2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Congress Center Hamburg
Sept 28 - Oct 2, 2015. Hamburg, Germany

Robust Visual Odometry to Irregular Illumination Changes with RGB-D camera

Pyojin Kim, Hyon Lim, and H. Jin Kim

***Abstract*— Sensitivity to illumination conditions poses a challenge when utilizing visual odometry (VO) in various applications. To make VO robust with respect to illumination conditions, they need to be considered explicitly. In this paper, we propose a direct visual odometry method which can handle illumination changes by considering an affine illumination model to compensate abrupt, local light variations during direct motion estimation process. The core of our proposed method is to estimate the relative camera pose and the parameters of the illumination changes by minimizing the sum of squared photometric error with efficient second-order minimization. We evaluate the performance of the proposed algorithm on synthetic and real RGB-D datasets with ground-truth. Our result implies that the proposed method successfully estimates 6-DoF pose under significant illumination changes whereas existing direct visual odometry methods either fail or lose accuracy.**

I. Introduction

Estimating egomotion of a robot with video sequences coming from the camera attached to it is called visual odometry (VO) [1]. Existing VO techniques can be broadly divided into two types, depending on pose estimation method: feature based methods [2] and direct methods [3]. Many studies have adopted feature-based methods with monocular [4], [5], stereo [2], [6], and RGB-D camera [7], [8]. However, direct methods are getting more interests recently [3], [9], [10]. In these direct methods, the core idea is to minimize the sum of squared photometric error between two images under the photo-consistency assumption [11]. The fundamental assumption of existing direct VO methods is that brightness constraint is valid only under sufficient and constant illumination in the environment [12], which is an impractical assumption in most real-world applications as illustrated in Fig. 1. Thus, it is difficult to directly apply the existing direct VO methods when the illumination change is not negligible.

To make robust VO algorithm with respect to illumination changes, we propose a direct VO method which works well under sudden or local illumination changes during the direct motion estimation process by considering individual illumination changes in selected patches in an image. An affine illumination change model [14] is applied to individual patches which are selected based on planarity test with RANSAC using depth map of patches. To the best of our knowledge, this is the first direct VO which takes into account irregular, local illumination changes in the patches that are selected based on the planarity condition with RANSAC to make patches have the same normal vector to satisfy the affine illumination model [14].

All authors are with Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea. {rlavywls,hyonlim,hjinkim}@snu.ac.kr

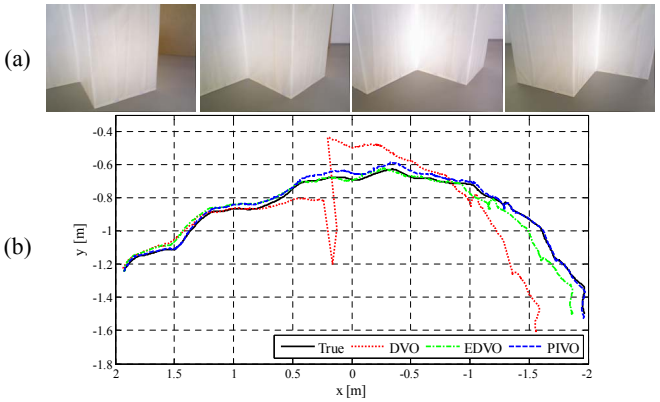

Fig. 1. **Illustrative examples of irregular illumination changes.** (a) An automatic exposure control of camera makes the intensity of images change when the camera moves in the TUM dataset 'fr3/struc&notex'. (b) Comparison of the estimation results of DVO [3], EDVO [13], and the algorithm proposed in this paper (namely PIVO) with (a). A large drift error takes place under DVO when the illumination changes occur. Our method shows best performance among other direct VO methods under illumination change, i.e. blue and black curves almost overlap.

The proposed method is evaluated with synthetic RGB-D dataset by modifying the TUM RGB-D benchmark dataset [15] and carefully chosen sequences that have illumination changes in [15]. We evaluate the performance of the proposed algorithm compared to the other direct VO methods [3], [13].

II. Related Work

For autonomous navigation of a robotic system, VO has been actively utilized both on the ground [2], [16] and in the air [7]. As discussed in Section I, the VO methods can be categorized as two in terms of a kind of information used in pose estimation process: so called feature-based methods [2] and direct methods [3].

The feature-based methods encode an image to a list of keypoints (i.e. a list of image coordinates of distinctive points) and solve the geometric pose estimation problem on that list of coordinates and association table. Many keypoint extraction and matching algorithms are applied to those feature-based VO, however, they require enough brightness and textures to extract consistent keypoints from an image. In varying illumination conditions that we consider in this paper, this requirement degrades the performance of feature-based VO. As a result, the feature-based methods cannot

correctly estimate their own position in featureless or dark environments.

Therefore, direct methods which exploit the entire image information are receiving attention recently with the help of hardware progress. In [17] and [18], the camera pose tracking is performed based on alignment of 3D point clouds (ICP), which presents successful results in terms of robustness, computation time, and accuracy. The direct VO techniques ([3], [19], [20]) are proposed, which minimize the photometric error between image frames. They are fundamentally based on the photo-consistency assumption, which means that a point in the 3D world represents the same brightness intensity at different camera poses [12]. In [19], a semi-direct method was successfully implemented on an aerial vehicle with a single downward-looking camera. [3] estimates the relative RGB-D camera motion accurately with a robust error function which rejects the noise and outliers in the image. In [20], quadrifocal geometry constraints are used to track the trajectory of a stereo camera. Even though the outlier rejection algorithms exist in the above methods such as a robust error function, a large drift of the estimated trajectory caused by the abrupt illumination changes is inevitable since the photo-consistency assumption is no longer valid.

Only a few direct VO methods give consideration to illumination changes during the direct motion estimation. It is assumed in [13] that the entire pixels in the image follow the same illumination change model [14]. A similar light variation model is also used in [21] and [22], which need the reconstructed 3D scene model for camera pose tracking and use a single global brightness (bias shift) parameter in the image. In order to ignore the illumination changes altogether between image frames, [23] estimate a pure albedo image of the texture. In contrast with the works mentioned above, the proposed direct VO method in this paper takes into account both global and local illumination changes. In particular, general illumination changes can be handled because each patch is allowed to have different model parameters of illumination changes.

III. Notation and Problem Statement

The superscript k is used to denote the index of an image frame. An intensity image obtained at time step k is denoted with I^k. In the intensity image I^k, i-th image patch is denoted with I_i^k. For an arbitrary 2D pixel point, pixel coordinates in I_i^k are denoted as $\mathbf{x}_{ij}^k = \left[x_{ij}^k, y_{ij}^k\right]^\top$, where the first subscript i is the patch index, and j is the pixel index. 3D points $\mathbf{X}_{ij}^k = \left[X_{ij}^k, Y_{ij}^k, Z_{ij}^k\right]^\top$ defined in camera coordinate $\{C^k\}$ are mapped to the pixel coordinates $\mathbf{x}_{ij}^k$ through the camera projection function $\pi : \mathbb{R}^3 \mapsto \mathbb{R}^2$

$$\mathbf{x}_{ij}^k = \pi\left(\mathbf{X}_{ij}^k\right) = \begin{bmatrix} \frac{X_{ij}^k \cdot f}{Z_{ij}^k} + p_x \\ \frac{Y_{ij}^k \cdot f}{Z_{ij}^k} + p_y \end{bmatrix} \tag{1}$$

The above projection function is determined uniquely with the camera intrinsic parameters f, p_x, p_y [24].

Conversely, a 3D point $\mathbf{X}_{ij}^k$ can be computed with the depth value Z_{ij}^k (from depth map of RGB-D sensor) and $\mathbf{x}_{ij}^k$ through the inverse projection function $\pi^{-1} : \mathbb{R}^2 \mapsto \mathbb{R}^3$

$$\mathbf{X}_{ij}^k = \pi^{-1}\left(\mathbf{x}_{ij}^k, Z_{ij}^k\right) = \begin{bmatrix} \frac{x_{ij}^k - p_x}{f} Z_{ij}^k \\ \frac{y_{ij}^k - p_y}{f} Z_{ij}^k \\ Z_{ij}^k \end{bmatrix} \tag{2}$$

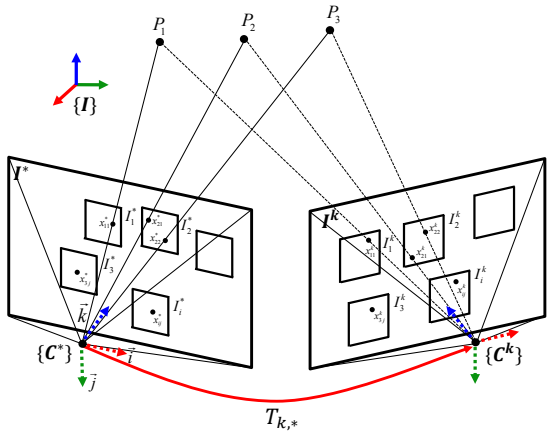

Fig. 2. **The notations and setting of the proposed visual odometry algorithm.** Our goal is to estimate $T_{k,*}$ given the patch-based keyframe gray image (I^*), and depth image (D^*), and the current gray image (I^k).

The relative position and orientation between the current camera frame $\{C^k\}$ and the keyframe $\{C^*\}$ are represented with the rigid body transformation matrix $T_{k,*} \in SE(3)$:

$$\tilde{\mathbf{X}}^k = T_{k,*}\tilde{\mathbf{X}}^* \tag{3}$$

where $\tilde{\mathbf{X}}^k = [\mathbf{X}^{k\top}, 1]^\top$ is the homogeneous form of $\mathbf{X}^k$. In this paper, a minimal representation of Lie group $SE(3)$, i.e. Lie algebra $se(3)$ parameter $\boldsymbol{\xi}$, is mainly used to express the incremental displacements during a numerical optimization algorithm. We can represent the Lie algebra parameter with a 6×1 vector $\boldsymbol{\xi} = \left[\boldsymbol{\nu}^\top, \boldsymbol{\omega}^\top\right]^\top$ where $\boldsymbol{\nu}$ and $\boldsymbol{\omega}$ are infinitesimal translation and rotation in the tangent space of the matrix group $SE(3)$. The rigid body transformation matrix $T \in SE(3)$ can be calculated by the exponential map:

$$T(\boldsymbol{\xi}) = \exp(\hat{\xi}) \tag{4}$$

where $\hat{\xi}$ is a 4×4 twist matrix from the Lie algebra $\boldsymbol{\xi}$ [25].

With the above notations, the problem we want to solve is to estimate the rigid body transformation matrix $T_{k,*}$ given a sequence of image frames and the corresponding depth maps from RGB-D sensor under arbitrary, abrupt, and partial illumination changes between consecutive image frames.

IV. Proposed Visual Odometry Algorithm

The schematic overview of the proposed visual odometry algorithm and the data flow are represented in Fig. 3. First, RGB and depth images from RGB-D camera are used to initialize a keyframe with patches. The pixel points that are the center of the patches in RGB image are detected by the blob detector like LoG, DoG, or SURF [26]. Among them, the only valid patches are saved and utilized in motion estimation process until a next keyframe is re-initialized.

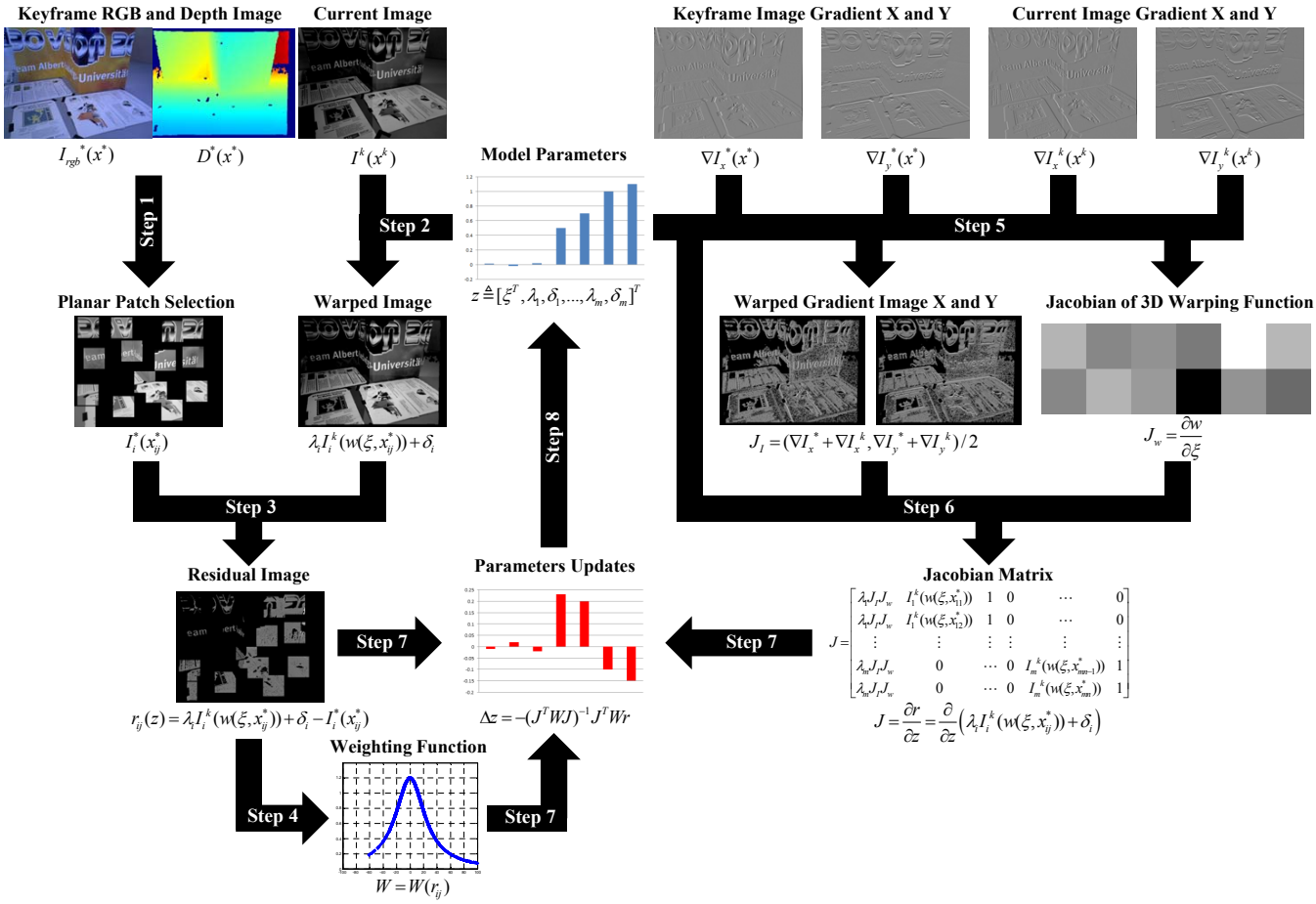

Fig. 3. **A overview of the proposed visual odometry algorithm.** (Step 1) Initially, the keyframe is initialized with the RGB I^* and depth D^* image with patch selection process. (Step 2) The current image I^k is warped with the current estimated model parameters $\mathbf{z}$. (Step 3) The residual image is then calculated with the results from Step 1 and Step 2. (Step 4) Weighting function is determined by the residual distribution. (Step 5) The gradient images of I^* and I^k are warped and Jacobian matrix of 3D warping function is calculated with the current model parameters. (Step 6) The entire Jacobian matrix is obtained with the results in Step 5 and the current model parameters to minimize the newly proposed photometric error. (Step 7) In the end, this step yields model parameter updates $\triangle\mathbf{z}$ by combining the Jacobian matrix J, the residual vector $\mathbf{r}$, and the diagonal weighting matrix W. (Step 8) $\triangle\mathbf{z}$ is added to the previous guess of the model parameters $\mathbf{z}$. The above procedure is repeated until convergence and the estimated trajectory of RGB-D camera is obtained by concatenating the motion estimation results.

After the keyframe is initialized, a residual image is obtained by using the keyframe with patches, current image frame, and the model parameters $\boldsymbol{\xi}$. Each weight of the residual value is determined by t-distribution of all residuals. Next, Jacobian matrix is calculated with the gradient images of the keyframe and the current image frame to minimize the newly proposed photometric error. The proposed visual odometry algorithm is based on not the photo-consistency assumption like [3], [19], and [27], but the photo-consistency assumption with compensation of illumination changes between the two consecutive images. By combining the Jacobian matrix J, the residual vector $\mathbf{r}$, and the diagonal weighting matrix W, the incremental displacements of the model parameter $\triangle\mathbf{z}$ is calculated. The model parameter $\mathbf{z}$ is updated and the above procedure is repeated until convergence. If the Euclidian distance between the keyframe and the current image frame is too far, the next keyframe is newly initialized with the current RGB and depth image. Finally, the whole trajectory of RGB-D camera is obtained by concatenating the frame-to-frame motion estimation results.

A. Illumination Change Model

The photo-consistency assumption employed in [3] and [19] is not always valid in real world because illumination changes such as highlights, shadows caused by the variation of the viewpoint of the camera, unpredictable changes of light source, and the camera automatic gain are unavoidable phenomena during the direct visual odometry. To reflect not only the global but also the local illumination changes between the keyframe and the current frame, we adopt the affine illumination change model [14] per patch as follows:

$$\lambda_i I_i^k + \delta_i = I_i^* \tag{5}$$

Here λ_i and δ_i are the model parameters to represent contrast and brightness changes of the i-th patch in the image. During the optimization process, these parameters per patch are estimated and utilized to compensate the irregular illumination changes between the keyframe and the current frame.

B. Planar Patch Selection

The patch-based keyframe image generation method is employed to solve two critical issues in direct visual odometry process: reducing computation time and taking into account both global and local illumination changes.

The computation time is proportional to the number of pixels in the direct visual odometry since every pixel should go through the 3D backward or forward image warping. For example, in [3], [13], the entire pixels in the image are used. We observe, however, that the number of pixels fewer than 50% of the entire pixels in the image is still enough to estimate the motion of the camera [19]. Thus, the patch-based keyframe image as depicted in Fig. 4(a) can be used to carry out the direct visual odometry process.

For the issue of illumination changes, although the global changes have been considered in [13] and [22], the partial changes have not been concerned yet. On the other hand, to take into account the local illumination changes, we assume that each patch follows the different illumination changes individually as illustrated in Fig. 4(a), where each patch has its own unique λ_i and δ_i. And to make the above assumption valid, the patches only on the planar surface in the real world are selected because 3D points on the same plane undergo the similar illumination changes [14]. It can be achieved by utilizing the blob detector like LoG, DoG, SURF [26] and the plane model based RANSAC algorithm [28]. At first, the blob detector is used to extract the patches on the planar surface in the 3D space. After that, RANSAC algorithm is applied to robustly fit the plane to a set of 3D data points from the extracted patch's pixel points and the depth image from RGB-D sensor. The plane of each patch is fitted with the following equation:

$$ax + by + cz + d = 0 \tag{6}$$

where a, b, c, and d are the model parameters of the plane and x, y, z is the 3D point on the plane. Based on Eq. (6), the error function of the plane RANSAC algorithm can be formulated as follows:

$$l_j = \frac{\left|aX_{ij}^* + bY_{ij}^* + cZ_{ij}^* + d\right|}{\sqrt{a^2 + b^2 + c^2}} \tag{7}$$

where l_j is the length between the plane (a, b, c, d) and the 3D point $(X_{ij}^*, Y_{ij}^*, Z_{ij}^*)$ expressed in the camera keyframe. Using RANSAC with the error function Eq. (7), we can determine how many points are out of the plane in each patch. Fig. 5(b) depicts the outlier points as red and inlier points as black. If more than half of the points of a patch are out of the plane, these kinds of patches are rejected and discarded. In this manner, only the valid patches that are on the plane in 3D space as described in Fig. 5(a) survive and pass through the direct motion estimation process.

C. Direct Motion Estimation

We generated the patch-based keyframe gray and depth image I^*, D^* in the previous step, and the current image frame I^k comes from the RGB-D camera. With I^*, D^*, and I^k, our goal is to estimate the relative camera pose $T_{k,*}$

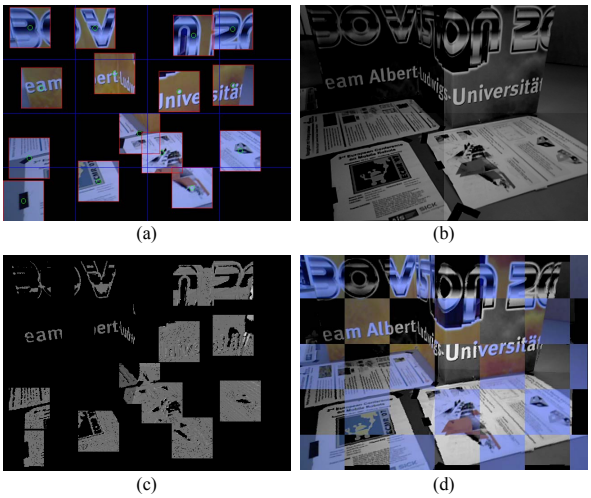

Fig. 4. **Input and output images of the proposed visual odometry algorithm.** (a) Patch-based keyframe image generated by the SURF and the plane RANSAC algorithm. (b) Current gray image frame captured under illumination changes. (c) Residual image between the patch-based keyframe and the current image frame, which we want to minimize. (d) Image alignment result with (a) and (b) based on estimated relative camera pose. The colored part comes from (a) and the gray part comes from (b).

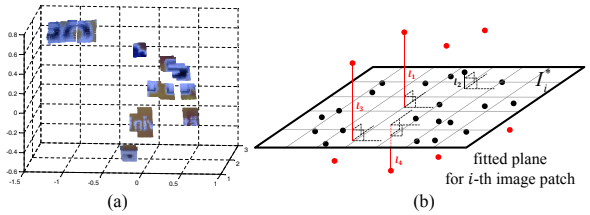

Fig. 5. **Planar patch selection results based on the plane RANSAC algorithm.** (a) The extracted valid patches which are on the planar surface in the 3D Cartesian coordinate. (b) The process of determining whether 3D points are on the same plane or not.

and the illumination change model parameters per patch, i.e., $\lambda_1, \delta_1, ..., \lambda_m, \delta_m$ where m is the number of patches in the keyframe image I^*. In contrast with the existing photo-consistency assumption [3], our photo-consistency assumption that considers the illumination changes can be written as the following equation:

$$\lambda_i I_i^k(w(\boldsymbol{\xi}, \mathbf{x}_{ij}^*)) + \delta_i = I_i^*(\mathbf{x}_{ij}^*) \tag{8}$$

$$w(\boldsymbol{\xi}, \mathbf{x}_{ij}^*) = \pi(T(\boldsymbol{\xi}) \cdot \pi^{-1}(\mathbf{x}_{ij}^*, Z_{ij}^*))) \tag{9}$$

where $\boldsymbol{\xi} \in \mathbb{R}^6$ represents the relative motion of the camera and $w(\boldsymbol{\xi}, \mathbf{x}_{ij}^*)$ is the 3D backward warping function which is a one-to-one mapping from a pixel point $\mathbf{x}_{ij}^*$ in the patch-based keyframe image to a pixel coordinate in the current image frame given the relative camera motion $\boldsymbol{\xi}$. To simplify expression of the overall model parameters which we have to estimate, the integrated new model parameter $\mathbf{z}$ is defined as follows:

$$\mathbf{z} := \left[\boldsymbol{\xi}^\top, \lambda_1, \delta_1, ..., \lambda_m, \delta_m\right]^\top \in \mathbb{R}^{6+2m} \tag{10}$$

Based on the defined notations and the modified photo-consistency assumption as written in Eq. (8), we define the

residual of the j-th pixel in the i-th patch as the photometric difference with compensation of the illumination changes between pixels observed in the keyframe and the current frame:

$$r_{ij}(\mathbf{z}) = \lambda_i I_i^k(w(\boldsymbol{\xi}, \mathbf{x}_{ij}^*)) + \delta_i - I_i^*(\mathbf{x}_{ij}^*) \tag{11}$$

We seek the optimal model parameter $\mathbf{z}^*$ that minimizes the weighted sum of squared residuals, which is the following non-linear weighted least square problem:

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \sum_{i=1}^{m} \sum_{j=1}^{n} W(r_{ij}) r_{ij}^2(\mathbf{z}) \tag{12}$$

where n is the number of pixels in each patch and $W(r_{ij})$ is the weighting function that gives the different weights to each residual value based on the residual distribution. We assume that the residual distribution follows the t-distribution by following [3]. We solve the iteratively re-weighted least square (IRLS) problem with weighting function determined by the t-distribution.

To find the optimal model parameter $\mathbf{z}^*$ written in Eq. (12), Gauss-Newton algorithm is selected. And there are several image alignment strategies in the direct methods: forward compositional (FC), inverse compositional (IC), and efficient second-order minimization (ESM). Among them, it is well known that ESM method outperforms the other methods [10], [13]. Thus, the Jacobian matrix is calculated with respect to the newly defined model parameter $\mathbf{z}$ based on the ESM algorithm [29]. By plugging and arranging the equations (8)−(12), the normal equation is obtained:

$$J^T W J \triangle \mathbf{z} = -J^T W \mathbf{r} \tag{13}$$

$$J \in \mathbb{R}^{(mn)\times(6+2m)}, W \in \mathbb{R}^{(mn)\times(mn)}, \mathbf{r} \in \mathbb{R}^{(mn)}$$

$$J = \begin{bmatrix} \lambda_1 J_I J_w & I_1^k(w(\boldsymbol{\xi}, \mathbf{x}_{11}^*)) & 1 & 0 & \dots & 0 \\ \lambda_1 J_I J_w & I_1^k(w(\boldsymbol{\xi}, \mathbf{x}_{12}^*)) & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \lambda_m J_I J_w & 0 & \dots & 0 & I_m^k(w(\boldsymbol{\xi}, \mathbf{x}_{mn-1}^*)) & 1 \\ \lambda_m J_I J_w & 0 & \dots & 0 & I_m^k(w(\boldsymbol{\xi}, \mathbf{x}_{mn}^*)) & 1 \end{bmatrix} \tag{14}$$

$$J_I J_w = \frac{1}{2}\left(\frac{\partial I^*(\mathbf{x}^*)}{\partial \mathbf{x}} + \frac{\partial I^k(w(\boldsymbol{\xi}, \mathbf{x}^*))}{\partial \mathbf{x}}\right)\frac{\partial w(\boldsymbol{\xi}, \mathbf{x}^*)}{\partial \boldsymbol{\xi}} \tag{15}$$

In Eq. (13), note that J is the stacked Jacobian matrix and W is the diagonal matrix that represents each residual's weight and $\mathbf{r}$ is the tall residual vector coming from Eq. (11). At every iteration in IRLS, we can compute the incremental value $\triangle\mathbf{z}$ and based on that incremental values, we update the relative camera pose $T_{k,*}$ and the illumination change model parameters λ_i, δ_i with $i = 1, ..., m$ until the integrated model parameter $\mathbf{z}$ converges. Additionally, similar to [3] and [13], a coarse-to-fine approach is employed with the image pyramid method for robustness and faster convergence. The gaussian pyramid is utilized to compute the image pyramid and run the above optimization process from the coarsest level to the finest. In this manner, we can compute the relative camera motion and the illumination change model parameters much faster and accurately.

Fig. 6. **The RGB image sequences in synthetic & author-collected RGB-D dataset.** (a) To simulate irregular illumination changes, we manipulate the intensity values of the images in TUM RGB-D dataset based on four different illumination change models. (b) The images are captured in a stationary position with calibrated Asus Xtion Pro Live RGB-D sensor. To test the robustness to illumination changes of each method in real world, lights in the room are turned on and off repeatedly by author.

V. Evaluation

The proposed *Patch-based Illumination invariant Visual Odometry* (PIVO) algorithm is tested with two types of datasets: synthetic RGB-D dataset which is based on TUM RGB-D benchmark [15] and author-collected RGB-D dataset. In RGB images of the synthetic RGB-D dataset, artificial illumination changes are applied to validate the proposed visual odometry method. Author-collected RGB-D dataset consists of the RGB and depth images captured under an actual illumination change with static pose. Three performance metrics are used to evaluate the performance of the proposed visual odometry algorithm: root mean square error (RMSE) of the relative pose error (RPE), and the absolute trajectory error (ATE) defined in [15] and the final drift error divided by the length of the entire trajectory. We compare the motion estimation results with author-implemented version of [3] and [13]. All calculations and processes are conducted on a desktop computer with Intel Core i5 with 3.2 Ghz with 8GB memory and the program is implemented in MATLAB. PIVO takes about 200-300 ms per frame in our current setting.

A. Synthetic RGB-D Dataset

The TUM RGB-D dataset consists of RGB and depth images taken at full frame rate (30 Hz) and ground-truth pose of the RGB-D camera obtained from a motion capture system (100 Hz). But they do not involve abrupt, irregular illumination changes. Thus, we modify intensity values based on the illumination change model [14] to simulate the irregular illumination changes in TUM RGB-D dataset. A image, at first, is divided into four regions and the intensity values in each of the four region are modified with four different illumination change models to give partial lighting changes. We call the modified image sequences as the synthetic RGB-D dataset and some of the synthetic RGB images are drawn in Fig. 6(a).

We evaluate the motion estimation accuracy of PIVO in an environment where the irregular illumination changes occur compared to the Dense Visual Odometry (DVO) [3], and Efficient DVO (EDVO) [13]. The evaluations are performed with the eleven synthetic RGB-D image sequences and the

TABLE I

ESTIMATION RESULTS WITH SYNTHETIC RGB-D DATASET

Name of Dataset	RPE [drift m/s]			Drift Error [%]		
	DVO	EDVO	PIVO	DVO	EDVO	PIVO
fr1/desk	0.257	0.076	0.057	3.23	1.65	1.15
fr1/desk2	0.616	0.252	0.183	15.34	6.18	3.42
fr1/floor	1.214	0.224	0.203	7.18	9.05	5.67
fr1/room	3.648	1.119	0.262	59.39	14.18	3.25
fr2/desk	0.288	0.232	0.231	13.88	1.19	0.79
fr2/largenoloop	5.300	3.624	2.268	74.24	48.06	12.32
fr3/longoffice	0.592	0.045	0.040	9.12	1.57	1.39
fr3/nostruc&notex	5.757	14.014	0.067	261.88	1325.86	15.62
fr3/nostruc&tex	1.615	0.228	0.107	106.63	13.16	8.49
fr3/struc&notex	10.062	1.483	0.021	372.98	312.43	5.03
fr3/struc&tex	0.105	0.034	0.023	31.56	4.54	2.34

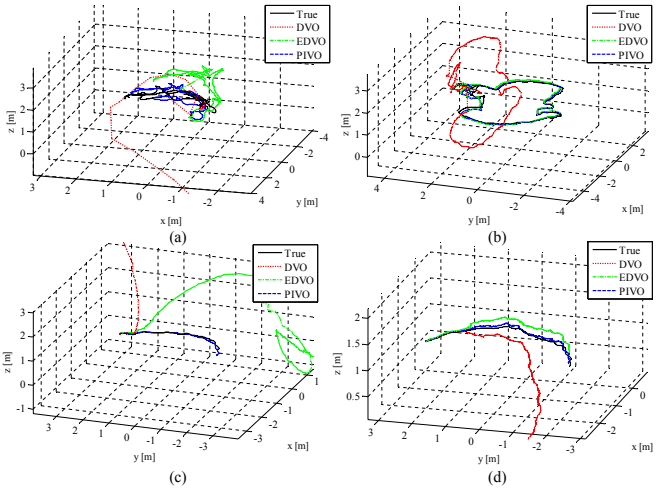

Fig. 7. **Comparison of three visual odometry estimation results against the ground truth.** (a) 'fr1/room' (b) 'fr3/longoffice' (c) 'fr3/struc&notex' (d) 'fr3/struc&tex'. The motion estimation results of the tested visual odometry methods (DVO, EDVO, and PIVO) are drawn with the ground truth trajectory. It is shown that PIVO is relatively more closer to the ground truth in all cases.

motion estimation results for each visual odometry method are summarized in TABLE I. In all cases, we observe that our method generates better results than DVO, EDVO. In most cases except 'fr1/desk', DVO has failed to estimate pose with illumination changes. EDVO presents good performances on some datasets: 'fr2/desk', 'fr1/desk', and 'fr3/longoffice' due to the compensation factor of the global illumination changes. However, it shows poor results on 'fr3/nostruc&notex', 'fr3/struc&notex'. Fig. 7 shows the 3D estimated trajectories of each visual odometry method with four different image sequences. Absolute trajectory error (ATE) of each method is also presented in Fig. 8 with the same datasets used in Fig. 7. Estimation error of the other two visual odometry methods except PIVO increases rapidly during the interval marked by the gray dotted lines where the illumination changes occur in the image sequences.

In particular, the dataset 'fr3/struc&notex' is selected to analyze the result in detail. During the period from 100 to 300 image frames where the illumination changes occur,

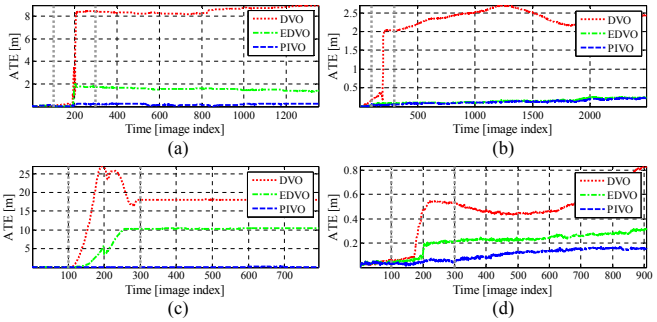

Fig. 8. **Absolute trajectory errors of each tested dataset.** (a) 'fr1/room' (b) 'fr3/longoffice' (c) 'fr3/struc&notex' (d) 'fr3/struc&tex'. Irregular illumination changes occur in the time interval between the gray dotted lines. During that interval, ATE of DVO, EDVO increases rapidly.

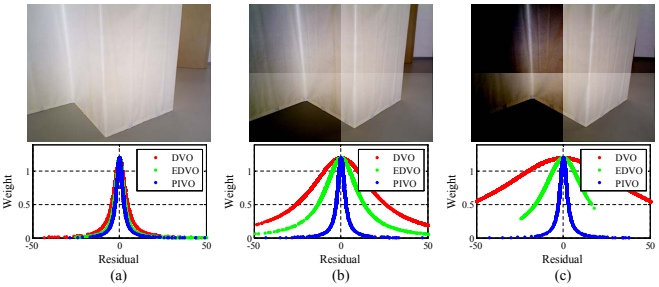

Fig. 9. **Weighting functions with respect to the residuals of each method.** (a+b+c) Top row shows 100-, 130-, and 200-th image frame of 'fr3/struc&notex', from left to right. The graphs are weight functions assigned to residual at each frame. DVO and EDVO assign high weight to large residual as the photo consistency assumption breaks down.

we can find that PIVO estimates the position, pose of the camera accurately whereas the drift of DVO, EDVO gradually increases as described in Fig. 8(c). The main reason for this difference is that the photo-consistency assumption is violated in this period. Although a robust weighting function is used for discarding outliers in DVO or a global affine illumination change concept is considered in EDVO, the cost functions in DVO, EDVO are not effective enough to take into account the sudden, partial lighting variations. PIVO efficiently copes with this kind of illumination changes by using the proposed cost function in Eq. (12). This can be confirmed in Fig. 9. DVO and EDVO assign high weight to large residual as the photo-consistency assumption breaks down in the cases of Figs. 9(b) and (c), which degrades the accuracy. On the other hand, under PIVO, the large weight remains only over small residual during the light variations, which means that the illumination changes are compensated properly.

B. Author-collected stationary RGB-D dataset

RGB and depth images in the RGB-D dataset collected by the author are taken in a fixed position, which means that RGB-D camera does not move at all throughout the whole image sequences as illustrated in Fig. 6(b). Instead, lights in the room are turned on and off repeatedly to test the robustness to illumination changes for the individual visual odometry methods. The evaluations are performed with three

TABLE II

ESTIMATION RESULTS WITH AUTHOR-COLLECTED RGB-D DATASET

Name of Dataset	RPE [drift m/s]			ATE [m]		
	DVO	EDVO	PIVO	DVO	EDVO	PIVO
LAB1	1.320	1.036	0.044	2.484	1.868	0.247
LAB2	1.545	0.210	0.006	4.499	0.716	0.014
LAB3	0.222	0.009	0.011	0.942	0.014	0.023

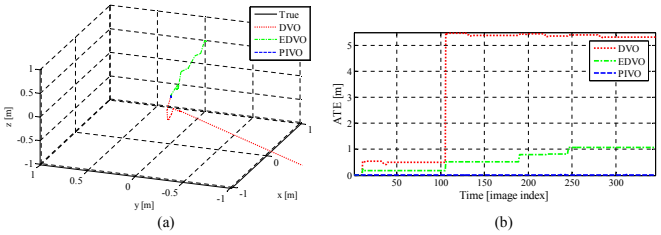

Fig. 10. **Comparison of 3D estimated trajectories and ATE in 'LAB2'.** (a) PIVO estimates the position of the fixed camera correctly. (b) ATE of each visual odometry method is drawn. Error of DVO and EDVO increases whereas PIVO maintains almost zero value throughout the entire sequences.

types of dataset. Sudden illumination changes take place in the entire images in 'LAB1'. Next, partial and irregular light variations occur in 'LAB2'. Lastly, lighting changes happen a little in 'LAB3'. The estimation results of each image sequence are summarized in TABLE II.

As we expected, PIVO estimates the position of the stationary RGB-D sensor correctly in 'LAB2' and 'LAB3'. On the other hand, incorrect movements of the camera as drawn in Fig. 10 are estimated by DVO and EDVO because of abrupt, irregular illumination changes in the images. DVO even produces a large drift error in 'LAB3', which means DVO is particularly sensitive to changes in light.

VI. CONCLUSION

In this paper, we proposed a patch-based illumination invariant visual odometry, which works well in the irregular illumination change. To consider the partial light variations, the planar patch selection process is employed and the illumination change model is adopted in each extracted patch. The proposed cost function reflecting the illumination changes is minimized by using the robust weighting function and the efficient second-order minimization (ESM) image alignment method. As a result, our method can accurately estimate the motion of the camera regardless of the partial lighting changes. Evaluation results with the synthetic RGB-D dataset and real experiments show that the accuracy of our algorithm is superior to the other direct visual odometry methods not only in the ordinary image sequences, but also in the illumination change.

ACKNOWLEDGMENT

This research was supported by a grant to Unmanned Technology Research Center funded by Defense Acquisition Program Administration and the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science, ICT & Future Planning (MSIP) (No. 2014034854).

REFERENCES

[1] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *Robotics & Automation Magazine, IEEE*, vol. 18, no. 4, pp. 80–92, 2011.

[2] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *CVPR*, vol. 1. IEEE, 2004, pp. I–652.

[3] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for rgb-d cameras," in *ICRA*. IEEE, 2013, pp. 3748–3754.

[4] G. Klein and D. Murray, "Parallel tracking and mapping on a camera phone," in *ISMAR 2009*. IEEE, 2009, pp. 83–86.

[5] H. Lim, J. Lim, and H. J. Kim, "Real-time 6-dof monocular visual slam in a large-scale environment," in *ICRA*. IEEE, 2014, pp. 1532–1539.

[6] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *IV*. IEEE, 2011, pp. 963–968.

[7] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an rgb-d camera," in *ISRR*, 2011, pp. 1–16.

[8] J. Zhang, M. Kaess, and S. Singh, "Real-time depth enhanced monocular odometry," in *IROS*. IEEE, 2014, pp. 4973–4980.

[9] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *ICCV*. IEEE, 2011, pp. 2320–2327.

[10] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *ECCV*. Springer, 2014, pp. 834–849.

[11] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.

[12] M. Irani and P. Anandan, "About direct methods," in *Vision Algorithms: Theory and Practice*. Springer, 2000, pp. 267–277.

[13] S. Klose, P. Heise, and A. Knoll, "Efficient compositional approaches for real-time robust direct visual odometry from rgb-d data," in *IROS*. IEEE, 2013, pp. 1100–1106.

[14] H. Jin, P. Favaro, and S. Soatto, "Real-time feature tracking and outlier rejection with changes in illumination," in *ICCV*, vol. 1. IEEE Computer Society, 2001, pp. 684–684.

[15] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *IROS*. IEEE, 2012, pp. 573–580.

[16] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the mars exploration rovers," *Journal of Field Robotics*, vol. 24, no. 3, pp. 169–186, 2007.

[17] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*. IEEE, 2001, pp. 145–152.

[18] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *ISMAR*. IEEE, 2011, pp. 127–136.

[19] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *ICRA*. IEEE, 2014, pp. 15–22.

[20] A. I. Comport, E. Malis, and P. Rives, "Real-time quadrifocal visual odometry," *IJRR*, vol. 29, no. 2-3, pp. 245–266, 2010.

[21] G. Silveira, E. Malis, and P. Rives, "An efficient direct approach to visual slam," *TRO*, vol. 24, no. 5, pp. 969–979, 2008.

[22] M. Meilland, A. Comport, P. Rives, and I. S. A. Méditerranée, "Real-time dense visual tracking under large lighting variations," in *BMVC*, vol. 29, 2011.

[23] C. Kerl, M. Souiai, J. Sturm, and D. Cremers, "Towards illumination-invariant 3d reconstruction using tof rgb-d cameras," in *3DV*, 2014.

[24] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[25] Y. Ma, *An invitation to 3-d vision: from images to geometric models*. springer, 2004, vol. 26.

[26] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *ECCV*. Springer, 2006, pp. 404–417.

[27] F. Steinbrucker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense rgb-d images," in *ICCV Workshops*. IEEE, 2011, pp. 719–722.

[28] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[29] S. Benhimane and E. Malis, "Real-time image-based tracking of planes using efficient second-order minimization," in *IROS*, vol. 1. IEEE, 2004, pp. 943–948.