2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Edge-based Robust RGB-D Visual Odometry Using 2-D Edge Divergence Minimization

Changhyeon Kim, Pyojin Kim, Sangil Lee and H. Jin Kim

Abstract— **This paper proposes an edge-based robust RGB-D visual odometry (VO) using 2-D edge divergence minimization. Our approach focuses on enabling the VO to operate in more general environments subject to low texture and changing brightness, by employing image edge regions and their image gradient vectors within the iterative closest points (ICP) framework. For more robust and stable ICP-based optimization, we propose a robust edge matching criterion with image gradient vectors. In addition, to reduce a bad effect of outlier residuals, we propose an improved edge registration problem of 2-D edge divergence minimization in the manner of an iterative re-weight least squares (IRLS) motion estimation. To accelerate the proposed approach, a pixel sub-sampling method is employed. We evaluate estimation performance of our method in changing brightness conditions and low-textured scenes. Our approach shows more robust motion estimation than state-of-the-art methods while maintaining comparable accuracy in challenging image sequences at real-time (25 Hz) operation.**

I. INTRODUCTION

Camera motion estimation from consecutive images, known as visual odometry (VO) [1], is receiving increasing attention in areas of autonomous robots [13] and virtual and augmented reality (VR/AR) applications requiring self-localization abilities [3], [4]. A main interest of the VO research has been on improving the estimation accuracy while maintaining the real-time applicability. Consequently, several algorithms are developed with competitive performance in real-time applications using various settings, such as mono [5], [6], stereo [7], and RGB-D cameras [8], [9].

However, most VO algorithms still depend on two main assumptions; consistent brightness and feature-abundant scenes. The first one can be easily violated in cases of in-and-out movement and auto-exposure adjustments causing sudden brightness changes. Furthermore, texture-less scenes, such as monotonous walls and ceilings, make the second assumption invalid. If these assumptions are violated, the VO performance is significantly degraded and the VO might even lose track of the motion estimation. Thus, comprehensive consideration of robustness is required in order to make use of the VO in more general situations.

In this paper, we focus on enhancing the robustness of the VO against both low-textured and changing brightness circumstances. From the fact that image edge features can be observed more naturally and stably than points and lines in low-textured scenes, we utilize edge pixels of reference and current images to estimate successive camera motions. This kind of motion estimation approach registering a pair of point clouds, such as edge pixels, is called an iterative closest points (ICP) algorithm [10]. To bootstrap the ICP-based motion estimation, exact pixel correspondences between a pair of edges are required. If the camera motion is sufficiently small between two images, the correspondences can be efficiently determined by comparing the neighboring pixels of a pixel of interest. However, sufficiently small camera motions are not always guaranteed in general situations. Consequently, the vicinity test becomes vulnerable to wrong pixel matching and the overall estimation performance can be degraded considerably. To match the edge pixels regardless of large camera motions, we suggest a robust pixel matching criterion using image gradient vectors which are invariant to the changing absolute brightness values of images.

Different from other edge-based VO methods, we formulate a new cost minimization problem using a 2-D image edge divergence, which is a form of a signed distance residual and allows to use an iterative optimization method, such as the Levenberg-Marquardt algorithm [11]. In this framework, camera motion can be robustly recovered using an iterative re-weight least squares (IRLS) suppressing outlier residuals with a robust weight function. Furthermore, we improve the real-time applicability of our approach upto 25 Hz by sub-sampling edge pixels and adaptively changing parameter settings during motion estimation while maintaining accurate estimation.

Main contributions can be summarized as follows:

- Robust and fast edge-based VO against both irregular brightness changes and texture-less scenes, running at 25 Hz on a single CPU laptop setting.
- Improvement on the pixel matching rate in large motions by proposing a robust pixel matching criterion.
- Introduction of a new signed residual with 2-D image edge divergence, which enables a robust motion update using the IRLS framework.
- Extensive performance evaluations and comparisons with the state-of-the-art VO algorithms using TUM RGB-D benchmark [12] including low-textured and changing brightness sequences.

A. Related Works

General feature-based & direct VO: Most VO approaches can be divided into largely two streams; sparse feature-based and dense direct methods. The feature-based methods generally utilize a basic structure which was firstly suggested in [1]; extracting sparse image feature points, matching them, and finding camera motions

All authors are with Department of Mechanical and Aerospace Engineering, Seoul National University and Automation and Systems Research Institute (ASRI), Seoul, South Korea. {rlackd93,rlavywls,sangil07,hjinkim}@snu.ac.kr

by triangulations. These methods have been actively investigated in autonomous robots [13], [14] as well as computer vision applications, such as VR (virtual reality) and AR (augmented reality) [2], [3], due to the their intuitive framework and low computational costs.

On the other hand, in direct methods, almost all pixel brightness information is used to recover camera motions within a minimization problem of the sum of squared photometric error between two images [8]. The optimization-based framework of direct methods provides sub-pixel accuracy, however, they require heavy calculations to handle all pixel photometric information on images. With exponentially increasing computing power recently, several researchers have been motivated to develop direct methods [9], [15], [16] with an enhanced real-time capability. To combine both advantages of feature-based and direct methods, a hybrid approach [5] is suggested, which shows faster and more accurate performance.

Although they significantly improve the accuracy and real-time applicability, most of aforementioned VO systems are designed with two main assumptions: 1) consistent brightness and 2) availability of abundant features. Thus, they could yield worse performance in challenging conditions such as varying brightness and rare textures.

Robust VO: To relieve the limiting assumptions of VO, several works move the focus toward the robust VO systems. To avoid the first assumption, the globally-uniform illumination changes are considered [16], [17]. Patch-based VO is suggested with a linear illumination model to compensate local brightness changes in [17], and camera intrinsic factors affecting image illumination changes are considered to treat camera auto-exposure and vignetting effect in [16]. Although [17] shows robust performance even in the case of sudden changes of brightness, it still relies on feature-abundant scenes, and [16] has a difficulty to deal with locally irregular changes of brightness. Other works attempt to relieve the second assumption by utilizing more generalized features, such as lines and surface normal vectors [18], [19]. [18] blends lines with semi-direct visual odometry [5] to improve the stability in texture-less scenes, and [19] develops a stereo VO using multiple lines. Despite the consideration of robustness to low-textured scenes, [18] depends on the consistent illumination assumption because it is based on [5], and [19] relies on the sufficient number of straight lines.

Edge-based VO: There are several works utilizing edge features to operate VO in more general environments. In [20], the sum of squared distances of a pair of edges plays a part of an optimization constraint to prevent a direct method updating motions toward a wrong direction in low-textured scenes. [21] makes a very fast intensity-assisted ICP-based VO utilizing intensity patterns near edge pixels. Both works show robust performance in low-textured scenes due to an additional geometric constraint from edge pixels. Still, they use the absolute intensity information, therefore, their performance might be degraded in changing brightness conditions. Other edge-based approaches try to relieve several issues that occur when utilizing the image edges into the VO system. [22] gives attention to a non-differentiable nature of Euclidean distances widely used in the edge-based VO such as [20], [21], and relieves this issue by a sub-gradient method. In [23], an approximated nearest neighbor field is suggested to enable a time-efficient and non-parametric edge pixel matching. From the perspective of robustness, there is a still remaining limitation that [23] depends on absolute brightness information to extract semi-dense regions.

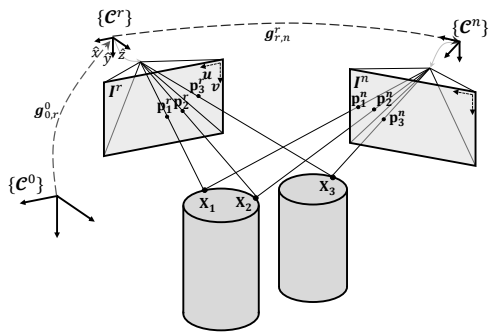


Fig. 1: **The geometric relationships among pixel points and camera frames.** The objective of the proposed approach is to find consecutive camera motions $g^r_{r,n}$ using image edge pixels.

In summary, there has not been comprehensive simultaneous consideration for both robustness to illumination changes and low-textured scenes.

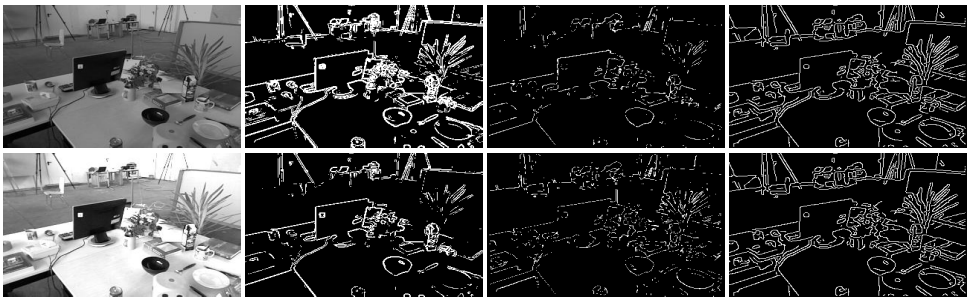
II. Preliminaries

We introduce notation rules referred throughout the paper and describe geometric relationships of consecutive cameras and edge pixel coordinates in 3-D space. We define all vectors as column vectors and write them in bold. Let $\mathcal{C}^n$ denote the n-th camera coordinate frame and a superscript $(\cdot)^n$ denotes a variable represented in the n-th camera coordinate frame. Especially, $(\cdot)^0$ means an inertial reference frame which is equal to the first camera coordinate frame. In addition, a single-subscript $(\cdot)_i$ refers to the i-th element of a set of specific variables. To describe an inter-frame 3D rigid body motion of cameras, let a double-subscript $(\cdot)_{r,n}$ mean a camera motion from the r-th frame to the n-th frame. With these notations, we develop a camera model and a parametrization of rigid body motions among cameras.

A. Camera Model and 3-D Motion among Cameras

A value of a gray-scale image and its depth map on the i-th 2-D image pixel coordinate of the k-th camera frame $\mathbf{p}_i^k \in \mathbb{R}^{2\times 1}$ from a RGB-D camera at a specific time epoch n are represented as $I^k\left(\mathbf{p}_i^k\right)$ and $D^k\left(\mathbf{p}_i^k\right)$, respectively. The relationship between $\mathbf{p}_i^k$ and corresponding 3D point $\mathbf{X}_i^k$ represented in the k-th camera coordinate frame can be represented by a perspective camera projection $\mathbf{p}_i^k = \pi\left(\mathbf{X}_i^k\right)$. Its inverse map re-projecting the $\mathbf{p}_i^k$ into the 3D space at epoch n is denoted as $\mathbf{X}_i^k = \pi^{-1}\left(\mathbf{p}_i^k, D^k\left(\mathbf{p}_i^k\right)\right)$.

Additionally, we define each element of an image gradient on $\mathbf{p}_i^k$ along each axis of a pixel coordinate frame as $G_u^k\left(\mathbf{p}_i^k\right)$ and $G_v^k\left(\mathbf{p}_i^k\right)$, respectively, and their vector forms can be represented as $\mathbf{G}^k\left(\mathbf{p}_i^k\right)$ and be achieved by Sobel operators.



(a) (b) (c) (d)

Fig. 2: **Results of edge extractions.** (a) images with different brightness, (b) Sobel edge algorithm, (c) Difference of Gaussian, (d) Canny algorithm.

We use a Lie group $g^r_{r,n} \in SE(3)$ and a Lie algebra $\xi^r_{r,n} \in se(3)$ to describe a relative 3D motion from the r-th to the n-th camera coordinate frames. We also define a warping function transferring $\mathbf{p}^n_i$ into a corresponding pixel coordinate represented in the r-th pixel coordinate,

$$w\left(\xi^r_{r,n}, \mathbf{p}^n_i\right) = \pi\left(g\left(\xi^r_{r,n}\right) \cdot \left[\pi^{-1}\left(\mathbf{p}^n_i, D^n\left(\mathbf{p}^n_i\right)\right)^{\mathrm{T}}, 1\right]^{\mathrm{T}}\right), \tag{1}$$

where we expand the projection function π to a homogeneous coordinate of 3-D point.

B. ICP-based Camera Motion Estimation

To utilize edge pixels for the camera motion estimation, we first have to find correspondences between reference and current edge pixels. We can determine the most probable matching candidate of a pixel coordinate $\mathbf{p}^n$ of the current n-th image frame by searching the nearest pixel coordinate among a set of reference pixels $\mathcal{R}$, i.e. by Nearest Neighbor (NN) searching [10]. We define a function yielding the nearest pixel of the current pixel coordinate $\mathbf{p}^n$ on the reference frame as,

$$near\left(\xi^r_{r,n}, \mathbf{p}^n\right) \in \mathcal{R}. \tag{2}$$

We can search the nearest pixel by minimizing a distance function $dist\left(\cdot\right)$ as below:

$$near\left(\xi^r_{r,n}, \mathbf{p}^n\right) = \underset{\mathbf{p}^r \in \mathcal{R}}{\arg\min}\, dist\left(w\left(\xi^r_{r,n}, \mathbf{p}^n\right), \mathbf{p}^r\right). \tag{3}$$

In most NN searching algorithms, the distance function is commonly defined as a form of 2-norm of the pixel Euclidean distance [10]. If the point sets of reference and current images are sparse and the camera motion is sufficiently small, using the pixel Euclidean distance is enough to find correct correspondences. However, according to [21] and our experiences, using it solely does not give a reliable matching when the point sets are complexly distributed and the sensor motion is relatively large. We will closely discuss it and relieve the issue by proposing a robust edge pixel matching criterion with image gradient vectors in the following sections.

The core concept of the ICP-based motion estimation is to find a rigid body motion $\xi^r_{r,n}$ minimizing a sum of edge pixel distances of N number of matched pairs of edge pixel coordinates. The objective function E can be written as

$$E = \mathbf{d}^T\mathbf{d} = \sum_{i=1}^{N} d_i^2. \tag{4}$$

The i-th element d_i of the 2-norm distance vector $\mathbf{d} \in \mathbb{R}^{N\times 1}$ is

$$d_i = \| w\left(\xi^r_{r,n}, \mathbf{p}^n_i\right) - near\left(\xi^r_{r,n}, \mathbf{p}^n_i\right) \|, \tag{5}$$

and the camera motion $\xi^r_{r,n}$ can be obtained by minimizing E with respect to $\xi^r_{r,n}$.

If the correspondences are correctly established, the minimization of the objective E can be simply performed in a closed-form by the Singular Value Decomposition (SVD) as noted in [10]. However, wrong matchings inevitably emerge and mislead overall estimation into wrong local minima. Thus, the problem is naturally converted into the IRLS optimization to suppress the bad influence of outliers via robust residual weighting [9], [21]. As mentioned in [22], [23], however, the sum of Euclidean distances in Eq. 4 is not a proper form of the residual for gradient-based optimization, such as the Levenberg-Marquardt algorithm, due to the non-negativeness of the Euclidean distance. We circumvent this issue via an idea of 2-D edge divergence minimization, which invokes a signed residual proper for gradient-based approaches. More detailed descriptions about how to relieve the mentioned issues are explained in the following sections.

III. Robust Visual Odometry minimizing 2-D Edge Divergence

This section explains the entire framework of the proposed robust visual odometry using the 2-D image edge divergence minimization.

A. Image Edge Region Extraction

Several VO algorithms make use of image edges for the camera motion estimation [20], [22]. In those works, a main reason for selecting edge features is the fact that edges can be more naturally observed than points and lines even in low-textured scenes. For similar reasons, we intend to utilize the robust characteristic of image edges.

There exist several edge extraction methods, such as Sobel edge algorithm, zero-crossing Difference of Gaussian (DoG) method, and Canny algorithm. Edge extraction results with two different illumination settings are shown in Fig. 2.

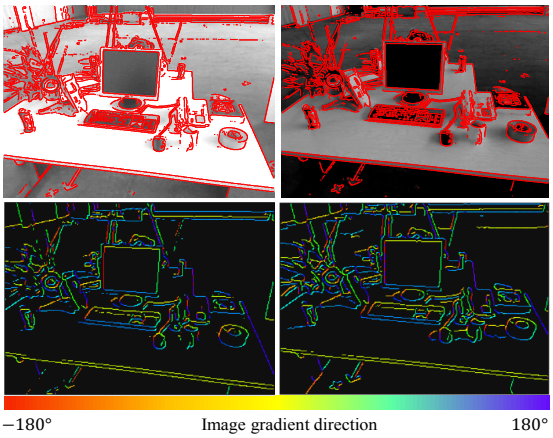


Fig. 3: **Directions of image gradient vectors on edge pixels visualized with colors.** As can be seen in lower images, image gradient directions are clearly and consistently observed regardless of illumination changes. Especially, the most cluttered region near the plant can be characterized by image gradient vectors well.

The results of the Canny algorithm show the most reliable performance among three methods because it locally finds the strongest edge pixels by non-maximum suppression of high gradient regions. Furthermore, the Canny algorithm shows robustness against brightness changes using a double-thresholding scheme for finding edge pixels while the others are not consistent because they use a single threshold value for edge detection, which is vulnerable to changes of the absolute brightness. Based on this analysis, we can conclude the Canny algorithm is the most proper edge extractor for the proposed VO framework.

B. Robust Edge Pixel Matching using Image Gradient Vectors

One of the key parts of the proposed algorithm is to find correct edge pixel correspondences, i.e. NN searching, between reference and current frames. As already noted in section II-B, the Euclidean distance is widely used to measure how similar a current point is with reference points in various ICP-based VO algorithms [10], [20]. It is straightforward to implement and shows reasonable performance in the case of sufficiently small camera motions and sparse point sets like [10]. But if the edge pixel distribution is complex and dense, wrong matchings often emerge because there can be edge pixels closer than the correct candidate. While points and lines can be clearly distinguished and matched by feature descriptors, no apparent descriptor for edges has been proposed yet.

To accurately find the edge correspondences in large motions and complex scenes, we suggest a robust criterion to distinguish and match edge pixels using image gradient directions. In accordance with the fact that edge regions have prominent image gradient magnitudes and regularly aligned directions as depicted Fig. 3, we consider not only the pixel 2-D Euclidean distance but also inner products of image gradient directions on the edge pixels when calculating the distance function in Eq. 3. The augmentation of image gradient directions can be an effective edge descriptor to distinguish each pixels.

We propose a new distance function as

$$dist(\mathbf{p}^r, \mathbf{p}^n) = \frac{\|\mathbf{p}^r - \mathbf{p}'\|^2}{width^2} + \frac{\gamma}{4}\left\| \frac{\mathbf{G}^r(\mathbf{p}^r)^T}{\|\mathbf{G}^r(\mathbf{p}^r)\|} \frac{\mathbf{G}^n(\mathbf{p}^n)}{\|\mathbf{G}^n(\mathbf{p}^n)\|} - 1 \right\|^2, \tag{6}$$

where p' is equivalent to the warped pixel point $w\left(\xi^r_{r,n}, \mathbf{p}^n\right)$. Eq. 6 is a linear combination of two terms, the Euclidean distance and inner product of image gradient vectors of a pair of edge pixel coordinates, and the ratio of contributions of each terms is adjusted by a scale factor γ. We heuristically select $\lambda = 0.7$ which shows the best matching results among various evaluations.

The proposed NN searching can be efficiently conducted by using a data tree structure to quickly narrow down a searching space. In particular, we employ a balanced k-d tree structure [24] and modify a correspondence evaluation term of the k-d tree as the form of Eq. 6 to fit the four-dimensional edge pixel data composed of a 2-D pixel coordinate and a 2-D image gradient vector.

C. 2-D Edge Divergence Minimization for Motion Estimation

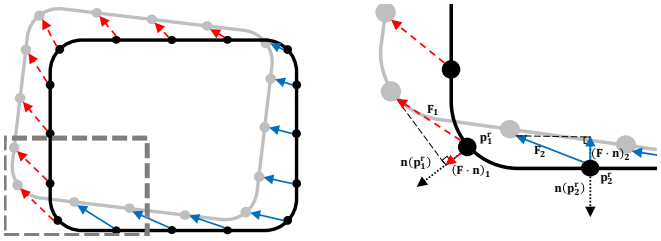
As mentioned in Section. II-B, there exist inevitable wrong pixel matchings and they have a serious influence on the overall estimation performance because outlier residuals induced by wrong matchings quadratically affect on the objective function in Eq. 4. Accordingly, the motion estimation problem is naturally converted to an iterative cost minimization problem using IRLS optimization. In this way, the motion estimation problem can be elegantly formulated as a form of a graident-based optimization, such as the Levenberg-Marquardt algorithm, and robust re-weighting methods [25] can be also applied to suppress the effects of outliers as [9].

In direct optimization-based approaches [20], [21], the sum of Euclidean distances of edge pixels is combined with intensity residuals as a geometric constraint not to let direct methods diverge in low-textured circumstances. However, if the Euclidean distance is solely used as the residual within the gradient-based optimization framework, the positive semi-definite nature of the Euclidean distance leads the algorithm to a biased update direction, which does not guarantee convergence toward a correct optimum.

In this paper, we propose a signed residual via a concept of 2-D edge divergence minimization in place of the Euclidean distances. Imagine a set of disparity vectors induced by subtraction of edge pixel correspondences as a vector field acting along whole streak of black-colored image edge regions as illustrated in Fig. 4a. With the vector field, we can define a 2-D divergence relationship in the sense of continuous line integral :

$$\int_C \mathbf{F} \cdot \mathbf{n}\, ds = \iint_V div\, \mathbf{F}\, dV \tag{7}$$

where $\mathbf{n}$ denotes a geometric normal vector of the reference edge region, and C and V represent sets of lines composed of all reference edge pixels and areas surrounded by edges,



(a) Misaligned edge regions. (b) Flux on reference edges.

Fig. 4: **Illustration of 2-D edge divergence**. (a) The black rectangle is a reference edge pixels and gray one is a current edge pixels. Red arrows mean outgoing vector fields of reference edge lines and blue ones denote incoming vector fields. (b) The detailed view of the gray dotted region. Flux along the reference edges can be calculated by inner product of vector fields $\mathbf{F}$ and edge normal vectors $\mathbf{n}$.

respectively. Assuming static scenes and rigid objects, two edges are exactly registered after the convergence of the motion estimation. It means that the ideal state of the edge registration is equivalent to the zero value of $div\,\mathbf{F}$ with respect to every edge pixels. Then, Eq. 7 becomes

$$\int_C \mathbf{F}\cdot\mathbf{n}\,ds = 0. \tag{8}$$

Due to the discrete nature of image pixels, the left-hand-side of Eq. 7 becomes the following summation form :

$$\sum_{i=1}^{N} \mathbf{F}_i\cdot\mathbf{n}_i\,ds = 0 \tag{9}$$

where $\mathbf{F}_i \in \mathbb{R}^{2\times 1}$ is the 2-D vector field and $\mathbf{n}_i \in \mathbb{R}^{2\times 1}$ is the geometric normal vector of edge regions, respectively.

Because the infinitesimal length ds cannot be zero, the inner product of $\mathbf{F}_i$ and $\mathbf{n}_i$ necessarily goes to zero at the end of the optimization. According to Fig. 4, $\mathbf{F}_i$ is denoted by

$$\mathbf{F}_i = w\left(\xi^r_{r,n}, \mathbf{p}^n_i\right) - near\left(\xi^r_{r,n}, \mathbf{p}^n_i\right), \tag{10}$$

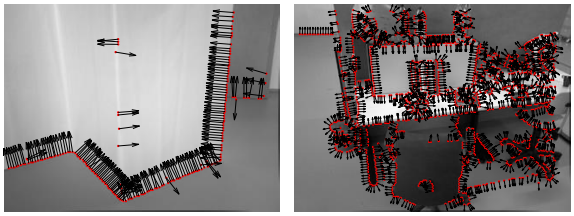
and $\mathbf{n}_i$ can be approximated by normalized image gradient vectors,

$$\mathbf{n}_i \approx \frac{\mathbf{G}^r(\mathbf{p}')}{\|\mathbf{G}^r(\mathbf{p}')\|}. \tag{11}$$

where $\mathbf{p}'$ is a nearest reference edge pixel of $\mathbf{p}^n_i$. Although image gradient vectors do not exactly represent geometric edge normal vectors $\mathbf{n}$, the approximation works appropriately due to two reasons: image gradient vectors are almost orthogonal to the streak of edge pixels as depicted in Fig. 5, and the role of geometric edge normal vectors are no more than the fixed reference vector to project F_i. Consequently, we can define the new residual vector $\mathbf{r}$ composed of N pairs of edge pixel correspondences, and the i-th element of $\mathbf{r}$ can be represented as below,

$$r_i = \mathbf{F}_i \cdot \frac{\mathbf{G}^r(\mathbf{p}')}{\|\mathbf{G}^r(\mathbf{p}')\|} \tag{12}$$

where $\mathbf{p}'$ is the nearest reference edge pixel point of currently interesting $\mathbf{p}^n_i$. Note that, entire procedure of making this residual totally avoids using absolute intensity values while other edge-based systems, such as [20] and [21], rely on



(a) (b)

Fig. 5: **Image gradient directions on edge regions.** (a) low-textured scene, (b) texture-abundant scene.

absolute intensity information to match the edge pixels and implement the motion estimation.

The main objective is to find the camera motion by minimizing the 2-D edge divergence :

$$\xi^* = \underset{\xi\in SE(3)}{\arg\min}\,\mathbf{r}^{\mathrm{T}} W \mathbf{r} \tag{13}$$

where W is a diagonal residual weight matrix. Determining the weight matrix will be discussed in the next section.

Let $J = \frac{\partial \mathbf{r}}{\partial \xi} \in \mathbb{R}^{N\times 6}$ be a Jacobian matrix of the residual vector with respect to ξ, then, the motion update $\Delta\xi$ is calculated as

$$\Delta\xi = -\left(H + \lambda diag\left(H\right)\right)^{-1} J^T W \mathbf{r}. \tag{14}$$

where the matrix $H = J^T W J \in \mathbb{R}^{6\times 6}$ is a weighted Hessian matrix and λ is a damping coefficient which scales the effect of diagonal damping term $diag\left(H\right)$.

To summarize, the proposed algorithm avoids any use of absolute intensity information by proposing two key components: the robust edge matching criterion and 2-D edge divergence minimization. They allow the proposed VO algorithm to run in challenging circumstances with both changing illumination and low-textured scenes.

IV. Implementation Details

We present detailed consideration on how to make the algorithm more time-efficient and accurate in this section.

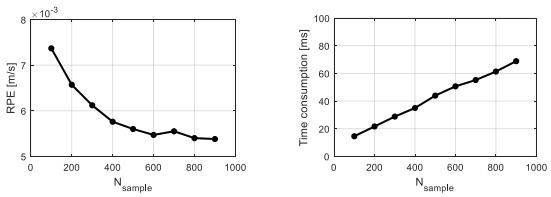
A. Sub-sampling of Edge Pixels

For real-time applicability, we examine average time consumption of each part of the proposed method. As can be seen in the left column of Table. I, the pixel matching step is the very bottleneck of the algorithm. This issue comes from thousands of queries to k-d tree to find the nearest neighbors.

To reduce the time consumption by the pixel matching, we sub-sample the reduced N_{sample} number of current edge points. Similar to feature-based methods [6], regularly distributed samples across the image are crucial for the stable

TABLE I: Average time consumption by each component of the proposed method without and with sub-sampling on the TUM RGB-D *fr3/long*.

Components	Time(w/o) [ms]	Time(w/) [ms]
Edge information extraction	2.97	2.91
Build k-d tree	3.54	3.62
Robust pixel matching	300.95	30.84
Update motions	31.13	3.54
Total	338.59	40.91



(a) (b)

Fig. 6: **Effect of the number of samples on the performance.** (a) the number of samples vs. relative pose errors, (b) the number of samples vs. averaged time consumption per frame. Each value is obtained by implementing the proposed VO with *fr3/long*.

and accurate operation. To equally spread samples, we divide a whole image into grids and sample at least five pixels per grid. Then, the remaining number of points are randomly selected from the entire grids.

We evaluate the effect of different N_{sample} on performance and results are depicted in Fig. 6. It shows clear trade-off between accuracy and time consumption. However, increasing N_{sample} does not always guarantee the performance improvement. According to tendencies of graphs, we can conclude that N_{sample} between 400 and 500, corresponding to 25 Hz frequency, is a reasonable selection to satisfy both accuracy and real-time applicability.

B. Robust Weight Function for Suppressing Outliers

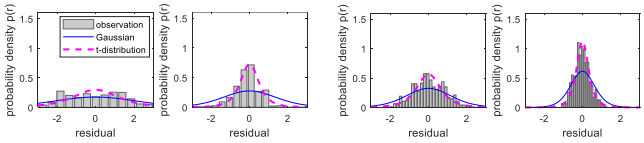
During the optimization, outlier residuals inevitably occur due to wrongly matched pixels. Since outliers can mislead the whole estimation to a wrong update direction, residuals abnormally larger than the majority have to be suppressed. Inspired by [9], we employ the t-distribution to suppress outliers. To fit the distribution, we first check the distribution of the residual vector. As depicted in Fig. 7, residual distributions of two datasets are quite different in terms of a steepness in contrast to [9]. For the best fitting result, we heuristically found that the best value for the degree of freedom ν of the t-distribution is two. As can be seen in Fig. 7, the fitting results of the t-distribution are superior to the Gaussian distribution. We can update a scale parameter σ that determines the form of the t-distribution within a few iterative calculations as described in [9].

V. Experimental Results

We now evaluate the performance of the proposed method by varying algorithm settings, and extensively compare ours with state-of-the-art VO algorithms. We use the TUM RGB-D benchmark dataset [12], which is publicly available and widely used to compare the performance of RGB-D VO.

A. Datasets and Experimental Settings

Each dataset of the TUM RGB-D benchmark consists of VGA resolution RGB color and depth images at 30 Hz from Microsoft Kinect camera, and 100 Hz ground truth data of motion capture systems. To evaluate robustness to irregular illumination, we use two types of datasets with and without synthetic illumination changes. To obtain realistically synthesized images, we employ two illumination models widely-used in computer graphics, such as ambient and



(a) (b)

Fig. 7: **Residual distributions of two datasets with respect to iterations.** (a) low-textured case (*fr3/ntxt/str/far*), (b) texture-abundant case (*fr3/long*). The two distributions in each case are observed at the 3-th and 20-th iterations, respectively.

diffusive models suggested in [26]. The first one is related to uniform illumination changes and the other is about locally-gradual changes induced by an oblique lighting source. The illumination varies at 2 Hz frequency and we intendedly add sudden illumination changes at several time instances, which mimics the camera auto-exposure. The representative images of the synthesized datasets are depicted in Fig. 11. All calculations are conducted on a laptop setting with Intel Core i7-7500U at 2.7 GHz with 8GB memory.

B. Performance 1 : Robust Edge Matching Criterion

To confirm the improvement by the proposed matching method, we evaluate the matching success rate defined as

$$success\ rate = \frac{N_{correct}}{N_{total}} \tag{15}$$

where N_{total} is a total number of pixels and $N_{correct}$ is the number of correctly matched pixels between original and warped images, and analyze the overall VO performance. Because the large motion lowers the matching success rate, we warp the original image along 6-DoF motions ξ and evaluate the matching success rates. According to Fig. 8, our approach shows several times higher success rates than the normal matching method that checks only spatial vicinities of pixels in all cases.

Note that the matching success rate of the proposed method is slightly lower than the normal criterion when the motion is near zero. It stems from the fact that the image gradient directions between two images are not exactly same due to the difference of view points. With these heuristics, we use the proposed method to lead the update to the correct direction when the motion gap between two images is large at first, and after the motion gap becomes sufficiently small, we use the normal criterion to improve the matching rate near the zero motion.

To show the benefit of using the proposed matching method, we run the proposed VO with two matching criteria. The results in Fig. 9 show that the proposed method maintains a steady level of iteration number while the number of iterations often soars to an intractable level when using the normal criterion solely. This can be attributed to the fact that the VO with the proposed matching method has higher matching rates than the VO with the normal method that leads to wrong local minima due to a small number of correct matching pairs in image sequences with large camera motions.

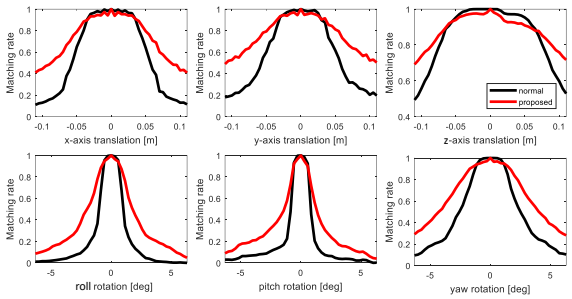


Fig. 8: **Comparison of edge pixel matching success rates of two pixel-matching methods along 6-DoF rigid body motions.** Black and red colors denote the normal matching method and the proposed robust matching method, respectively.

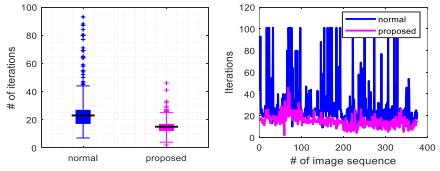


Fig. 9: **The improved stability of whole algorithm by the robust edge matching criterion.** The average number of iterations is reduced from 28.8 with 15.1 standard deviation to 18.7 with 5.42 standard deviation.

C. Performance 2 : Frame Distance between Reference and Current Images

Because drifts incrementally accumulate in every change of the reference image, the *key-frame method* fixing the reference image during several image inputs is actively employed in many VO systems [5], [6]. In our method, we also adopt this method to reduce the drift accumulation. When using it in our VO system, the interval of replacing the reference image has to be carefully analyzed because the pixel matching rate directly related to the algorithm stability sensitively responds to the large spatial gap between reference and current images, as discussed in Section. V-B.

We investigate the estimation performance and the algorithm stability with various frame gaps by changing the pixel matching methods. The results in Fig. 10 imply that the larger gap generally guarantees the lower drifts. Note that, in the case of the normal matching method, drifts are less reduced than the proposed matching method because the low matching rate of the normal method causes the insufficient motion estimation when frame gaps are large. Additionally, the number of iterations of the normal method abnormally surges due to its low matching rate while the proposed matching method shows more consistent and smaller number of iterations regardless of incremental frame gaps. Hence, we can conclude that the proposed matching method gives a larger basin of convergence and improves the further performance via the *key-frame method*.

D. Evaluation of the Overall Estimation Performance

We extensively compare the motion estimation performance of the proposed algorithm with open-source state-of-the-art VO algorithms: two feature-based methods (SVO [5] and ORB-SLAM [6]) and two direct methods (DVO [8] and DSO [16]). For fair comparison in terms of pure VO, we disable the re-localization functionalities of SVO and ORB-

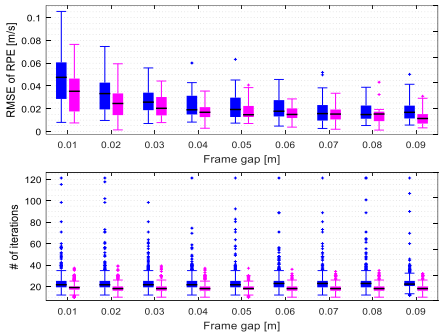


Fig. 10: The top graph shows a relationship between drifts and frame, and the bottom graph between iterations and frame gaps. Blue and magenta colors denote the normal method and the proposed matching method, respectively.

SLAM. We employ the relative pose error (RPE) and its median value as error metrics suggested in [12]. We mark the best performance of each sequence using a bold letter in Table. II and III, and the cross marks mean the failed cases including a severe divergence.

First, we compare the estimation performance without illumination changes. In texture-abundant scenes, corresponding to the first three rows of Table. II, the proposed algorithm shows similar performance with state-of-the-art algorithms. On the other hand, feature-based methods fail to build the initial feature map in texture-less scenes while the edge-based methods including our method can stably operate. Especially, SVO fail to track the sufficient feature at 948-th image of $fr3/long$ when the features disappear due to sudden closing up. Note that DSO and DVO show motion jump problems at several low-textured images of $fr3/str/ntxt/far$ and $fr3/str/ntxt/near$.

In the cases of illumination changes, ORB stably runs maintaining the similar performance with the consistent illumination cases. Other algorithms except for ours, however, degrade and fail regardless of whether there are many textures in scenes or not. Particularly, despite the feature-based method, SVO fails to initialize even in texture-abundant scenes because it directly exploits the absolute illumination information. By contrast, the proposed VO algorithm can robustly operate in all challenging sequences with the competitive performance.

VI. CONCLUSIONS

In this paper, we proposed the edge-based robust RGB-D visual odometry using 2-D edge divergence minimization. Our approach was targeted to be operated in more general environments, such as low-textured scenes and changing brightness conditions, by utilizing image edges and their image gradient vectors. For more robust and stable ICP-based optimization, we proposed the robust edge matching criterion with image gradient vectors. Additionally, we suggested the idea of 2-D edge divergence minimization to enable the iterative re-weight least squares (IRLS) motion estimation problem. We evaluated the estimation performance using TUM RGB-D datasets with varying brightness conditions and low-textured scenes. Our approach showed the most

Fig. 11: **The representative images used in evaluations.** Texture-abundant scenes: (a) $fr3/txt/str/far$, (b) $fr3/txt/str/near$, (c) $fr3/long$, and low-textured scenes: (d) $fr3/ntxt/str/far$, (e) $fr3/ntxt/str/near$, (f) $fr3/cabinet$. The first and second rows have time difference of 1 second, corresponding to the half period of changing illumination.

TABLE II: Comparisons of performance using TUM RGB-D datasets without illumination changes.

	RMSE of drift (Relative Pose Error) [m/s]					Median of drift (Relative Pose Error) [m/s]				
	SVO	ORB	DVO	DSO	Ours	SVO	ORB	DVO	DSO	Ours
fr3/txt/str/far	0.027	0.020	0.000	**0.015**	0.016	0.025	0.016	0.000	0.014	**0.013**
fr2/txt/str/near	0.038	0.018	0.018	**0.014**	0.015	0.030	0.013	0.015	**0.010**	0.010
fr3/long	0.020	0.023	0.024	**0.017**	0.021	0.016	0.017	0.018	0.013	**0.011**
fr3/ntxt/str/far	×	×	0.311	0.109	**0.012**	×	×	0.305	0.112	**0.009**
fr3/ntxt/str/near	×	×	0.367	0.277	**0.033**	×	×	0.219	0.151	**0.016**
fr3/cabinet	×	×	0.145	×	**0.068**	×	×	0.130	×	**0.043**

TABLE III: Comparisons of performance using TUM RGB-D datasets with illumination changes.

	RMSE of drift (Relative Pose Error) [m/s]					Median of drift (Relative Pose Error) [m/s]				
	SVO	ORB	DVO	DSO	Ours	SVO	ORB	DVO	DSO	Ours
fr3/txt/str/far	0.062	0.020	0.105	×	**0.019**	0.062	0.017	0.088	×	**0.014**
fr2/txt/str/near	0.059	0.018	0.085	×	**0.014**	0.049	0.011	0.064	×	**0.009**
fr3/long	0.041	0.021	0.582	0.031	**0.019**	0.025	0.016	0.352	0.018	**0.010**
fr3/ntxt/str/far	×	×	×	×	**0.012**	×	×	×	×	**0.010**
fr3/ntxt/str/near	×	×	×	×	**0.037**	×	×	×	×	**0.019**
fr3/cabinet	×	×	×	×	**0.067**	×	×	×	×	**0.036**

robust performance among state-of-the-art methods in challenging image sequences in real-time operation at 25 Hz.

To relieve the static environment assumption required by the 2-D divergence minimization, the future work will deal with more general scenes including dynamic objects.

ACKNOWLEDGMENT

This work was supported by SAMSUNG Research, Samsung Electronics Co.,Ltd. and the Ministry of Trade, Industry & Energy(MOTIE, Korea) under the Industrial Technology Innovation Program(No.10067206).

REFERENCES

[1] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in IEEE CVPR, 2004.
[2] T. Schops, J. Engel, and D. Cremers, "Semi-dense visual odometry for ar on a smartphone," in IEEE ISMAR, 2014.
[3] G. Klein, and D. Murray. "Parallel tracking and mapping for small ar workspaces," in IEEE ISMAR, 2007.
[4] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. "DTAM: dense tracking and mapping in real-time," in IEEE ICCV, 2011.
[5] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: fast semi-direct monocular visual odometry," in IEEE ICRA, 2014.
[6] R. Mur-Artal, and J. D. Tardos, "ORB-SLAM2: an open-source slam system for monocular, stereo, and rgb-d cameras," IEEE T-RO, vol. 33, no. 5, 2017, pp. 1255-1262.
[7] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," IEEE IV, 2011.
[8] F. Steinbrucker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense RGB-D images," in IEEE ICCV Workshop, 2011.
[9] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in IEEE ICRA, 2013.
[10] I. Dryanovski, R. G. Valenti, and J. Xiaom, "Fast visual odometry and mapping from rgb-d data," in IEEE ICRA, 2013.
[11] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," Journal of the society for Industrial and Applied Mathematics, vol. 11, no.2, 1963, pp. 431-441.
[12] J. Sturm, et al. "A benchmark for the evaluation of rgb-d slam systems," in IEEE IROS, 2012.
[13] A. S. Huang, et al., "Visual odometry and mapping for autonomous flight using an rgb-d camera," in ISRR, 2011, pp. 1-16.
[14] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the Mars exploration rovers," in Journal of Field Robotics, vol. 24, no. 3, 2007, pp. 169-186.
[15] J. Engel, T. Schops, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in ECCV, 2014.
[16] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," IEEE T-PAMI, 2017.
[17] P. Kim, H. Lim, and H. Jin Kim, "Robust visual odometry to irregular illumination changes with rgb-d camera," in IEEE IROS, 2015.
[18] R. Gomez-Ojeda, J. Briales, and J. Gonzalez-Jimenez, "Pl-svo: Semi-direct monocular visual odometry by combining points and line segments," in IEEE IROS, 2016.
[19] J. Witt and U. Weltin, "Robust stereo visual odometry using iterative closest multiple lines," in IEEE IROS, 2013.
[20] X. Wang, et al., "Edge Enhanced Direct Visual Odometry," in BMVC. 2016.
[21] S. Li and D. Lee, "Fast visual odometry using intensity-assisted iterative closest point," in IEEE RAL, vol.1, no.2, 2016, pp. 992-999.
[22] M.Kuse and S. Shen, "Robust camera motion estimation using direct edge alignment and sub-gradient method," in IEEE ICRA, 2016.
[23] Y. Zhou, L. Kneip, and H. Li, "Semi-dense visual odometry for rgb-d cameras using approximate nearest neighbour fields," in IEEE ICRA, 2017.
[24] R. A. Brown, "Building a balanced k-d tree in $O(knlogn)$ time," Journal of Computer Graphics Techniques, vol. 4, no. 1, 2015, pp. 50-68.
[25] Z. Zhang, "Parameter estimation techniques: a tutorial with application to conic fitting," Image and Vision Computing, vol. 15, no. 1, 1997, pp. 59-76.
[26] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," ACM Transactions on Graphics (TOG), vol. 1, no. 1, 1982.