degree of freedom (connectivity constraint). The quasi-rigid body assumption enables the deformation of edge line which preserves the overall shape of the edge line while maintaining local flexibility. In the typical road environments, the assumptions in the proposed method are reasonable since each edge segment obtained by dividing the edge line around the branching point usually exists on flat surface on the object, shown as Fig. 2. There are exceptional cases, such as where edge segments on an object with complex curved shape do not exist in a plane, or when edge segments connected by occlusion boundaries are discontinuous in 3D space, but these cases can be handled by outlier removal.

The second approach is to introduce constraints using epipolar geometry and line gradient as a cue to improve closest point search in ICP process. These constraints narrow down the candidates of correspond pixel, which contributes to reducing the false correspondences in scenes where the closest point search algorithm does not work correctly.

(a) Raw input image (b) Edge segments

Fig. 2: Raw input image and edge segments. the right image shows the edge segments obtained by dividing the edge line by branch points (each edge segment is visualized wtith a different color)

Fig. 3 shows the entire process of the proposed method. First, edge line is extracted from two consequtive camera images

(image1 and image2). After that, joints (branch points) are extracted from the edge line in image1 which is a matching source. Next, the edge line in image1 is superimposed on the image2, and the matching of edge line is performed. Finally, the depth is calculated based on the motion parallax obtained from matching result. Note that in the depth estimation process, a reliability test is performed to remove pixels with large error. The detail of the process is described in the following sections.

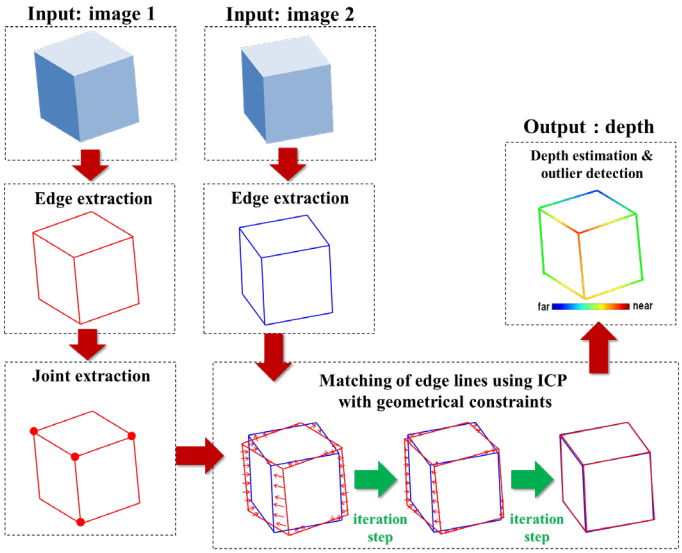

Fig. 3: Overview of the proposed approach. After edge extraction, the edge line of image 1 is overlaid to that of image 2, and they are matched using ICP with geometric constraints. Finally, the depth is estimated, and outliers are removed.

B. *Joint extraction*

The detail of the branch point extraction algorithm is shown in Fig. 4. For each edge pixel, eight adjacent pixels are used to determine whether or not they are branch points. The key is the continuity of the eight adjacent pixels, and if the number of discontinuous boundaries between the eight adjacent pixels is six or more, the edge pixel is determined as a branch point. Note that, for convenience, the endpoint of an edge, which has two discontinuous boundaries, is also defined as a branch point. After the extraction of branch points, all edge pixels around the branch point are grouped into different edge segments.

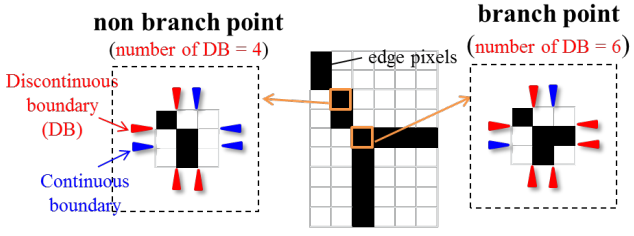

Fig. 4: Extraction of branch point. If the number of discontinuous boundaries between eight adjacent pixels is 6 or more, the edge pixel is determined as a branch point.

C. *Matching of edge pixels*

In this section, the matching method of edge pixels in consecutive camera images is detailed. First, the basic method of ICP[1] is explained, then our proposed method is explained.

1) Basics of matching using ICP

A basic algorithm of ICP is shown that matches two edge segments $\mathbf{L1}$ and $\mathbf{L2}$. $\mathbf{L1}$ and $\mathbf{L2}$ are the edge segments extracted in consecutive camera images 1 and 2, and they are represented as follows:

$$\mathbf{L1} = \{\mathbf{u_i}\}, \quad \mathbf{L2} = \{\mathbf{v_j}\} \tag{1}$$

where $\mathbf{u_i}$ and $\mathbf{v_j}$ are the edge pixels that constitute $\mathbf{L1}$ and $\mathbf{L2}$, respectively. i and j are the indices of the pixels in edge segments. The cost function of the matching optimization is defined as follows:

$$f(\mathbf{R}, t, Z_i) = \sum_{i \in \mathbf{L1}} \mathrm{dmin}\left(P(\mathbf{R}, \mathbf{t}, \mathbf{u_i}, \mathbf{Z_i}), \mathbf{L2}\right) \tag{2}$$

where $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and the translation vector of the camera pose, respectively, Z_i is the depth of pixel i, P is the projection function that projects the position of $\mathbf{u_i}$ in image 1 onto the image 2 based on the camera motion and the depth, and $\mathrm{dmin}(,)$ is the function that calculates the minimum distance from the arbitrary pixel to set of edge pixels. The algorithm searches for the $\mathbf{R}$, $\mathbf{t}$ and Z_i which minimize the cost function. However, the accuracy of the solutions of (2) may become unstable since the number of unknown parameter Z_i is equal to the number of edge pixels of $\mathbf{L1}$ and it usually becomes large. In order to deal with this problem, our proposed method set geometric constraints on the pixels of $\mathbf{L1}$ so that the number of unknown parameters is reduced. The geometric constraints are also used to avoid the false correspondences between edge pixels. The detail of the geometric constraints are described in the next paragraph.

2) Introduction of geometric constraints

There are two approaches using geometric constraints to improve the robustness of matching. The first is to introduce quasi-rigid body assumption to the edge line, and the second is to introduce the constraints to the closest point search. The details of these two approaches are described below.

Quasi-rigid body assumption:

The assumption of quasi-rigid body is that a set of pixels that constitutes an edge segment is treated as a rigid body existing on the same plane in 3D space (planar constraint), and further, edge segments are connected by joints having degree of freedom (connectivity constraint). The advantage of introducing quasi-rigid body is that both global rigidity and local flexibility are given to the edge line. In other words, the number of unknown parameters estimated in ICP is properly reduced using these constraints. In the following, it is shown that the number of unknown parameters is reduced in the

process of formulation of the planar constraint and connectivity constraint.

The planar constraint relies on the hypothesis that all the pixels in an edge segment **L1** are on the same plane in 3D space, which is represented as follows:

$$Z_i = \alpha X_i + \beta Y_i + \gamma \tag{3}$$

where X_i, Y_i, Z_i are the coordinates of pixel *i* in 3D space, and α, β, γ are the coefficients of the planar constraint. Then, considering the equation of camera projection represented as (4), (3) becomes (5):

$$x_i = \frac{X_i}{Z_i} f + p_x \ , \quad y_i = \frac{Y_i}{Z_i} f + p_y \tag{4}$$

$$Z_i = \frac{-f\gamma}{\alpha(x_i - p_x) + \beta(y_i - p_y) - f} \tag{5}$$

where *f* is the focal length of the camera, p_x and p_y are the coordinates of the camera center. From (5) it is seen that the number of unknown parameters of Z_i, which is equal to the number of pixels constituting L1, is reduced to only 3 (α, β, γ).

In the next step, the connectivity constraint is introduced to consider the connection relationship of edge segments. Up to this point, we focused on the matching of a single edge segment, but now turn to the matching of multiple edge segments. First, the cost function considering all the edge segments, shown as (6), is derived by modifying (2).

$$f(\mathbf{R}, \mathbf{t}, Z_{L1,i}) = \sum_{\mathbf{L1} \in \mathbf{L1_{all}}} \sum_{i \in \mathbf{L1}} \mathrm{dmin}\left(P(\mathbf{R}, \mathbf{t}, \mathbf{u_{L1,i}}, Z_{L1,i}), \mathbf{L2_{all}}\right) \tag{6}$$

where $\mathbf{L1_{all}}$ and $\mathbf{L2_{all}}$ are sets of entire edge segments in image 1 and 2, respectively. $\mathbf{u_{L1,i}}$ and $Z_{L1,i}$ are the image coordinate and depth at pixel *i* of the edge segment **L1**, respectively.

After that, connectivity constraint is formulated based on the premise that edge segments connected to each other share joints. Assuming that the depth of two joints at both ends of an edge segment **L1** are $Z_{L1,1}$ and $Z_{L1,2}$, respectively, the (7) can be derived from (5):

$$Z_{L1,1} = \frac{-f\gamma_{L1}}{\alpha_{L1}(x_{L1,1} - p_x) + \beta_{L1}(y_{L1,1} - p_y) - f}$$
$$Z_{L1,2} = \frac{-f\gamma_{L1}}{\alpha_{L1}(x_{L1,2} - p_x) + \beta_{L1}(y_{L1,2} - p_y) - f} \tag{7}$$

where $\alpha_{L1}, \beta_{L1}, \gamma_{L1}$ are the coefficients of the planar constraint on **L1**. Transforming (7), α_{L1} and β_{L1} can be represented by unknown parameters $Z_{L1,1}$, $Z_{L1,2}$, and γ_{L1} as (8).

$$\alpha_{L1}(Z_{L1,1}, Z_{L1,2}, \gamma_{L1}) = \frac{f\left\{Z_{L1,1}(y_{L1,1} - p_y)(\gamma_{L1} - Z_{L1,1}) - Z_{L1,2}(y_{L1,2} - p_y)(\gamma_{L1} - Z_{L1,2})\right\}}{Z_{L1,1}Z_{L1,2}\left\{(x_{L1,1} - p_x)(y_{L1,2} - p_y) - (x_{L1,2} - p_x)(y_{L1,1} - p_y)\right\}}$$
$$\beta_{L1}(Z_{L1,1}, Z_{L1,2}, \gamma_{L1}) = \frac{f\left\{Z_{L1,1}(x_{L1,1} - p_x)(\gamma_{L1} - Z_{L1,1}) - Z_{L1,2}(x_{L1,2} - p_x)(\gamma_{L1} - Z_{L1,2})\right\}}{Z_{L1,1}Z_{L1,2}\left\{(x_{L1,2} - p_x)(y_{L1,1} - p_y) - (x_{L1,1} - p_x)(y_{L1,2} - p_y)\right\}} \tag{8}$$

After that, (9) is derived based on (5) and (8).

$$Z_{L1,i} = \frac{-f\gamma_{L1}}{\alpha\left(Z_{L1,1}, Z_{L1,2}, \gamma_{L1}\right)(x_{L1,i} - p_x) + \beta\left(Z_{L1,1}, Z_{L1,2}, \gamma_{L1}\right)(y_{L1,i} - p_y) - f} \tag{9}$$

In (9), unknown parameters $Z_{L1,i}$ are represented by unknown parameters $Z_{L1,1}$, $Z_{L1,2}$, and γ_{L1}. The final cost function is derived by substituting (9) into (6).

$$f(\mathbf{R}, \mathbf{t}, Z_{L1,1}, Z_{L1,2}, \gamma_{L1}) = \sum_{\mathbf{L1} \in \mathbf{L1_{all}}} \sum_{i \in \mathbf{L1}} \mathrm{dmin}\left(P(\mathbf{R}, \mathbf{t}, \mathbf{u_{L1,i}}, Z_{L1,1}, Z_{L1,2}, \gamma_{L1}), \mathbf{L2_{all}}\right) \tag{10}$$

In (10), the unknown parameters are $\mathbf{R}$, $\mathbf{t}$, $Z_{L1,1}$, $Z_{L1,2}$ and γ_{L1_n}. Since the $Z_{L1,1}$ and $Z_{L1,2}$ are defined as the depth of joints, the total number of unknown parameters for $Z_{L1,1}$ and $Z_{L1,2}$ is equal to the number of all the joints. And the number of unknown parameters for γ_{L1_n} is equal to the number of edge segments in $\mathbf{L1_{all}}$. Compared with conventional approach formulated by (6), the number of unknown parameter Z_i is dramatically reduced because generally the number of jonts and edge segments is much smaller than the total number of all the edge pixels in images.

Constraints on closest point search:

In the ICP algorithm described above, pixel $\mathbf{u_{L1,i}}$ is associated with the closest pixel on $\mathbf{L2_{all}}$, however in case of the dense edge lines, the closest pixel frequently becomes the false correspondence. To deal with this problem, constraints using epipolar geometry and line gradient are introduced. These constraint are used to narrow down the candidate pixels among $\mathbf{L2_{all}}$, then the closest pixel from $\mathbf{u_{L1,i}}$ is searched among the refined candidate pixels.

For the constraint using epipolar geometry, firstly an epipolar line for $\mathbf{u_{L1,i}}$ is drawn, then pixels in $\mathbf{L2_{all}}$ whose distance from the epipolar line exceeds a certain threshold are excluded.

Then, for the constraint using line gradient, firstly the gradient of edge line at $\mathbf{u_{L1,i}}$ is calculated, then pixels in $\mathbf{L2_{all}}$ whose gradient differ from that of $\mathbf{u_{L1,i}}$ by a certain threshold or more are excluded.

After the refinement through two constraints, $\mathbf{L2_{all}(u_{L1,i})}$ is defined as the set of refined candidate pixels. Substituting $\mathbf{L2_{all}(u_{L1,i})}$ into $\mathbf{L2_{all}}$ in (10), the cost function can be represented as in (11).

$$f(\mathbf{R}, \mathbf{t}, Z_{L1,1}, Z_{L1,2}, \gamma_{L1}) = \sum_{\mathbf{L1} \in \mathbf{L1_{all}}} \sum_{i \in \mathbf{L1}} \mathrm{dmin}\left(P(\mathbf{R}, \mathbf{t}, \mathbf{u_{L1,i}}, Z_{L1,1}, Z_{L1,2}, \gamma_{L1}), \mathbf{L2_{all}(u_{L1,i})}\right) \tag{11}$$

D. Outlier removal

Outlier detection is carried out on each edge segment, and if an edge segment is determined to be an outlier, all of the pixels constituting it are removed. The outlier detection is performed based on two approaches.

Residual check:

In the optimization process, the residual is evaluated at each edge pixel. If the average of the residuals in an edge segment exceeds the threshold, the edge segment is marked as outlier.

Consistency check:

The consistency of bi-directional motion parallaxes in consecutive images (image 1, image 2) is evaluated as follows. As shown in Fig. 5, if an motion parallax from image 1 to image 2 (f1(x1)) is correct, the motion parallax from image 2 to image 1 (f2(x1+f1(x1))) is expected to be a vector with opposite direction and the same length as f1(x1). Thus, the consistency value is defined as the absolute value of sum of bi-directional motion parallax. If the average of the consistency values for all the pixels in an edge segment exceeds the threshold, the edge segment is determined as outlier.

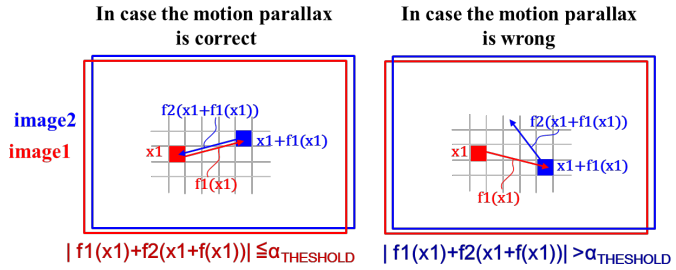

Fig. 5: Consistency check of bi-directional motion parallaxes in consecutive images. If the sum of two vectors of bi-directional motion parallaxes is far from zero, the optical flow is expected to be an outlier.

IV. Experiments

The performance on motion parallax and depth estimation was evaluated using KITTI dataset [13] which includes urban scenes where dense and complex edge lines exist. The ground truths of motion parallax and depth are given by the "optical flow evaluation 2015" [14] and the "depth prediction evaluation" [15] respectively. The input images are obtained from "raw data" [16], synchronized with the ground truth data. The evaluation is performed only on the pixels having both ground truth and estimation result. Additionally, the evaluation is performed only on the static region of the image using the label for moving objects in the KITTI dataset, since we are aiming at construction of static maps. Three methods were compared in the evaluation: our proposed method and two recent edge-based SFM techniques [8,9]. All the results in this evaluation were obtained from sets of two consecutive images, not using multi-view images for simplicity.

A. Motion parallax estimation

Fig. 6 shows a result of motion parallax estimation and Fig. 7 shows its error. In Fig. 7, the color of edge pixels indicates the error range, and gray color indicates the pixel having an estimated value but no ground truth value. Each method works well near the image center because generally the matching becomes much easier in a region where the motion parallax is sufficiently small. However, in case the motion parallax is increased, the matching becomes more difficult, especially with dense edge lines. In the region shown in (d)-(f) where dense edge lines with large motion parallax exist, the estimations using previous methods are degraded or invalidated. Meanwhile, the performance of the proposed method remains accurate.

Fig. 8 shows the average error of motion parallax estimation in the dataset. The vertical axis shows the occupancy rate of the pixels with an error within certain thresholds (1, 3, 5 pixels). As seen from Fig. 8, more than 95% of the motion parallaxes estimated with our method had error within 5 pixels, and more than 70% within 1 pixel. The graph shows that our method outperforms the previous ones. Table 1 shows the number of pixels for which estimation result has been obtained per image. The number of pixels with our method is greater than that of the method from Ramalingam et al[8] and nearly equal to the method from Tarrio et al[9]. This indicates that the availability of the estimation is not degraded with our method, while the accuracy is improved compared with the previous methods.

Fig. 6: Result of motion parallax estimation. (a)input image, (b)ground truth, (c)Tarrio et al[9], (d)Ramalingam et al[8], and (e)our method.

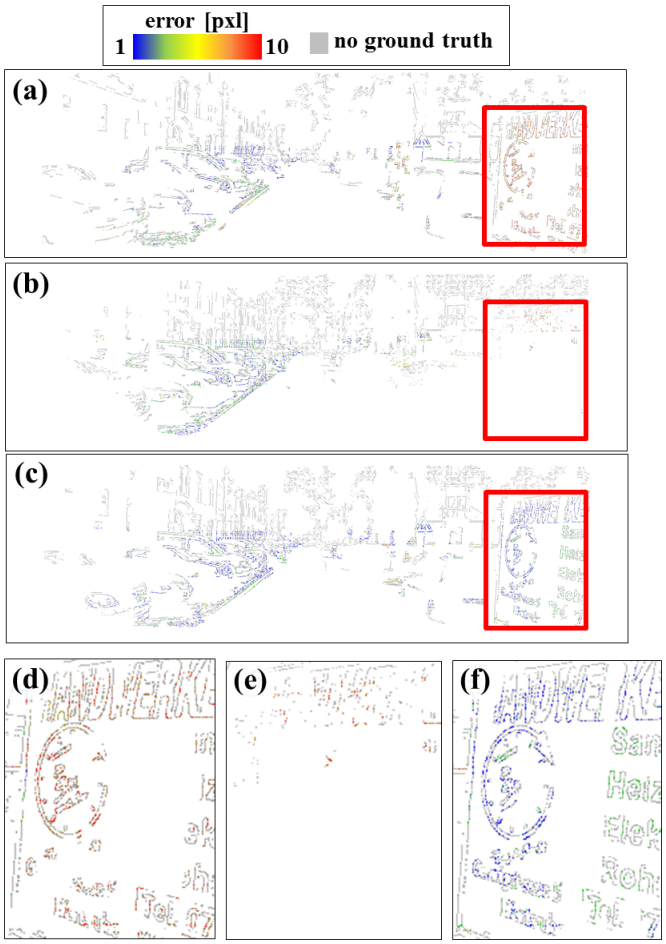

Fig. 7: Error of the motion parallax estimation resulting from (a)Tarrio et al[9], (b)Ramalingam et al[8], and (c)our method. (d)-(f) show the zoomed-up images of the red frame areas in (a)-(c), respectively.

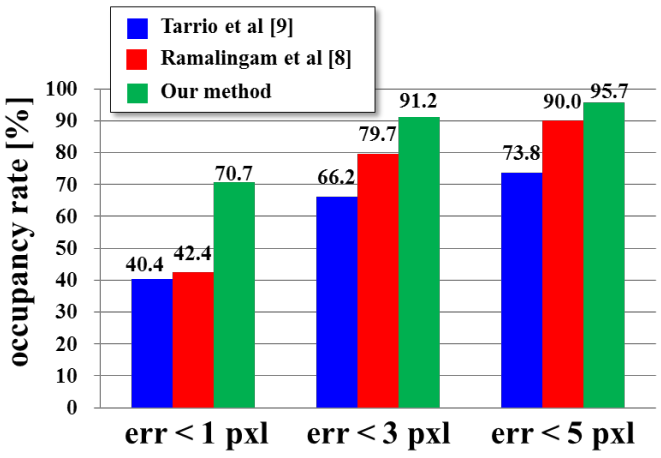

Fig. 8: Error evaluation of motion parallax estimation.

	Tarrio et al [9]	Ramalingam et al [8]	Our method
Number of pixels	3212	2795	3154

Table 1: Number of pixels per image for which the result of motion parallax estimation was obtained.

B. Depth estimation

Fig. 9 shows a result of depth estimation and Fig. 10 shows the error. In Fig. 10, the color of edge pixels indicates the range of depth error, and gray color indicates the pixel with an estimated value but no ground truth value. The unit of depth error is expressed in percentage, since it is normalized by true depth. As seen in Fig. 10, in the region shown in (d)-(f) where dense edge lines and large motion parallax exist, our method performs accurate estimation, while the estimations using previous methods are degraded or invalidated.

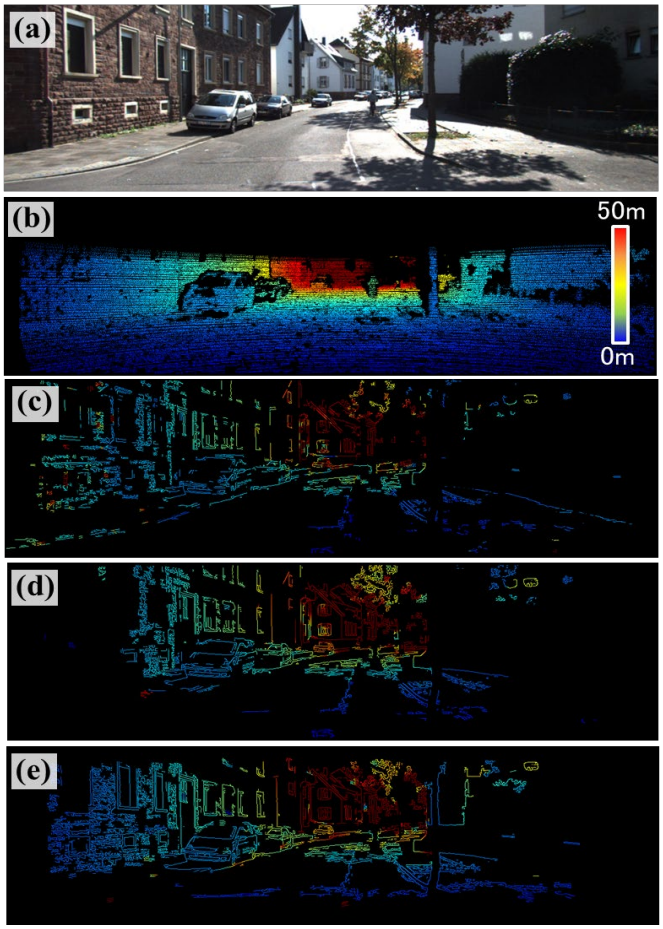

Fig. 9: Result of depth estimation. (a)input image, (b)ground truth, (c)Tarrio et al[9], (d)Ramalingam et al[8], and (e)our method.

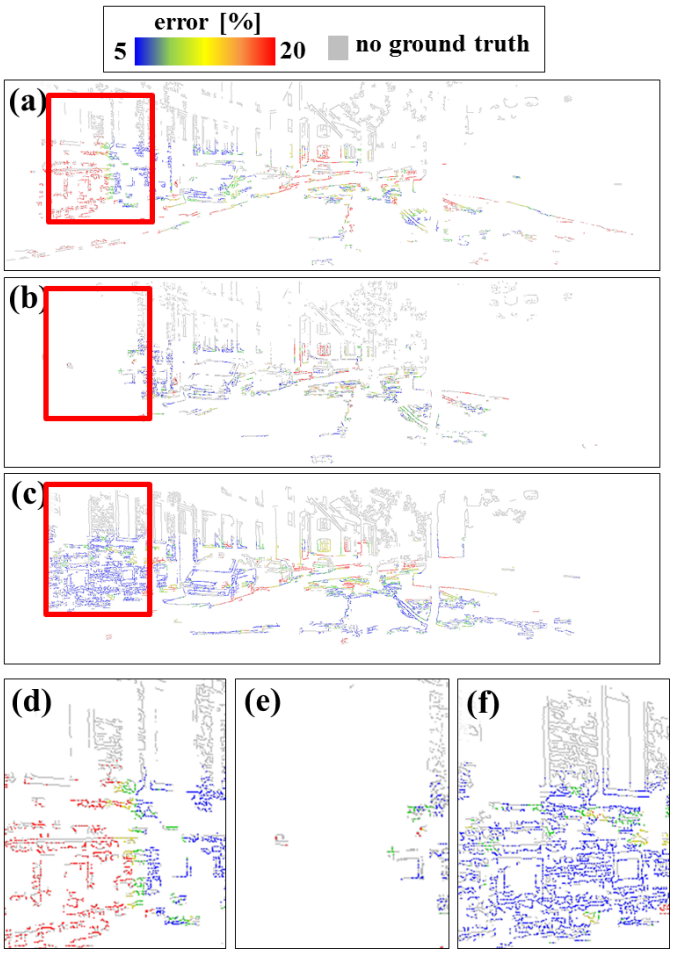

Fig.10: Error of the depth estimation resulting from (a)Tarrio et al[9], (b)Ramalingam et al[8], and (c)our method. (d)-(f) show the zoomed-up images of the red frame areas in (a)-(c), respectively.

Fig. 11 shows the average error of depth estimation in the dataset. The vertical axis shows the occupancy rate of the pixels with an error within certain thresholds (5, 10, 15 %). In Fig. 11, nearly 80% of the depth estimation on each pixel had an error within 15%, and more than 50% within 5%. Compared with the previous methods, the performance of our method is dramatically improved. Table 2 shows the number of pixels for which estimation result has been obtained per image. The number of available pixels with our method is larger than the previous methods. As a result, it is shown that both the accuracy and the availability of the depth estimation with our method are much improved compared with the previous methods.

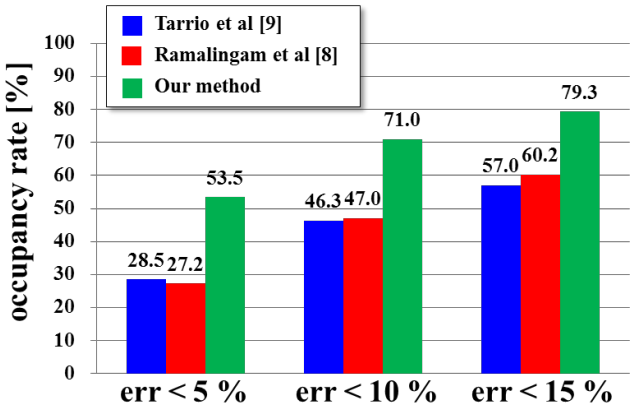

Fig. 11: Error evaluation of depth estimation.

	Tarrio et al [9]	Ramalingam et al [8]	Our method
Number of pixels	3585	3502	4055

Table 2: Number of pixels (per image) for which the result of depth estimation was obtained.

V. CONCLUSION

In this paper, a novel edge-based 3D reconstruction method for monocular camera has been proposed. The main contribution of this study is the improvement of the accuracy and the robustness of the 3D reconstruction in environments with dense edge lines by introducing geometric constraints to ICP algorithm. In our method, the overall stability of the matching is improved by introducing quasi-rigid body assumption to the edge line deformation, and also the accuracy of pixel correspondence in closest point search is improved by introducing constraints using epipolar geometry and line gradient. Experimental evaluation using KITTI dataset shows that our method outperforms two recent edge-based methods for both motion parallax estimation and depth estimation.

References

[1] C.Tomasi, and T.Kanade, “Shape and motion from image streams under orthography: a factorization method”, International journal of Computer Vision, No.9, vol.2, pp.137-154.

[2] G.Klein and D.Murray, “Parallel tracking and mapping for small AR workspace”, In Proc. of ISMAR, pp.225-234, 2007

[3] R.Mur-Artal, J.D.Tardos, “ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras”, IEEE Transactions on Robotics, Vol.33, pp.1255-1262, 2017

[4] R.Newcombe, S.Lovegrove, and A.Davidson, “DTAM: Dense tracking and mapping in real-time”, In Proc of ICCV, pp.2320-2327, 2011.

[5] J.Stühmer, S.Gumhold, and D.Cremers, “Real-time dense geometry from a handheld camera”, In Proc. of DAGM, 2010

[6] J.Engel, J.Sturm, and D.Cremers, “Lsd-slam: Large-scale direct monocular slam”, In Proc. of Computer ECCV, pp.834-849, 2014.
[7] J.Engel, J.Sturm, and D.Cremers, “Semi-dense visual odometry for a monocular camera”, In Proc. of ICCV, pp.1449-1456, 2013
[8] S.Ramalingam, M.Antunes, D.Snow, G.H.Lee, and S.Pillai, “Line-sweep: Cross-ratio for widebaseline matching and 3d reconstruction”, In Proc. CVPR（2015）.
[9] J.J.Tarrio, S.Pedre, “Realtime edge-based visual odometry for a monocular camera”, In Proc. of ICCV, 2015.
[10] C.Kim, P.Kim, S.Lee, and H.J.Kim, “Edge-Based Robust RGB-D Visual Odometry Using 2-D Edge Divergence Minimization”, In Proc. of IROS, 2018.
[11] X.Wang, D.Wei, M.Zhou, R.Li, and H.Zha, “Edge Enhanced Direct Visual Odometry”, In Proc. of BMVC, pp.35.1-35.11, 2016.
[12] J.Canny, “A computational approach to edge detection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.8, Issue.6, pp.679-698（1986）
[13] A.Geiger, P.Lenz, R.Urtasun, “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”, In Proc. CVPR（2012）.
[14] Optical Flow Evaluation 2015, “http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow”
[15] KITTI depth prediction, “http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow”
[16] KITTI Raw Data, “http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow”
[17] A.Gordon, H.Li, R.Jonschkowski, A.Angelova. "Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras." Proceedings of the IEEE International Conference on Computer Vision. 2019.