International Journal of Control, Automation and Systems 18(X) (2020) 1-12
http://dx.doi.org/10.1007/s12555-019-0284-1

ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Fast and Accurate Self-calibration Using Vanishing Point Detection in Manmade Environments

Sang Jun Lee and Sung Soo Hwang*

Abstract: Interests of auto-calibration have been increased in several camera systems. This paper presents a novel self-calibration method using fast and accurate vanishing point detection algorithm that works in manmade environments. The proposed algorithm estimates focal length assuming that the principal point is the center of an image to satisfy the orthogonality of three vanishing points. By using proposed vanishing point detection algorithm and minimization of the proposed objective function, the proposed system detects accurate vanishing points with focal length outperforming other methods. The proposed vanishing point detection algorithm detects vanishing points by using J-linkage based method that is more delicate by fragmentation and re-merging strategies. The proposed objective function finally detects vanishing points that meets orthogonality among estimated hypotheses for vanishing points by checking several geometric relationships. We believe that the proposed method can be used for automatic camera calibration, localization of a camera in an autonomous navigation system, and three-dimensional reconstruction of a single-view image.

Keywords: Auto camera calibration, line clustering, single view geometry, vanishing point detection.

1. INTRODUCTION

Camera calibration is the essential issue for ubiquitous applications in robotics and control such as visual servoing [1] and autonomous navigation system [2]. For examples, position-based visual servoing or hybrid visual servoing takes visual information from 3D geometry for feedback control of the pose and motion of robot. Thus, it highly depends on the quality of accurate intrinsic parameters, and it is sensitive to calibration errors [1]. Moreover, visual SLAM or visual odometry system [2] needs accurate calibration parameters since this system is basically operated on a map including 3D points generated by camera poses with calibration matrix. Thus, estimating accurate calibration matrix is important problem for theses applications that need data association between 3D-2D points.

While various methods are proposed for camera calibration using different approaches such as Morphological pattern [3], laser [4] calibration matrix by using vanishing points [8–10]. Vanishing points are similar to the principal of image projection since an ideal point in 3D space is projected in finite 2D point by camera calibration matrix. Therefore, given orthogonal three vanishing points according to three axes x, y, and z, a relevant camera calibration matrix can be estimated as discussed in [5].

Conventional approaches for vanishing point detection are as follows: Vanishing points are detected as intersection points from each cluster of line segments in the image space that are parallel in the world coordinate system. To estimate line clusters easily, most approaches are operated on Manhattan world assumption [6], which is manmade environments that edges detected in those scenes are according to three axes. For calibrated images, vanishing points can be found as intersection points by mapping the line segments onto a Gaussian sphere [7]. However, most of the images are un-calibrated, and line clustering required for the vanishing point detection is much more complex.

To the end, various studies have been conducted for vanishing point detection on un-calibrated images. Tardif [8] successfully and rapidly detected vanishing points in un-calibrated images by applying line clusters simultaneously. However, in this method, there are several errors in the clusters and the orthogonality constraints are not ensured. Geometric parsing suggested by Barinova [9] can successfully detects the vanishing points from the non-Manhattan world by using energy function what they suggest. However, this method is extremely slow and does not satisfy the orthogonality constraint. Wildenauer and Han-

Manuscript received April 17, 2019; revised December 3, 2019 and January 14, 2020; accepted February 4, 2020. Recommended by Associate Editor DaeEun Kim under the direction of Editor Euntai Kim. This journal was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education [No. 2016R1D1A3B03934808].

Sang-Jun Lee and Sung-Soo Hwang are with the School of Computer Science and Engineering, Handong Global University, 558, Handong-ro, Heunghae-eup, Buk-gu, Pohang-si, 37554, Korea (e-mails: eowjd4@naver.com, sshwang@handong.edu).
* Corresponding author.

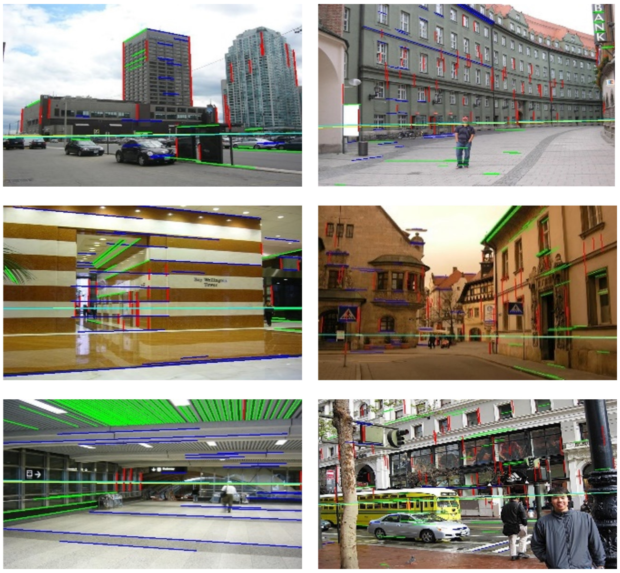


Fig. 1. Examples of detected vanishing points of the proposed system. The yellow line is the horizontal line of the ground truth, and the cyan line is the estimated horizontal line.

bury proposed a Random Sample Consensus (RANSAC)-based method [10] that could detect vanishing points satisfying the orthogonality constraints with a high accuracy. However, this method depends on the Manhattan world assumption for obtaining good samples for detecting vanishing points because the RANSAC-based method is weak for random samples.

This paper proposes a new vanishing point detection algorithm to obtain accurate camera calibration. The proposed system estimates hypotheses for vanishing points by resolving defects in J-linkage based line clustering by using fragmentation and re-merging with regard to characteristics of each clusters. The estimated hypotheses, then, are filtered by the proposed objective function minimization satisfying orthogonality constraints with several geometric relationships to detect vanishing points that are more robust.

The proposed method has advantages regarding the discussed issues. First, the proposed method uses the specifically constructed line segments (filtered line segments) for accurate and fast line clustering. Second, the proposed strategies of line clustering further enhances the speed and accuracy of line clustering. And consistent detection is guaranteed as using non-iterative approach of J-linkage based method unlike RANSAC. Third, the proposed method detect accurate vanishing points with camera calibration matrix even Non-manhattan world by using the proposed objective function because design of the objective function reflects robust geometric relationships such as non-similarity between true vanishing points, multiple image-based consistency, and perpendicularity between horizontal line and zenith point. We show the examples of the detected vanishing points by using the proposed system in the Fig. 1. Finally, the detected accurate calibration matrix can be utilized for navigation systems.

This paper is organized as follows: We present the related works and some knowledge for the comprehension of this paper in Section 2 and Section 3. The details of the proposed method are reported in Section 4. In Section 5, we compare the performance of the proposed method with previous methods, and we conclude this paper in Section 6.

2. RELATED WORKS

Various researches for vanishing point detection have been conducted on man-made environments. Tardif suggested a method using J-linkage [11] in [8]. This method utilizes a non-iterative solution for simultaneously estimating vanishing points and line clusters using J-linkage, which is suggested for fitting multiple models. However, the vanishing points detected by this method in non-Manhattan world are not accurate and do not guarantee orthogonality between them. An improved method using expectation maximization (EM) suggested in [8] also does not ensure the satisfaction of the orthogonality because the initial vanishing points were already not orthogonal. Moreover, the EM method is highly affected by the initial values. Furthermore, EM-based methods [12–15] are computationally expensive.

Ransac 4 Line (R4L), [10] uses RANSAC [16] for vanishing point detection. This method randomly selects four lines and calculates three hypotheses for the vanishing points, with the focal length reflecting the orthogonality constraints. Then the hypotheses are detected as the vanishing points if the consensus set of inliers with regard to the line segments is maximized. This method ensures orthogonality between three vanishing points. However, this method uses random sampling, and so, it depends on the quality of the selected line segments. Therefore, this method relies on the Manhattan world assumption to select good-quality line segments.

Geometric parsing suggested in [9] simultaneously estimates vanishing points, clusters of line segments, and pixels of the line segments by optimizing the proposed energy function. The energy function assumes that the principal point is the center of the image and that the zenith line passing through both the zenith and principal points and the horizontal line are perpendicular to each other. This information is important for the optimization of the function. Thus, this method works well under the non-Manhattan world assumption. However, the orthogonality of the estimated vanishing points using this method is not proven because this method is only designed for their detection. In addition, this method requires a few tens of minutes to 1 h for optimizing the energy function to estimate the vanishing points.

An algorithm for line clustering and vanishing point detection was proposed in [17] as the previous work. This method uses a J-linkage-based method for clustering of the detected and filtered line segments. In the method, hypotheses for vanishing points are generated from the clusters The objective function can detect robust vanishing points in a non-Manhattan world because this method reflects the corresponding assumption in the geometry parsing. However, it also do not guarantee that the detected vanishing points are orthogonal to each other because this objective function uses a heuristically selected zenith point in the minimization phase.

3. PRIOR KNOWLEDGE

3.1. Notations

All the vectors are represented in a homogeneous coordinate system. Therefore, point x in 2D coordinates is represented in lower case and italics as $x = (x_1, x_2, x_3)^{\mathrm{T}}$, where $x_3 = 1$; hence, $\tilde{x} = (x_1, x_2)^{\mathrm{T}}$ in inhomogeneous coordinates. Similarly, in 3D coordinates, point X is represented in uppercase and italics as $X = (X_1, X_2, X_3, X_4)^{\mathrm{T}}$, where $X_4 = 1$; hence, $\tilde{X} = (X_1, X_2, X_3)^{\mathrm{T}}$ in inhomogeneous coordinates. Vanishing point v can be calculated as cross product $\mathrm{l_i} \times \mathrm{l_j}$ of two lines $\mathrm{l_i}$ and $\mathrm{l_j}$. Line l can be calculated from two end-points e_1 and e_2 of its line segment s by cross their product $e_1 \times e_2$. The orthogonal distance between x and l in the Euclidean space is $\mathrm{d}(\mathrm{l}, x) = |\mathrm{l}^{\mathrm{T}} x / (x_3 \sqrt{\mathrm{l}_1^2 + \mathrm{l}_2^2})|$. The measurement of the image-based consistency between vanishing point v and line segment s is $\mathrm{dist}(\mathrm{s}, v) = \mathrm{d}(e_1, \bar{e} \times v)$ as suggested in [13], where $\bar{e}$ is the average point between the two end-points of segment s.

3.2. Mapping onto Gaussian sphere

A line or line segment in 3D space can be mapped onto a Gaussian sphere as a large circle [7, 18]. Line segment S existing in the 3D space (hence, line L) can define a plane passing through two end-points $\tilde{E}_1$, $\tilde{E}_2$ and the center of the Gaussian sphere (typically, origin $O = (0, 0, 0, 1)^{\mathrm{T}}$). The plane meets the Gaussian sphere as a large circle. Therefore, a normal vector can represent the large circle, and (1) can calculate the normal vector.

$$\varphi = \frac{\widetilde{E_1} \times \widetilde{E_2}}{\left|\widetilde{E_1}\right| \left|\widetilde{E_2}\right|}, \quad \varphi = (\varphi_x, \varphi_y, \varphi_z)^{\mathrm{T}}, \tag{1}$$

where φ is a large circle. This is illustrated in Fig. 2. If the focal length is known and the principal axis is the center of an image, then the intersection point is easily detected as a vanishing point using the Hough transformation when the large circles meet at one intersection point.

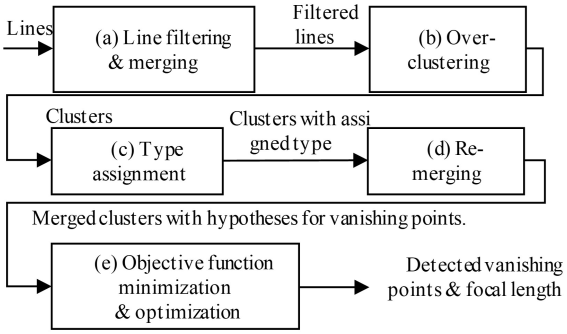

Lines
(a) Line filtering & merging
Filtered lines
(b) Over-clustering
Clusters
(c) Type assignment
Clusters with assi gned type
(d) Re-merging
Merged clusters with hypotheses for vanishing points.
(e) Objective function minimization & optimization
Detected vanishing points & focal length

Fig. 2. System overview of the proposed method.

3.3. Line clustering using J-linkage

J-linkage is a method that clusters the hypotheses having high similarities using the Jaccard distance for the measurement of the similarity. The line clustering method are conducted by these step: i) M hypotheses are generated for the vanishing point by intersecting random two-line segments. ii) $N \times M$ preference matrix P is generated as a Boolean type, where N is the number of line segments. iii) Value 1 is assigned to $\mathrm{P_{ij}}$ if $\mathrm{dist}(\mathrm{s_i}, v_{\mathrm{j}}) \leq \phi (= 2)$, otherwise it is 0, where $0 \leq \mathrm{i} < N$, $0 \leq \mathrm{j} < M$. Each generated row is a representation of each line segment over the hypotheses for the vanishing points (i.e., columns) as a set. iv) Then each set as the representation of a line segment is merged with another set by using the Jaccard distance. Jaccard distance $J(\cdot, \cdot)$ is defined as

$$J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}, \tag{2}$$

where A and B are any sets represented by any two-line segments, respectively. The Jaccard distance measures the similarity between A and B, such that more similar a pair, the closer it is to zero. Therefore, the line clustering method using J-linkage performs the clustering of the line segments until the Jaccard distance between all the representations of the line segments reaches unity.

4. METHODS

4.1. System overview

Fig. 2 shows overview of the proposed system. We utilize line segments detected by Line Segment Detector (LSD) [19] for the vanishing point detection. In the proposed system, first, the detected line segments are filtered and re-merged to avoid using erronous lines by the proposed line filtering and merging method as preprosessing shown in Section 4.2 and 4.3 respectively. Second, the filtered line segments are initially clustered by a J-linkage-based clustering method, and the initial clusters are fragmented to remove the outlier clusters and re-clustered to tune the clusters more precisely by the proposed over-clustering method in Section 4.4. Then, third, the reclustering is performed by assigning each type according to the characteristic of each cluster and re-merging each of the clusters by each type as explained in Section 4.5 and 4.6. Fourth, in Section 4.7, the vanishing points are found by minimizing the proposed objective function designed to ensure the orthogonality constraints. The detected orthogonal vanishing points are optimized by using the maximum likelihood estimation with the Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) [20] method, which is a nonlinear optimization method proven to exhibit a good performance for non-smooth optimizations. Finally, the calibration matrix is estimated by utilizing the estimated orthogonal vanishing points. We explain more details on the following Sections.

4.2. Line filtering

LSD line detector detects lots of useful line segments in linear time without any parameter tuning. An example of the line segments extracted by LSD is shown in Fig. 3(a). As shown in the image, numerous line segments are extracted. These line segments are useful indicators for vanishing point detection, but as many numerous outliers also appear. The outliers hinder the accurate detection of vanishing points. Therefore, it is necessary to address the effect of occurrence of outliers.

When the line segments are drawn as a histogram of their lengths, most of the results show that they tend to make an f-distribution [21] in practice, as shown in Fig. 4(a). We observe that the line segments that represent a structure of the scene in the image well are relatively long, but their number is small. In contrast, the line segments that can be found in eco environments are short and form numerous outliers. Thus, we should determine an appropriate threshold for removing the line segments whose lengths are shorter than the threshold and act as outliers. To this end, we used the kernelized regression [22] method. We approximate the histogram of the lengths by a Gaussian kernel with the 20th regression, and the threshold δ is detected by the maximum value of the numerical quadric differential of the approximated histogram as an inflection point. Fig. 4(b) shows the result of applying the kernelized regression to the histogram shown in (a) with the determined threshold value. The threshold value is automatically detected by taking maximum value of quadratic function as in around 25.

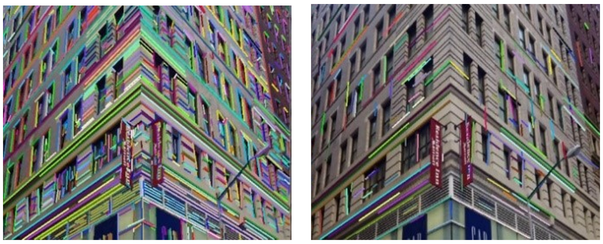
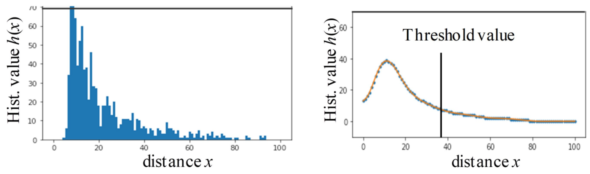

(a) (b)

Fig. 3. Change in the line segments. (a) Detected whole line segments. (b) Filtered line segments by the proposed method.



(a) (b)

Fig. 4. Graphs for line filtering. (a) A histogram of the lengths of the detected lines. (b) An approximated histogram and detected threshold value.

4.3. Line merging

Line segments cut from a line may be defects in the detection of vanishing points. For example, these line segments yield idle points as the intersection points located at infinity. These may cause confusion at the stage of hypotheses generation for vanishing points with unnecessary computation.

Therefore, we use the line merging method suggested in [23] for merging these line segments. This line merging method uses Gaussian mapping of the line segments mentioned in Section 3.2. In the implementation of this method, we assume the focal length to be the width as used in [12] because we do not know the actual focal length. After the image center is set on the optical axis in the 3D coordinate system, end-points e_1^i, e_2^i of each line segment in the image coordinates are set as $\tilde{E}_1^i$, $\tilde{E}_2^i$ in the 3D coordinates. The corresponding depth is the assumed focal length, $\hat{f}$ (i.e., $\tilde{E}_1^i = (e_{11}^i, e_{12}^i, \hat{f})^T$, where $i \in$ index for the line segments). Then, the normal vector of each large circle from the two end-points is calculated by (1).

Now, the line segments are represented as unit vectors in the 3D space, with each being a normal vector. For all the line segments, if the angle between any two-unit vectors is less than θ $(= 1°)$, the two line segments are added as nodes to the strongly connected graph [24]. After this execution is performed for all the line segments, each of the strongly connected set of nodes in the graph is merged into a single line segment. Therefore, Fig. 3(b) shows the detected and merged line segments that represent the structure of the image well without any segmented line segments as one line.

4.4. Over-clustering

The line segments undergoing the line filtering and merging stage are initially clustered via J-linkage. For J-

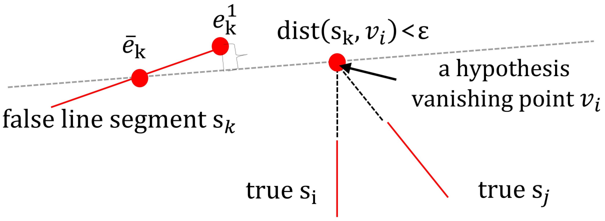


Fig. 5. Case of errors that occurred during the use of the J-linkage.

linkage, as described in Section 3.3, we generate the hypotheses for the vanishing points by selecting any two line segments for which the angle between their normal vectors of the large circles is at least less than ϑ (= 5°). We adopt this procedure rather than using two random line segments for M. This method definitely generates the candidate vanishing points as the hypotheses because it is on the Gaussian sphere, any two line segments meet the condition of being approximately parallel with an improvement in the performance. In contrast, using two randomly selected line segments generates an intersection point as a hypothesis, which is unrelated to a vanishing point.

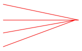
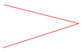
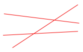
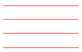
Concurrently, type 2 errors [25] in the form of misclustering occur when the clustering is performed by J-linkage using the generated Mhypotheses, as shown in Fig. 5. These errors found in the generation of preference matrix P. The reason for the occurrence of these errors is that a false line segment that should not be clustered with some cluster gets clustered because it satisfies dist(s_i, v_j) $\leq \phi$, as shown in like Fig. 5. To address this problem simply, we use a sufficiently small threshold number, ε, for dist(s_i, v_j) $\leq \varepsilon$. Therefore, the type 2 error can be prevented and numerous line clusters are generated more skillfully.

4.5. Type assignment

Numerous clusters are created by the over-clustering step. However, some clusters are divided even they should be the same cluster as called type 1 error [25]. To address this problem, we suggest a re-clustering step of the fragmented clusters by assigning specific types for each cluster and merging the clusters according to the types. First, in this sub-section, we present the proposed assignment method with suggested criterions. Prior to this, we observe the characteristics of the clusters, as listed in Table 1. In practice, most of the clusters follow the shapes that represent the characteristics presented in Table 1.

According to the table, based on the shapes there are five types. Type 0 represents an inlier cluster that all the line segments intersect to one point. Type 1 is a cluster that has only one line segment. Similar to type 1, type 2 is a cluster that has two line segments that pass through exactly one intersection point. Type 3 is an outlier cluster in which the line segments meet at multiple intersection points. Last, type 4 is the cluster in which all the line segments are parallel.

Table 1. Various types.

Type	Shape	Type	Shape
Type 0		Type 1	
Type 2		Type 3	
Type 4			

The method assigning each type to each cluster is as follows: Types 1 and 2 are simply assigned according to the number of line segments in a cluster. Types 0, 3, and 4 are assigned by using the mean line segment in each cluster because it represents the tendency of the line segments in a cluster. Therefore, type 0 is assigned to a cluster if the mean line segment in the cluster passes through the intersection point from any two line segments in the cluster, i.e., dist($\bar{s}$, x) $< \varepsilon$ where the $\bar{s}$ is the mean segment line and x is the intersection point. For the assignment of type 4, we use the distribution of the intersection points generated by the mean line segment with all the line segments. Because the intersection points from parallel lines are located at infinity, the standard deviation from the intersection points is large. Therefore, we assign type 4 if the standard deviation of a cluster of intersection points between the mean line segments and all line segments is larger than σ (= 3000). A cluster that does not satisfy any of these conditions is assigned as type 3 as an outlier.

The clusters that each of the types is assigned generate the hypotheses for the vanishing points. The clusters of types 0 and 4 use the intersection point of any two line segments as the hypotheses in each cluster, and the clusters of type 2 use a single intersection point in each cluster as the hypothesis. Because the clusters of type 3 are outliers, they do not create the hypotheses and are excepted for the next phases.

4.6. Re-merging

The clusters that each type is assigned according to their characteristics are re-merged by the different criteria obtained from the characteristics again.

The clusters that have a mean line and hypothesis for the vanishing point similar to the clusters of type 0 or type 4 are merged into one cluster when the mean line of a cluster intersects the hypothesis for the vanishing point of another cluster, and vice versa. Thus, when the types of any two clusters C_i and C_j are type 0 and type 4, type 0 and type 0, or type 4 and type 4, they are merged if the orthogonal distance error between the mean line of a cluster and hypothesis of another cluster and vice versa is

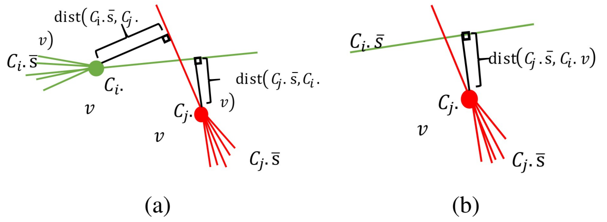

dist($C_i.\bar{s}$, C_j.
v)
$C_i.\bar{s}$
C_i.
v
dist($C_j.\bar{s}$, C_i.
v)
C_j.
v
$C_j.\bar{s}$
(a)
$C_i.\bar{s}$
dist($C_j.\bar{s}$, $C_i.v$)
C_j.
v
$C_j.\bar{s}$
(b)

Fig. 6. Illustrations for the remerging phase. (a) The case of remerging when the two types of clusters are type 0 and type 4, type 0 and type 0, or type 4 and type 4. (b) The case of remerging when the two types of are type 1 and type 4 or type 1 and type 0.

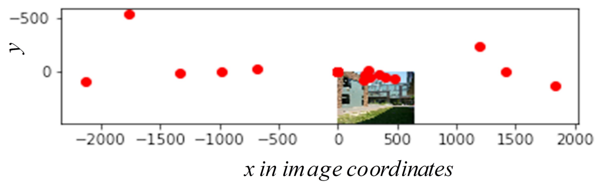


Fig. 7. An example of Detected the hypotheses for vanishing points. The red dots indicate the hypotheses generated by each re-merged clusters.

less than ε when using the root mean square error as

$$\sqrt{\frac{1}{2}(\text{dist}^2(\bar{\text{s}}_i^C,\ v_j^C)+\text{dist}^2(\bar{\text{s}}_j^C,\ v_i^C))}<\varepsilon, \tag{3}$$

where $\bar{\text{s}}_i^C$ and $\bar{\text{s}}_{\text{j}}^C$ are the mean line segments of C_i and C_j, respectively, v_i^C and v_j^C denote the hypotheses for the respective vanishing points, and $0<i<j\leq K$, where K is the number clusters. Fig. 6 (a) illustrates (3).

A type 1 cluster has to be included in a cluster if the cluster intersects the same hypothesis of another type 1 cluster. Thus, when any two C_i and C_j have type 1 for C_i and type 0 or type 4 for C_j, they are merged into a cluster if they satisfy the following equation:

$$\text{dist}(\bar{\text{s}}_i^C, v_j^C)<\varepsilon. \tag{4}$$

Because a type 1 cluster does not have the hypothesis for the vanishing point, (4) reflects one side of the orthogonal distance error, as shown in Fig. 6(b).

Type 2 clusters typically have only one exact intersection point, so that the hypotheses of this type clusters are directly used for the objective function minimization without any re-merging process. All the remerged clusters follow the type of front cluster as own type. Fig. 7 shows the detected hypotheses for vanishing points of each remerged clusters.

4.7. Objective function minimization for calibration

We briefly describe how to compute camera calibration through known orthogonal vanishing points. Then, we introduce the proposed objective function minimization for robust detection of orthogonal vanishing points.

Given three vanishing points, a camera intrinsic matrix K can be estimated by decomposing IAC $\boldsymbol{\omega}$ [5] such that $(\text{KK}^{\text{T}})^{-1}$ by the Cholesky factorization. To obtain $\boldsymbol{\omega}$, following orthogonality constraints should be satisfied:

$$v_1^{\text{T}}\boldsymbol{\omega}v_2=v_1^{\text{T}}\boldsymbol{\omega}v_3=v_2^{\text{T}}\boldsymbol{\omega}v_3=0, \tag{5}$$

where v_{i} (i = 1, ..., 3) denotes vanishing points.

Concurrently, assuming pixels of an image are squared and the principal point of the image is the center, $\boldsymbol{\omega}$ is simplified as diag$(1/f^2,\ 1/f^2,\ 1)$ matrix, where f is a focal length. Then, camera calibration matrix K_f becomes:

$$\text{K}_f=\begin{bmatrix} f & 0 & 0\\ 0 & f & 0\\ 0 & 0 & 1\end{bmatrix}. \tag{6}$$

The only unknown parameter is the focal length, which can be estimated by two vanishing points under one constraint [26] as follows:

$$f=\sqrt{\frac{v_{11}v_{21}+v_{12}v_{22}}{-v_{13}v_{23}}}. \tag{7}$$

If focal length f is known with K_f, then another vanishing point that satisfies the orthogonality constraint can be easily estimated by using two other vanishing points as

$$v_3=\text{K}_f\left(\left(\text{K}_f^{-1}v_1\right)\times\left(\text{K}_f^{-1}v_2\right)\right). \tag{8}$$

To detect robust vanishing points we propose three terms related to geometric constraints. Note that $\{\text{s}\}$ is the given set of all the line segments, and v_i and v_j are the vanishing points that we are searching, for which $0<i\leq j<\text{I}$, where I is the number of hypotheses for the vanishing points. For the comprehension with following explanation of each term, we illustrated a figure in Fig. 8.

The first term, $L(v_i,v_j\mid\{\text{s}\})$, examines the validity of the hypotheses for the vanishing points as true vanishing points by measuring the image-based consistency with regard to all the line segments. This implies that the hypotheses are true vanishing points and many line segments intersected them. Therefore, the equation of the first term is as follows:

$$L(v_i,v_j\mid\{\text{s}\})=1-\frac{1}{N}\sum_{\tau\in\{i,j,k\}}\sum_{\forall\text{s}_t\in\{\text{s}\}}\text{f}(v_\tau,\text{s}_t), \tag{9}$$

$$\text{f}(v_\tau,\text{s}_t)=\begin{cases}1, & \text{dist}(\text{s}_t,v_\tau)<\rho,\\ 0, & \text{else},\end{cases} \tag{10}$$

where N is the number of line segments and ρ is a threshold parameter for the measurement of the intersection.

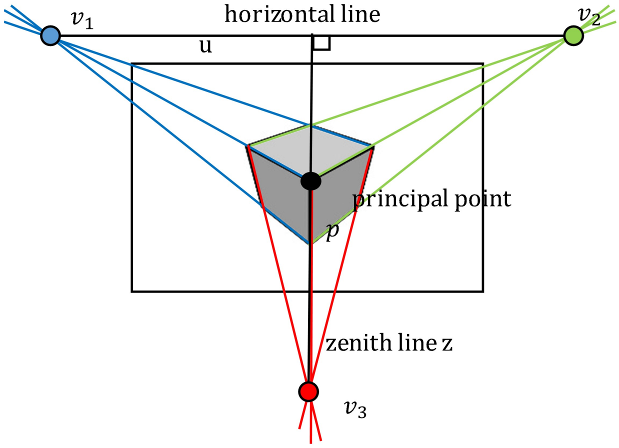


Fig. 8. The illustration for comprehension of the objective function.

Function f$(\cdot,\cdot)$ returns 1 if a line segment intersects a hypothesis for a vanishing point, otherwise it gives 0 for all the line segments over the three hypotheses for the vanishing points. Then, all the elements calculated by f$(\cdot,\cdot)$ are summed and divided by N for normalization. The reason for subtracting one is to yield a smaller value when more many line segments are intersected.

The second term, $J(v_i, v_j \mid \{s\})$ calculates the dissimilarity between the two hypotheses for vanishing points v_i and v_j. If the two hypotheses are true vanishing points, nothing can intersect both the hypotheses simultaneously, except the horizontal line. Therefore, using (2), we calculate the Jaccard distance for the calculation of the dissimilarity in the two consensus sets calculated by two hypotheses v_i and v_j, respectively, given $\{s\}$, which is discussed in Section 3.3, step iii. The second term yields that a smaller number of the line segments intersecting two hypotheses simultaneously implies a lower cost.

Third term $\phi(v_i, v_j)$ regularizes the geometric consistency used in the geometric parsing [9]. When assuming that the principal point is the center of the image if the principal point is unknown, then the vertical line connecting the zenith point and principal point is perpendicular to the horizontal line. To calculate this constraint, in the previous work, the zenith point is selected heuristically such that the zenith point is a hypothesis for a vanishing point in a cluster of type 4. This includes their slope is larger than 10°. Thus, the zenith point selected by the heuristic approach cannot surely meet the orthogonality constraint with other vanishing points. In this work, the zenith point is used by the v_k generated by (8), which satisfies the orthogonality constraint. Therefore, zenith line $\mathrm{z} = v_\mathrm{k} \times p$, $\mathrm{z} = (\mathrm{z_a}, \mathrm{z}_b, \mathrm{z}_c)^\mathrm{T}$, where p is the principal point, and the perpendicular line of the zenith line is $\mathrm{z}_\perp = (-z_b, z_a, c)^\mathrm{T}$, where c is a constant. Then, $\phi(\cdot,\cdot)$ calculates $\tan|\theta_\mathrm{u} - \theta_{\mathrm{z}_\perp}|$, where horizontal line $\mathrm{u} = v_\mathrm{i} \times v_j = (\mathrm{u}_a, \mathrm{u}_b, \mathrm{u}_c)^T$, θ_u is the angle of u, and $\theta_{\mathrm{z}_\perp}$ is the angle of $\mathrm{z}_\perp$ along the x-axis from the origin on the counter clockwise. The calculation of the tangent gives an effect that a larger difference in the angle between two angles, the larger is the cost exponentially.

Therefore, the proposed objective function is as follows:

$$\begin{aligned}\mathrm{E}(v_i, v_j|\{\mathrm{s}\}) =& n_c \cdot L(v_i, v_j \mid \{\mathrm{s}\})^2 \\ &+ n_{js} \cdot J(v_i, v_j \mid \{\mathrm{s}\})^2 + n_{hor} \cdot \phi(v_i, v_j)^2,\end{aligned} \tag{11}$$

where the $\mathrm{n_c}, n_{js}$, and n_{hor} are the coefficients for each term. The two vanishing points v_i, v_j are detected by

$$\underset{i,j \in \mathrm{I}}{\mathrm{argmin}}(\mathrm{E}(v_i, v_j \mid \{\mathrm{s}\})), \tag{12}$$

and another vanishing point $v_k$is automatically calculated by Equation (8) with focal length f by (7), hence camera calibration matrix K_f. We input the hypotheses in order of $0 < i < j \leq \mathrm{I}$ to the objective function so that its minimization is very fast.

To refine these detected vanishing points with the focal length, we defined the log-likelihood mixture model for the vanishing points as the following equation based on the Cauchy distribution [27] suggested in [10]:

$$P(\mathrm{s_k} \mid \Psi) = \sum_{\mathrm{i}=1}^{3} \theta_\mathrm{i} P(\mathrm{s_k} \mid v_i(\Psi)) + \theta_4 P(\mathrm{s_k} \mid O), \tag{13}$$

$$P(\mathrm{s_k} \mid v_i) = \frac{1}{\pi}\left(\frac{\gamma}{\mathrm{dist}(\mathrm{s_k}, v_i)^2 + \gamma^2}\right), \tag{14}$$

where $\Psi = \{f, \mathrm{R}\}$, R is a rotation matrix constructed by three vanishing points, which is represented by exponential maps [28]. The rotation matrix $\mathrm{R} \in \mathrm{SO}(3)$ can be estimated as

$$\begin{aligned}\mathrm{R} = &\left(\pm\frac{\mathrm{K}_f^{-1} v_1}{\parallel \mathrm{K}_f^{-1} v_1 \parallel}, \pm\frac{\mathrm{K}_f^{-1} v_2}{\parallel \mathrm{K}_f^{-1} v_1 \parallel}, \pm\frac{\mathrm{K}_f^{-1} v_3}{\parallel \mathrm{K}_f^{-1} v_3 \parallel}\right) \\ &\text{s.t. } \det(\mathrm{R}) = 1.\end{aligned} \tag{15}$$

$P(\mathrm{s_k} \mid O)$ is the term for the outliers. $\Theta = \{\theta_{\mathrm{i}=1,\ldots,4} \mid \theta_\mathrm{i} \geq 0 \wedge \sum_{i=1}^{4} \theta_\mathrm{l} = 1\}$ includes the mixture coefficients. γ is the scale parameter for the likelihood function, and $\mathrm{s_k} \in \{\mathrm{s}\}$. Note that the vanishing points from Ψ can be estimated as $v_\mathrm{i}(\Psi) = \mathrm{K}_f r_\mathrm{i}$ [28] where the r_i is the i-th column of the rotation matrix.

We optimized the model by maximizing the likelihood using BFGS [29] as

$$\Psi^* = \mathrm{argmax}_{\Psi,\Theta} \sum_k \log P(s_k \mid \Psi, \Theta). \tag{16}$$

Therefore, the optimization phase refines vanishing points with focal length regard to accurate camera calibration matrix.

5. EXPERIMENT

5.1. Environment setting

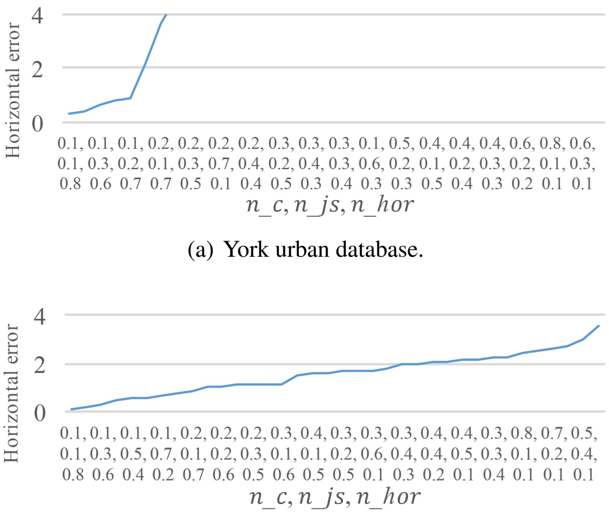
We implement this work using python and tested the experiments on i7 CPU, 8G RAM. We use the York Urban Database that has 102 images taken by assuming the Manhattan world and Eurasian Cities Database that has 103 images captured by assuming the non-Manhattan world. In this work, we compare *prev_work* suggested in the previous work [17] to compare improvement, and *R4L_MLE_Rf* (our implementation) that is suggested to obtain self-calibration using vanishing point detection. We call the proposed works to *our* and *our_opt* which is to use optimization. Other vanishing points methods are not compared due to they are not suggested for estimating camera calibration matrix. Also, we omit the comparison of the geometric parsing [9] because this method is extremely slow. We except comparisons of any deep learning approaches such as Vpgnet [30], because this system is considered to be used for embedded system without any GPUs.

Parameters setting: All the methods use the line segments that are detected by the LSD line segment detector with line filtering. For line filtering, δ is automatically estimated as a value between 20 and 30, and the number of filtered line segments are between 500 and 1000 with good representation of the scene.

The parameters for *R4L_MLE_Rf* are used according to $K = 500$ and $\varepsilon = 0.5$, as suggested in [10]. The parameters for the previous work are set as $n_c = 0.2$, $n_{js} = 0.3$, and $n_{hor} = 0.5$ for the objective function suggested in [17]. In the proposed method, we use $M = 500$ and $\varepsilon = 0.01$ for over-clustering and remerging, and $\gamma = 1$ as suggested in [10], $\rho = 0.5$ also used in previous work for the objective function to make more precise clusters than γ, $n_c = 0.1$, $n_{js} = 0.1$, and $n_{hor} = 0.8$ as tested in Fig. 9. Fig. 9(a) and (b) show the sorted total horizon errors based various combinations, where the sum of each combination is one in the York Urban Database and Eurasian Cities Database. In Fig. 9, a combination of these parameter values for the coefficients of the objective function shows a minimized error compared to other combinations of both these databases. This implies that the constraint of geometric consistency that the zenith line be perpendicular to horizontal line when assuming the principal point as the center of the image is the core factor for the accurate detection of vanishing points.

5.2. Measurement

We test four measurements for the focal length error, horizon error, zenith point error, and speed using the accumulative histogram. All the methods are affected by the random samples, and so we tested all the methods over ten times in the measurements. First, the focal length error is measured by subtracting the estimated focal length to the ground truth focal length by taking the absolute value. For the test, we only use the York Urban Database because the ground truth focal length is only provided from this database. To estimate the focal length, the three vanishing points must satisfy the orthogonal constraints between each other to yield a positive-definite matrix for IAC $\boldsymbol{\omega}$. However, the previous work cannot meet the constraints and cannot calculate $\boldsymbol{\omega}$, and hence, the focal length. Therefore, we only compare *R4L_MLE_fR, our*, and *our_opt,* which can estimate the focal length by satisfying the orthogonal constraints, except *prev. work*.



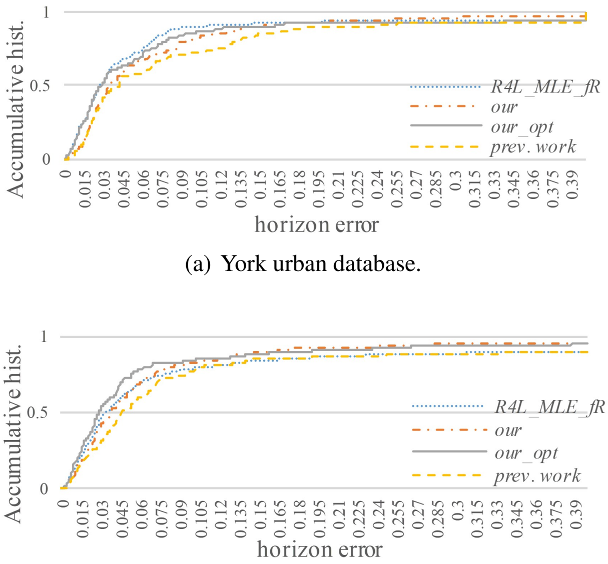
(a) York urban database.

(b) Eurasian cities database.

Fig. 9. Parameter searches for the proposed methods by comparison of the horizontal errors according to the combinations of n_c, n_{js}, and n_{hor} in different database, York Urban Database and Eurasian Cities Database.

Measurement of the horizon and zenith point error: The measurements of the horizon error and zenith point error are for the accuracy of the detection of the vanishing points. The horizon error measures the differences between the ground truth horizontal line and estimated horizontal line by both the estimated vanishing points that are located along the horizon.

The horizon error is calculated by $\max|\mathrm{u}^*(x) - \widehat{\mathrm{u}}(x)|/(\text{height of image})$ where $\mathrm{u}^*(x)$ is the y-axis value of the ground truth horizontal line over the x-axis value on the image, $\widehat{\mathrm{u}}(x)$ is the y-axis value of the estimated horizontal line on the x-axis value, too. It means that the horizon error measures the longest distance normalized by the height between $\mathrm{u}^*(x)$ and $\widehat{\mathrm{u}}(x)$ in the image. The zenith point error is calculated by the differences in the angle between the unit vector of ground truth zenith point mapped onto the unit sphere and the unit vector of the estimated zenith point mapped onto the same unit sphere.



(a) York urban database.

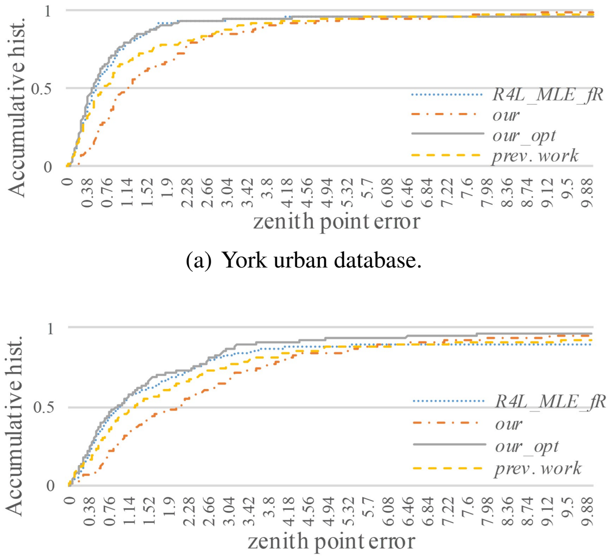
(b) Eurasian cities database.

Fig. 10. Comparison of performances on accumulative histograms of the horizontal errors between ours with other methods.

Figs. 10(a) and (b) show the results of the accumulative histogram of the horizontal error in the York Urban Database and Eurasian Cities Database, respectively. Moreover, Figs. 11(a) and (b) display the results of the accumulative histogram of zenith point error in the York Urban Database and Eurasian Cities Database, respectively. In Fig. 10(a), the best performance is *R4L_MLE_fR*, but the performance of *our_opt* is very similar in the York Urban Database. The next best performance is of the proposed method and then of the previous work. However, in Fig. 10(b), *our_opt* is the best performance outperforming other methods in the Eurasian Cities Database. The results of the zenith point error illustrated in Fig. 11(a) and (b) are similar to the results of the horizon error. In the York Urban Database, *our_opt* is the best performance as shown in Fig. 11(a). In the Eurasian Cities Database, *our_opt* outperforms *R4L_MLE_fR* slightly.

Measurement of focal length error. The experimental results on the focal length error are illustrated in Fig. 12(a). The test shows that the *our_opt* outperforms *R4L_MLE_fR,* whereas *our* slightly degrades the performance versus *R4L_MLE_fR*. The main reason for the occurrence of the focal length errors, except the inaccurate estimation of the vanishing points, is that the real principal point is not center of the image. Nevertheless, these evaluations show that the assumption that the principal point is the center of the image estimates the calibration matrix quite well.

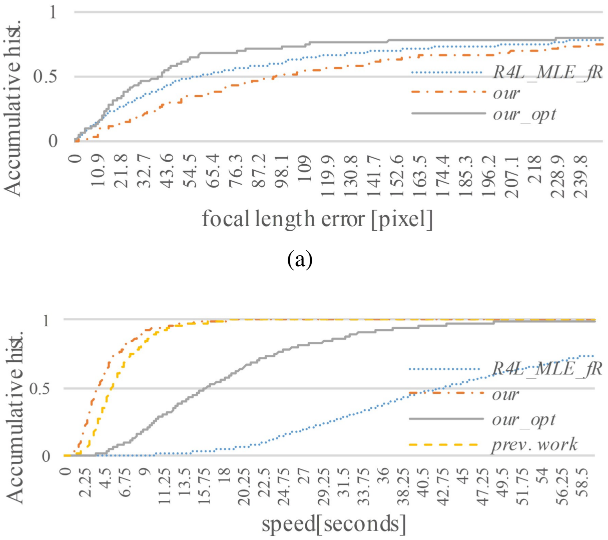
Comparison of speed. Last, we draw the accumulative histogram of the speed for both the databases together. Consequently, as illustrated in Fig. 12(b), *our* is the fastest method with a mean of 4.81 s. The previous method is the



(a) York urban database.

(b) Eurasian cities database.

Fig. 11. Comparison of performances on accumulative histograms of the zenith point errors between ours with other methods.



(a)

(b)

Fig. 12. Comparison of performances on accumulative histograms between ours with other methods in both of York Urban Database and Eurasian Cities Database. (a) Accumulative histogram of the focal length error in the York Urban Database. (b) Cumulative histogram of the speed in the York Urban Database and Eurasian Cities Database.

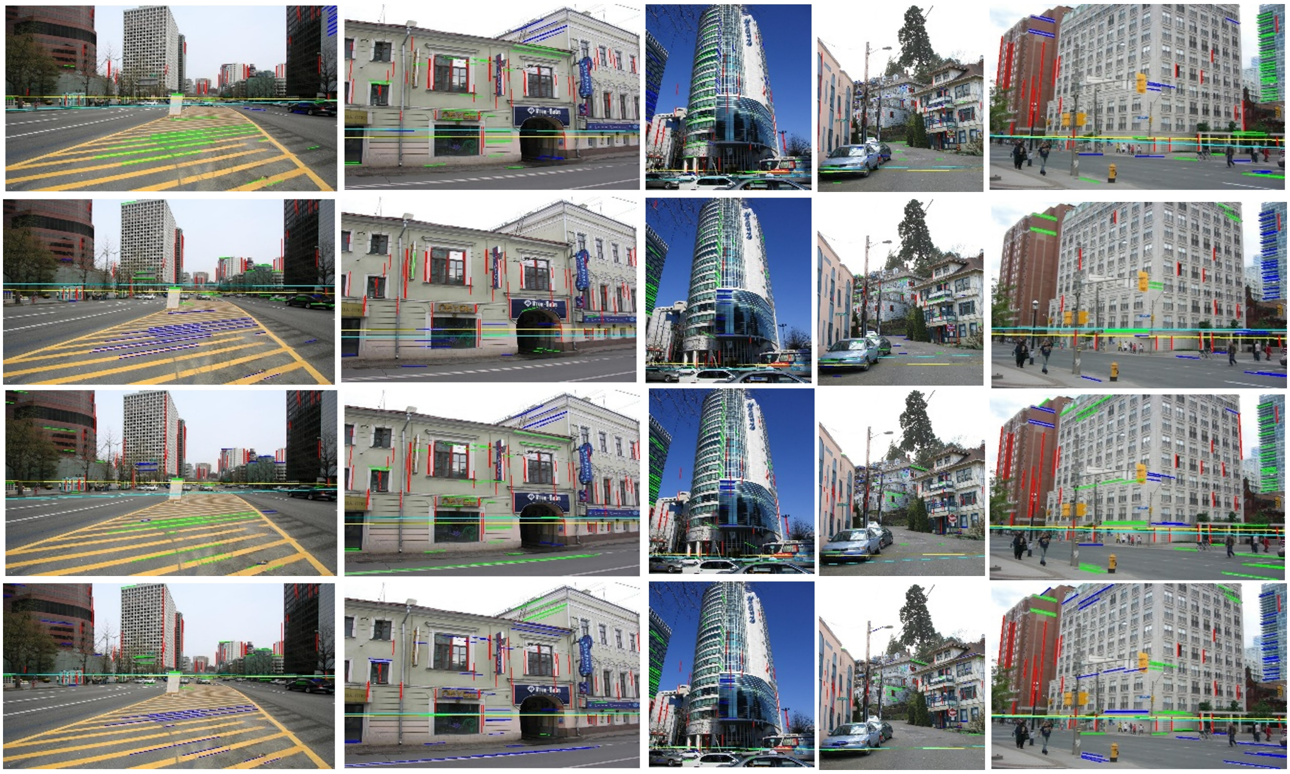


Fig. 13. Examples of all methods. The first row images are prev. work, the second row images are our, the third row images are *R4L_MLE_Rf*, and fourth row images are *our_opt*. The yellow line is the horizontal line of ground truth, and the cyan line is the estimated horizontal line.

next best with a mean of 6.32 s. *out_opt* is slower than for these two methods with a mean of 18.55 s. It is much faster than *R4L_MLE_fR* having a mean of 47.20 s. The proposed methods are within several seconds of each other. Because code of the proposed system is not optimized, if we optimize this code in C++, we expect an improvement of speed as mean 0.02 s for *our* and mean 0.08 s for *our_opt* as experimented in [31].

5.3. Analysis

Fig. 13 shows the examples of the results for which the proposed method is better than the other methods. It also gives a better performance in the Eurasian Cities Database for both the horizon error and zenith point error, with a more improved speed. Moreover, the proposed method is the best for the focal length estimation.

The results exhibit that the proposed method is robust to the non-Manhattan and Manhattan world assumptions because the hypotheses generation based on J-linkage using the proposed method, over-clustering, type assignment, and re-merging is lighter and more robust to be the true hypotheses than the RANSAC-based method.

In addition, because the proposed objective function is designed by satisfying the orthogonality constraints for auto-camera calibration and non-iteratively minimization, the proposed method is very fast and powerful to estimate the vanishing points and camera calibration matrix. By parameter searching and the assumption used in [9], which is robust to the non-Manhattan world, the proposed method is robust to the non-Manhattan world as well as the Manhattan world.

The RANSAC-based method must include the true hypotheses within the K-th trial. Therefore, it depends on random sampling and may give a different result in each term. However, J-linkage based on the initial clustering is suggested as a non-iterative clustering method for high speed. It yields constant results even it is also affected by sampling. Therefore, the suggested clustering method rectifies the influences of the sampling.

6. CONCLUSION

This paper presents a fast and accurate detection of vanishing points as well as camera calibration matrix. This method operates accurate line clustering for detecting hypotheses for vanishing points by fragmenting and re-clustering. The proposed objective function estimates the vanishing points that minimize the objective function non-iteratively from the hypotheses with the optimization of log-likelihood maximization. The proposed objective function is designed to reflect the orthogonality constraints in these hypotheses and detect robust vanishing points by measuring the image-based consistency regarding all the line segments and several geometric relations between the hypotheses. Therefore, the proposed method exhibits good performance in terms of the speed and accuracy, outperforming the other methods with both the Manhattan and non-Manhattan world assumptions. We believe that the proposed method can be useful for the camera pose estimation such a SLAM system as an auto-navigation system in real-time.

REFERENCES

[1] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, 1996.

[2] G. Zhang and I. H. Suh, "A vertical and floor line-based monocular SLAM system for corridor environments," *International Journal of Control, Automation, and Systems*, vol. 10, no. 3, pp. 547-557, 2012.

[3] V. Hoang and K. Jo, "Automatic calibration of camera and LRF based on morphological pattern and optimal angular back-projection error," *International Journal of Control, Automation, and Systems*, vol. 13, no. 6, pp. 1436-1445, 2015.

[4] Y. Shin, J.-H. Park, J.-H. Bae, and M.-H. Baeg, "A study on reliability enhancement for laser and camera calibration," *International Journal of Control, Automation, and Systems*, vol. 10, no. 1, pp. 109-116, 2012.

[5] R. Hartley and A. Zisserman, *Multiple View Geometry n Computer Vision*, Cambridge University Press, 2nd ed, New York, NY, USA, 2003.

[6] J. M. Coughlan and A. L. Yuille, "The Manhattan world assumption: regularities in scene statistics which enable Bayesian inference," *Proc. NIPS*, Denver, CO, USA, pp.809-815, 2000.

[7] S.T. Barnard, "Interpreting perspective images," *Artificial Intelligence*, vol. 21, no. 4, pp. 435-462, Nov. 1983.

[8] J. P. Tardif, "Non-iterative approach for fast and accurate vanishing point detection," *Proc. of 12th IEEE Conf. International Conference on Computer Vision*, Kyoto, Japan, pp. 1250-1257, 2009.

[9] O. Barinova, V. Lempitsky, E. Tretiak, and P. Kohli, "Geometric image parsing in man-made environments," *Proc. of European Conference on Computer Vision*, Springer, Berlin, Heidelberg, Germany, pp. 57-70, 2010.

[10] H. Wildenauer and A. Hanbury, "Robust camera self-calibration from monocular images of Manhattan worlds," *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2831-2838, 2012.

[11] R. Toldo and A. Fusiello, "Robust multiple structures estimation with j-linkage," *Proc. of European Conference on Computer Vision*, Berlin, Heidelberg, Germany, pp. 537-547, 2008.

[12] J. K. Jana and W. Zhang, "Video compass," *Proc. of European Conference on Computer Vision*, Berlin, Heidelberg, Germany, pp. 476-490, 2002.

[13] M. E. Antone and S. Teller, "Automatic recovery of relative camera rotations for urban scenes," *Proc. of IEEE Conf on Computer Vision and Pattern Recognition*, Hilton Head, SC, USA, pp. 282-289, 2000.

[14] Y. Xu, S. Oh, and A. Hoogs, "A minimum error vanishing point detection approach for uncalibrated monocular images of man-made environments," *Proc. of IEEE Conf. on IEEE, Computer Vision and Pattern Recognition (CVPR)*, Portland, Oregon, USA, pp. 1376-1383, 2013.

[15] G. Nebehay and R. Pflugfelder, "A self-calibration method for smart video cameras," *Proc. of 12th IEEE International Conference on Computer Vision*, Kyoto, Japan, 2009.

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

[17] S. J. Lee and S. S. Hwang, "Fast and robust vanishing point detection on uncalibrated images," *Proc. of IEEE International Conference on Image Processing*, Athens, Greece, 2018.

[18] J. C. Bazin, Y. Seo, C. Demonceaux, P. Vasseur, K. Ikeuchi, I. Kweon, and M. Pollefeys, "Globally optimal line clustering and vanishing point estimation in manhattan world," *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Rhode Island, USA, pp. 638-645, 2012.

[19] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, G. Randall, "LSD: a line segment detector," *Image Processing On Line*, vol. 2, pp. 35-55, Mar. 2012.

[20] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley & Sons, New York, Jul. 2000.

[21] A. V. Lazo and P. Rathie, "On the entropy of continuous probability distributions," *IEEE Transactions on Information Theory*, vol. 24, no. 1, pp. 120-122, Jan. 1978.

[22] E. A. Nadaraya, "On estimating regression," *Theory of Probability and its Applications*, vol. 9, no. 1, pp. 141-142, Sep. 1963.

[23] S. J. Lee and S. S. Hwang, "A robust line merge method using Gaussian mapping," *Proc. of Conf. Image Processing and Image Understanding*, Jeju, Korea, 2018.

[24] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed., MIT Press and McGraw-Hill, Cambridge, Massachusetts, Sec. 22.5, pp. 552-557, USA, 2001.

[25] P. S. Mann, *Introductory Statistics*, 9th ed., John Wiley & Sons, Hoboken, New Jersey, USA, 2016.

[26] B. Caprile and V. Torre, "Using vanishing points for camera calibration," *International Journal of Computer Vision*, vol. 4, no. 2, pp. 127-139, Mar. 1990.

[27] W. Feller, *An Introduction to Probability Theory and Its Applications*, 2nd ed., John Wiley & Sons Inc., New York, pp. 704, Jan. 1991.

[28] J. C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 63-81, Jan. 2012.

[29] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley & Sons, New York, Jul. 2000.

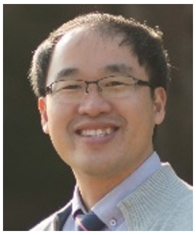
[30] S. Lee, J. Kim, J. S. Yoon, S. Shin, O. Bailo, N. Kim, T.-H. Lee, H. S. Hong, S.-H. Han, and I. S. Kweon, "VPGNet: Vanishing point guided network for lane and road marking detection and recognition," *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pp. 1965-1973, 2017.

[31] S. B. Aruoba and J. F. Villaverde, "A comparison of programming languages in macroeconomics," *Journal of Economic Dynamics and Control*, vol. 58, pp. 265-273, Sep. 2015.

[32] K. Sirisantisamrid, K. Tirasesth, and T. Matsuura, "A determination method of initial camera parameters for coplanar calibration," *International Journal of Control, Automation, and Systems*, vol. 7, no. 5, pp. 777-787, 2009

[33] D. Kang, J. Ha, and M. Jeong, "Detection of calibration patterns for camera calibration with irregular lighting and complicated backgrounds," *International Journal of Control, Automation, and Systems*, vol. 6, no. 5, pp. 746-754, 2008.

Sang Jun Lee received his B.S. degree in Computer Science and Engineering from Handong Global University, Pohang-si, Korea, in 2017. He is currently pursuing an M.S. degree in the Dept. of Information Technology at the Handong Global University. His research interests include the SLAM system for the localization of self-driving cars, robotics, or augmented reality, and optimization of these technologies using machine learning.



Sung Soo Hwang received his B.S. degree in Electrical Engineering and Computer Science from Handong Global Unveristy, Pohang, Korea in 2008, and his M.S and a Ph.D. degrees in Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2010 and 2015, respectively. From 2015 to now, he has been an Assistant Professor with School of Computer Science and Electrical Engineering, Handong Global University, Pohang, Korea. His research interests include image-based 3D modeling, 3D data compression, augmented reality.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.