2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

A Novel Line Matching Method Based on Intersection Context

Hyunwoo Kim and Sukhan Lee

Abstract— **This paper presents a novel line matching method based on the intersection context of coplanar line pairs especially working in poorly textured indoor scenes. To overcome the matching ambiguity of single line segments, intersecting line pairs in 2D images are utilized for line matching. Coplanarity of the intersecting line pairs and their corresponding intersection context discriminate true intersecting line pairs from false intersecting ones in 3D world. Compared to previous approaches, the proposed method can match line segments and estimate camera geometry simultaneously not knowing camera geometry in advance or not considering topological relations of all line segments. Comparison studies and experimental results prove the accuracy and speed of the proposed method in real world situations.**

I. INTRODUCTION

3D scene modeling is an active and important research field in computer vision and graphics, with applications including in TV/film production, augmented reality, robotics, navigation, and surveillance systems. In scene modeling, the detection and matching of image features is the first crucial step because given feature correspondence among multiple views 3D scenes are reconstructed by triangulation-based formulations [1], [2].

Most scene modeling techniques have been developed assuming interest points [3], *e.g.,* corners, are detected and matched using the photometric characteristics and invariance such as color, shape, and textile of richly textured regions, and they can only applied to well-structured and richly textured scenes, which contains texture information to be used for feature extraction and matching. However, in real world situations, scenes are sometimes unstructured and even contain poorly textured environments and objects, including tables, desks, chairs, sinks, refrigerators, microwave ovens, monotone walls, and hallways.

In those cases, interest points are hardly detected and poorly localized because of lack of texture information [4]. Therefore, line features are good candidates as image features because man-made objects are mostly constructed and modeled by 3D lines, and in poorly textured indoor scenes, 2D lines can be the only image feature sometimes to be utilized for automatic 3D modeling [1], [5], [6], [2].

While lines are robustly detected and localized under environmental change, they are difficult to match because of lack of photometric invariance to be used for measuring similarity. Conventionally, most of line matching methods have been studied based on the assumption of known camera geometry, *i.e.,*the geometric relationship among multiple views. Given the known camera geometry, which is estimated from interest point matches or is calibrated using calibration pattern beforehand, the matching candidates are constrained on the epipolar lines. So the matching problem converted into the matching problem between the points on the lines.

Schmid and Zisserman [7] automatically matched line segments by exploiting the intensity neighborhood of the line segments, guided by the epipolar constraints between different camera views, to provide point-to-point correspondences along the line segments. Resolving the resulting ambiguity, Werner and Zisserman [8] improved the previous algorithm by a 'line sweep', or search to register the photometric neighborhood. Those approaches presume that accurate camera geometry should be estimated beforehand. When those methods are applied to poorly textured scenes, the detection of interest point is not easy and camera geometry estimation is inaccurate, resulting the failure of the algorithms. Therefore, the need exists for an automatic line matching algorithm without predetermined camera geometry, possibly estimating camera geometry and matching line segments simultaneously.

Recently, several reported investigations pioneered methodologies for matching line segments without assumed camera geometry in unstructured real world situations. Bay *et al.* [9] proposed a line matching algorithm targeting poorly textured scenes. First, an initial set of line segment correspondences are obtained by comparing the histograms of the neighboring color profiles in both views. Then a topological filter is applied to find correct line matches while removing wrong candidate matches for the initial matches. Kim *et al.* [10] introduced a spectral line matching algorithm to find the subset of correspondences with the greatest consistency, which are learned using logistic classifiers. Those approaches require heavy computation via topological relation analysis or a learning stage. Besides, the photometric information for only single line segments is not discriminative enough to match line segments.

Another group of researchers has utilized junction features for line matching. Vincent and Laganière [11] matched junctions by estimating the local perspective distortion between the neighborhoods of junctions, then estimated the

This work is supported in part by the Intelligent Robotics Development Program, one of the Frontier R&D Programs funded by the Ministry of Knowledge Economy (F0005000-2009-31), in part by the KORUS Tech Program administered by the KOTEF (Korea Industrial Technology Foundation) with the fund provided by the Ministry of Knowledge Economy, and in part by the MEST (Ministry of Education, Science and Technology), Korea, under the WCU (World Class University) Program supervised by the KOSEF (Korea Science and Engineering Foundation) (R31-2008-000-10062-0).

Hyunwoo Kim is with the department of new media of Korean German Institute of Technology, Seoul, Korea, hwkim@kgit.ac.kr

Sukhan Lee is with the school of information and communication engineering of Sungkyunkwan University, Suwon, Korea. (Corresponding author, Email: lsh@ece.skku.ac.kr)

fundamental matrix based on a constrained minimization, assuming crude camera pose estimates. Bay *et al.* [12] identified polyhedral junctions resulting from the intersections of the line segments, then segmented the images into planar polygons using an algorithm based on a binary space partitioning tree. However, junction-based approaches are only applied to well-structured scenes, where lines and the junctions are robustly extracted. Such scenes are limited to houses built by brick or tiles, well-structured indoor aisles, and well-edged furniture.

To address the challenge of line matching in poorly textured scenes, we used the intersection context of line features by combining the geometrical invariance of 3D line intersection with the photometric invariance of the projected 2D line intersection. Instead of matching single line segments individually, we matched a pair of line segments coplanar in 3D. The coplanarity of the intersecting line pairs and their corresponding intersection context discriminate true intersecting line pairs from false intersecting ones in 3D world.

The novelties of the proposed method are three-fold. (1) First, in feature extraction, the intersection of line pairs is newly modeled as an image feature, which has a favorable localization property and captures geometrically meaningful structures with fair photometric invariance, for instance corners and junctions of furniture or electronic appliances (Sec. II). (2) Second, the feature matching stage performs camera geometry estimation and line matching simultaneously without predetermined camera geometry, so that the error of camera geometry estimation is not propagated into the consecutive line matching step (Sec. III). In addition, in many real world situations, the camera calibration cannot be easily applied, so the camera geometry should be estimated. They include the situations when only the monocular camera is available or allowed due to hardware constraints or cost, when the camera zooms in/out to focus on the interested objects/scenes, and when the camera rigs themselves rotate and translate on the mechanical platform to actively monitor the environments. (3) Last, a practical and fast solution is presented when camera geometry is given. Compared to the previous approaches based on single line feature invariance and topology, the computational speed of the line matching method is much faster, and the chance of miss-matching is lower (Sec. IV).

In Sections II and III, feature extraction and matching techniques are described, respectively. Section IV introduces a fast and simplified version for the rectified stereo images, and Section V presents the comparison studies and experimental results in real world scenes including line-based scene modeling. Finally, Section VI concludes with a discussion of the performance of the line matching method reported herein and future investigation.

II. Feature Extraction

To solve the line matching problem, coplanar line pairs are considered as matching features instead of single line segments. Two line segments can produce more constraints for matching by combining the individual similarity, but the simple combination of the similarity of two line segments does not improve the matching result much. The geometric relation and photometric invariance of coplanar line pairs warrant investigation.

For line pairs or groups of lines, special 3D geometric relations can be observed and measured in 2D image space: parallelism, orthogonality, and coplanarity. The first two, parallelism and orthogonality, are strong geometric constraints. The 2D projections of 3D parallel lines meet a vanishing point, and the set of the parallel lines with different directions are projected into images while constructing a vanishing line [2]. However, accurate and automatic computation of those features are not easy because the accuracy of the extracted line is limited by line quantization. The latter, coplanarity, is a rather weaker constraint, compared to the first two properties, but can be determined easily among line pairs and/or groups. A coplanar line group can be determined based on the inter-image homography between different views [12]. However, doing so without information about the cameras and/or structures can be difficult because homography alone is not only enough to discriminate coplanar lines pairs from non-coplanar line pairs accurately, and measuring coplanarity for all four line groups is also computationally expensive.

In this paper, aside from inter-image homography, coplanarity of intersecting line pairs is investigated. First, when two lines are coplanar in 3D, the projected 2D lines are also coplanar. Moreover, when the intersection of a 3D coplanar line pair is projected into 2D images, the projected intersection is also the intersection of the 2D line pairs that are projected from the 3D coplanar line pair. Therefore, the intersection of the coplanar line pair is geometrically invariant under perspective projection. That is, the geometric relationship among coplanar line pairs and their intersection are preserved among different views under perspective projection. However, determining coplanar line pairs from 2D line pairs still remains a problem. Any line pair in a 2D image can be either coplanar or non-coplanar. The lines of a coplanar line pair meet in 3D, but those of a non-coplanar pair do not. To discriminate a coplanar line pair from a non-coplanar line pair, the intersection of the lines of the pair is further considered in the following sections.

A. Determining Intersecting Line Pairs

The proposed algorithm begins with the extraction of line segments from each image using a Canny edge detector with hysteresis, followed by edge linking, with the linked edges fitted to the line segments. The extracted line segments in an image are represented by $\{\mathbf{l}_1, \cdots, \mathbf{l}_{k1}\}$, where $k1$ denotes the number of extracted line segments. Note that better line detectors may improve the performance of the proposed algorithm [13] in the cost of extra computation.

Due to noise and occlusion, the line segments intersecting in 3D do not explicitly meet in 2D images. Thus, to find candidates for intersecting line pairs, their end points of line segments are extended virtually. A proximity rule is used to prevent non-existing intersecting lines in 3D.

Given a line segment $\mathbf{l}_i$,, a line segment $\mathbf{l}_j$ is considered as an intersecting pair when at least one of the extended end points are within a specific threshold d_{th}. Formally, given the index set $\Pi(:\mathbb{N}\mapsto\mathbb{N}^2)$ of intersecting line pairs of the image, represented by $\pi(k)=(i,j)$, such that $d(\mathbf{l}_i,\mathbf{l}_j)<d_{th}$, where $d(\mathbf{l}_i,\mathbf{l}_j)$ denotes the distance between line segments $\mathbf{l}_i,\mathbf{l}_j$, the intersecting line pairs determined by the proximity rule are represented by $\mathscr{L}_{pair}=\{\{\mathbf{l}_{\pi(k)_1},\mathbf{l}_{\pi(k)_2}\},(\pi(k)_1,\pi(k)_2)\in\Pi\}$, where $k=1,\cdots,\#$ of Π.

B. Line Intersection Context Feature (LICF)

Now, a new image feature is defined based on the candidates of intersecting line pairs. In each line pair, the intersection context of the line pair is utilized as an image feature. The intersection context includes the intersecting position and the local texture region center at the position. The feature is called the "Line Intersection Context Feature (LICF)" in this paper. The newly defined feature contains geometric information, as well as photometric information. The former is the positional information of intersection computed from a line pair, and the latter is the region information of the local image patch centered at the intersection position. Formally, the LICFs are represented by the intersection positions and the corresponding region patches in an image, as follows: $\mathscr{F}\equiv\{\mathbf{x}_k,R(\mathbf{x}_k)\}=\{\mathbf{x}_k,R(\mathbf{x}_k),\mathbf{l}_{\pi(k)_1},\mathbf{l}_{\pi(k)_2}\}$, where $k=1,\cdots,\#$ of Π. $\mathbf{x}_k$ denotes the intersection positions of the corresponding intersecting line pairs $\mathscr{L}_{pair,k}(=\{\mathbf{l}_{\pi(k)_1},\mathbf{l}_{\pi(k)_2}\})$, and $R(\mathbf{x}_k)$ denotes the region patch centered at the intersection position $\mathbf{x}_k$. For convenience, a LICF ($\mathscr{F}_k$) sometimes refers to only the position ($\mathbf{x}_k$) instead of the set of the position and the neighboring region patch ($R(\mathbf{x}_k)$).

Fig. 1: (Left) An example of LICF. The line pair, colored in magenta and cyan, intersects in the circled position, with an intersection context bounded by the red box. (Middle) The zoom-in image of the intersection context. (Right) A model of LICF.

Figure 1 illustrates the model of the LICF and an example. The texture regions around LICFs have weaker texture characteristics, compared to Harris corners, a typical interest point features [3], [14]. To compare Harris corners and LICFs, eigen-analysis of their neighborhoods is performed [15]. While Harris corners are extracted from corners and junctions with mainly two large eigenvalues, LICFs include even plat areas whose intersection are covered by coplanar line pairs with wide-ranging eigenvalues. In addition, the interest point feature, e.g. Harris corners, and the LICFs are complimentary in detection and matching, so that the unification of both features can produce more informative image features.

The characteristics of LICFs can be summarized, as follows: (1) Good localization owing to derivation from the line pair intersection. (The line fitting is done in sub-pixels, so the LICFs have sub-pixel accuracies. It means that we do not need any extra fitting process as the interest points. In addition, the experiment results showed that its accuracy compared to Harris corners.) (2) Fair photometric invariance of the local patch centered at the intersection position, including junctions, corners, lines, and flat areas. (The features are 2D projections from 3D surface patches with photometric information, and the LICFs can be detected even in poorly textured regions, supported by coplanar line pairs, and in well-textured regions as the interest points.) (3) Fast line matching speed because of similarity measure only in the intersection context instead of all line segment context. (While the previous methods compare all the points on line segments [7] or the histogram of line profiles [9], the proposed method only compares the single intersection points. Therefore, the proposed method need less computation compared to the previous approaches.)

III. Feature Matching

To match the line intersection context features (LICFs) among different views, feature similarity is defined. In this stage, the LICFs projected from true 3D points need to be matched while prohibiting those from false 3D points, which do not exist in 3D world. While LICFs projected from true 3D points have a local texture region around the features with similar photometric properties under projective distortion among different camera views, ones projected from false 3D points do not. In this paper, region information about the intersection context of the LICFs is utilized for matching. The transformation of the local texture region among different views can be modeled as 2D projective transformation with scale, translation, rotation, and shearing under the assumption of local planarity.

For implementing a region descriptor of LICFs, any local feature descriptor can be used to measure the texture similarity under perspective distortion and illumination variation. They include the sum of square difference (SSD), sum of absolute distance (SAD), normalized cross-correlation (NCC) [16], color histogram, and scale-invariant feature transform (SIFT) [17], maximally stable extremal regions (MSER) [18] etc. Assuming that the perspective distortion is small in the local area, NCC is sufficient to describe the local texture invariance, which is known to be fast and robust despite moderate illumination change and small perspective distortion. Although NCC is thought to be adequate in the case of narrow-baseline stereo matching with small scale and rotation changes in local texture regions, in practice, NCC has been applied to match moderate wide-baseline stereo pairs, as demonstrated in Sec. V, owing to the characteristics of LICFs. In particular, NCC is a good similarity measure for scene modeling for off-the-shelf stereo cameras or fixed multiple cameras. However, matching in wide-baseline camera views with large perspective distortion can be achieved by using the scale and rotation invariant version of NCC [19]

or by incorporating more advanced local feature descriptors, such as SIFT and MSER with LICFs, which will be our future work, especially for general visual recognition.

For matching LICFs using NCC, the same process as with interest points is adopted [16]. After computing NCC scores from both images, the most strongly correlated matches with each other in both images are selected and represented by $\mathscr{M}^{init}_{\mathbf{x},\mathbf{x}'} = \{\mathbf{x}_k, \mathbf{x}'_k; \mathscr{L}_{pair,k}, \mathscr{L}'_{pair,k}\}, k = 1,...,K^{init}$, where $\{\mathbf{x}_k, \mathscr{L}_{pair,k}\} \in \mathscr{F}$ and $\{\mathbf{x}'_k, \mathscr{L}'_{pair,k}\} \in \mathscr{F}'$ and K^{init} denotes the number of matching LICFs by NCC [16], [19].

A. Matching Refinement

Not surprisingly, after the matching process based on the NCC, the results contain mismatches because of the limited discriminative power of the similarity measure; therefore, a refinement stage is crucial. While conventional line matching approaches assume known camera geometry, our approach is to estimate the line matching and camera geometry simultaneously by using LICFs. For 2-view, 3-view, and N-view, the camera geometry is represented up to projective transformation by fundamental matrix, trifocal tensor, and camera projection matrices, respectively [20]. In this paper, the results are shown for the 2-view case. Extension to 3-view and N-view is also possible in the same framework.

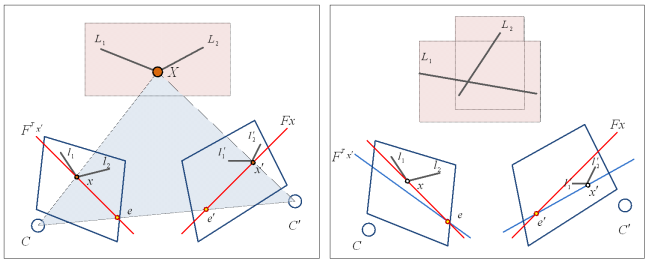


Fig. 2: LICF matching using epipolar constraint and coplanarity. (Left) Coplanar line pair case. (Right) Non-coplanar line pair case. C and e (C' and e') are the camera projection center and the epipole of the first (second) camera, respectively.

Figure 2 presents geometric relations of a LICF between two different views. One is the configuration of a coplanar line pair, and the other is that of a non-coplanar line pair in 3D. Given two camera views, the relations between cameras and 3D points/lines are constrained by epipolar geometry in 2D image space, and, based on the epipolar geometry, two configurations can be discriminated.

Given a coplanar line pair, $\mathbf{L}_1$ and $\mathbf{L}_2$, the intersection meets on the 3D point, $\mathbf{X}$, and these lines are located on the same plane (colored in pink). The projections, $\mathbf{x}$ and $\mathbf{x}'$, of the 3D intersection point into the two different views, $\mathbf{I}$ and $\mathbf{I}'$, are also the intersection points of the projected line pairs, $\{\mathbf{l}_1, \mathbf{l}_2\}$ and $\{\mathbf{l}'_1, \mathbf{l}'_2\}$, in 2D image space. Since the intersection points, $\mathbf{x}$ and $\mathbf{x}'$, computed from the projected line pairs, come from the true 3D point $\mathbf{X}$, they meet the epipolar constraint. That is, the 2D intersection point $\mathbf{x}$ in the first view is on the epipolar line $F^T\mathbf{x}'$, transferred from the corresponding point $\mathbf{x}'$ in the second view. $\mathbf{x}'$ is also transferred from $\mathbf{x}$ by the epipolar constraint $F\mathbf{x}$.

On the contrary, non-coplanar paired lines, $\mathbf{L}_1$ and $\mathbf{L}_2$, do not intersect in 3D. Lines $\mathbf{l}_1$ and $\mathbf{l}_2$, projected into the first view intersect at the point, $\mathbf{x}$, constructing a LICF. Also, for the second view, the projected lines, $\mathbf{l}'_1$ and $\mathbf{l}'_2$, intersect at $\mathbf{x}'$. Since neither LICFs back-projects into a real 3D point, neither meet the epipolar constraint or exist on the same epipolar plane. Although the LICFs are matched by NCC in the initial matching step, the LICFs from non-coplanar line pairs can be eliminated by testing whether they meet the epipolar constraints or not.

In the refinement stage, based on LICF correspondences between two different views, the fundamental matrix is estimated, and mismatches are removed using RANdom SAmple Consensus (RANSAC) [20]. The refinement stage using RANSAC is the same as the conventional ones except that the algorithm is applied to matching LICFs instead of matching interest points. To find the fundamental matrix giving the best matches, the fundamental matrix with maximum inliers is selected as a solution. The inlier matches LICFs for which the fitting error is within a user-defined threshold. The fitting error, given a fundamental matrix, is defined by the symmetric transfer error:

$$E_{trans}(\mathbf{x},\mathbf{x}') = \frac{1}{N_{trans}} \sum_{i=1}^{N_{trans}} d(\mathbf{x}'_i, F\mathbf{x}_i)^2 + d(\mathbf{x}_i, F^T\mathbf{x}'_i)^2 \quad (1)$$

where $d(\mathbf{x},\mathbf{y}) = (\mathbf{y}^T\mathbf{x})/\sqrt{\mathbf{y}_1^2 + \mathbf{y}_2^2}$, the distance of the point $\mathbf{x}$ from the line $\mathbf{y}$, and N_{trans} denotes the number of matching LICFs identified as inliers. The refined matching LICFs are represented by $\mathscr{M}^{refine}_{\mathbf{x},\mathbf{x}'} = \{\mathbf{x}_k, \mathbf{x}'_k; \mathscr{L}_{pair,k}, \mathscr{L}'_{pair,k}\}, k = 1,...,K^{refine}$, where $\{\mathbf{x}_k, \mathscr{L}_{pair,k}\} \in \mathscr{F}$ and $\{\mathbf{x}'_k, \mathscr{L}'_{pair,k}\} \in \mathscr{F}'$, and K^{refine} denotes the number of matching LICFs after the matching refinement.

B. Line Segment Matching

Given a matching LICF $(\{\mathbf{x}_k, \mathbf{x}'_k; \mathscr{L}_{pair,k}, \mathscr{L}'_{pair,k}\})$, matching between individual line segments ($\{\mathbf{l}_{\pi(k)_1}, \mathbf{l}_{\pi(k)_2}\}$ and $\{\mathbf{l}'_{\pi'(k)_1}, \mathbf{l}'_{\pi'(k)_2}\}$) from the coplanar line pairs ($\{\mathscr{L}_{pair,k}, \mathscr{L}'_{pair,k}\}$) must be resolved. To find the match of each line segment from the matched line pair, a sophisticated method based on oriented projective geometry can be used [21], [20] if the epiploar geometry and the matches of line pair sets are known. However, in this paper, selecting a line segment with smaller angle difference between the lines of a matched pair is sufficient, assuming small rotation change between views. This assumption is reasonable when a monocular camera is attached to a mobile platform that moves on the ground or stereo cameras are verging toward the same direction to capture depth information of a scene. Moreover, the method is robust even when the fundamental matrix estimation is erroneous.

The corresponding line pairs $\{\mathbf{l}_1, \mathbf{l}_2; \mathbf{l}_a, \mathbf{l}_b\} \in \mathscr{M}^{refine}_{\mathbf{x},\mathbf{x}'}$ between two camera views are assumed to be given. For a line segment $\mathbf{l}_1$ in the first image, the matching line segments with index $k \in \{a,b\}$ are found from

the matching line pairs in the second image by selecting the line segments with small angle differences using the following Equation: $k = min_j\{min(|\theta_1 - \theta_j|, |\pi - \theta_1 + \theta_j|)\}, j \in \{a,b\}$. The final matching lines are represented, as follows. $\mathscr{M}_{\mathbf{l},\mathbf{l}'} = \{\mathbf{lm}_k, \mathbf{lm}'_k\}, k = 1,...,K^{match}$, where $\{\mathscr{L}_{pair,k}; \mathscr{L}'_{pair,k}\} = \{\mathbf{lm}_{k1}, \mathbf{lm}_{k2}; \mathbf{lm}'_{k1}, \mathbf{lm}'_{k2}\} \in \mathscr{M}^{refine}_{\mathbf{x},\mathbf{x}'}$ denoting K^{match} the number of matching line segments while satisfying Equation III-B.

IV. Simplification by Known Camera Geometry

Known camera geometry, a simplified and fast algorithm is further proposed by rectifying the image pairs in advance [20]. It is suitable for real-time practical applications, such as robotics or video surveillance. For fixed stereo camera configurations, such as off-the-shelf stereo cameras and fixed multiple wide-baseline cameras, accurate camera geometry, including factors such as epipolar constraints, can be precomputed by calibration or manual correspondence, allowing rectified image pairs to be easily produced. When the rectified image pairs are given, the initial matching and the refinement matching are merged into one stage because camera geometry estimation cannot be omitted. The simplified algorithm, developed for the rectified stereo pairs by known camera geometry, is called "*the simplified version*," compared to the original proposed algorithm, called "*the full version*," can be used to perform line matching and camera geometry estimation at the same time.

The procedure is as follows: First, stereo image pairs are rectified using known camera geometry; then LICFs are extracted and matched from line segments. Thanks to the rectification, a matching LICF can be found on the same scan line in the other image among the LICFs. Among them, the LICF match with the highest NCC is selected as a match. The matching LICFs with high NCC scores are regarded as non-occluded features, and those with low NCC scores as occluded.

V. Experimental Results

The proposed method is first compared with the conventional interest-point-based matching method, involving Harris corners, by evaluating the number of (correct) matching features and the accuracy of the estimated fundamental matrix by the symmetric transfer error, defined in Equation 1. For fundamental matrix estimation, a RANSAC method based on the eight-point algorithm is implemented [20].

Test image pairs are from richly textured outdoor scenes because they include line features and corner features in the same scene, and they are collected through the Internet websites [22], [21]. In [22] and [21], narrow-baseline stereo image pairs and relatively wide-baseline ones are collected, respectively, as shown in Figure 3.

Overall, the comparison studies prove that the method proposed herein can match line features correctly, as did the interest-point-based method, in well-textured regions. Moreover, the proposed method works better in the poorly textured regions. More interestingly, correct line matching is observed even in the scenes and local regions with large perspective distortion. Apparently, the proposed method extracts geometrically meaningful structures in those scenes, resulting in a higher possibility for correct matching despite quite low NCCs for those regions.

Figure 4 shows the results from the scene "*apt*". Feature matching results of both Harris corners and LICFs are fairly correct and accurate while giving reasonable fundamental matrix estimation results. The difference is that, while Harris corners are detected and accurately matched more often for textured regions, such as apartments and white cars, LICFs occur at the junctions of balconies/bricks of the apartment and the parking lot lanes. In parking lot lanes, we observe matching performance under large perspective distortion, resulting from the close distance between the scene and the cameras. Looking at the details, a mismatch of LICFs occurs at # 21 when the repetitive textured LICFs were matched on the same epipolar lines, resulting in mismatches of the related individual line segment #24. Note that the end points of matching line segments are not determined here, and the equations of line matching need to be compared. The figures of our experimental results are best viewed in color and with PDF magnification.

Fig. 4: A comparison study results for the scene "apt". (First row) Images from the first view. (Second row) Images from the second view. In each column, original test images, matching line segments, matching LICFs, and matching Harris corners are drawn in order. The estimated epipolar constraints are overlaid with red lines.

For a small set of relatively widely separated image pairs provided in [21], we obtained correct matching results, although the proposed method is not designed for wide-baseline image matching, even in the scene for which the interest-point-based method fails. The scene "kampa" in Figure 5 is more challenging because image pairs have large perspective distortion. The proposed method gives correct matching results with correct and accurate epipolar geometry, while the interest-point-based method fails in this regard. As mentioned earlier, the reason is that the proposed method detects meaningful geometric structures by eliminating the less meaningful photometric features. However, the matching NCC scores in the wide-baseline stereo matching are low, compared to the narrow-baseline stereo matching; therefore, sophisticated handing of the widely separated matching must

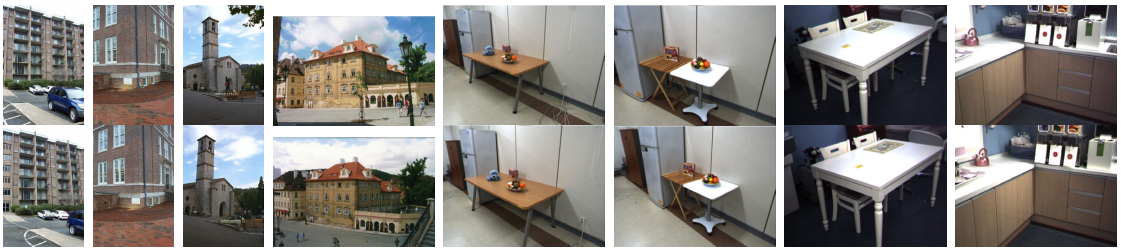


Fig. 3: Test images. In each column, the stereo image pairs of the scenes apt, tcorner, valbonne, kampa, lab1, lab2, table, and sink are shown, in that order.

be part of future work.

Lastly, the quantitative performance comparison is summarized in Table I in terms of the number of matching features and the accuracy of the estimated fundamental matrix. To evaluate the performance, we count the number of matching features after the RANSAC-based refinement, compared to the conventional matching evaluation manually counting correct matches. However, the difference is not large, *i.e.*, no more than 10% in our experiments. On average, including other experiments discussed in the next section, the number of matching Harris corners, LICFs, and line segments are 152.00, 63.13, and 57.75, respectively. Although more correct Harris corners matches can be achieved greater quantity, the proposed method (LICF/line, 0.32 pixels) is slightly better than the interest-point-based method (0.63 pixels) in terms of the accuracy of the estimated fundamental matrix evaluated using the symmetric transfer error.

A. Results on Poorly Textured Scenes

Now, the proposed method is applied to poorly textured scenes to show line matching performance in real world situations and to demonstrate the accuracy of the simplified version, compared to the full version. The image pairs are captured by off-the-shelf cameras, PointGrey's Bumble-bee stereo cameras (30 Hz images, 640×480 pixels), attached to a mobile platform. To simulate the unknown camera geometry cases, the general stereo image pairs are captured from the same off-the-shelf stereo camera yet at a different time frame for each pair.

The experimental results on poorly textured scenes show that the proposed method can detect and match most important geometric structures of poorly textured scenes, including tables, chairs, sinks, and electronics appliances, while the interest-point-based method fails to match and recover these geometric structures. In addition, when camera geometry is given, the simplified version is comparable to the full version, and runs fast enough for line matching and its 3D recovery in robotics applications.

Fig. 5: Comparison study results for the scene "kampa". Refer to Figure 4 for details.

Figure 6 is the result of the scene "lab1", which contain noticeable scale change and perspective effect due to forward camera motion into the objects. The experimental results show that the proposed method can detect and match most important structures, such as the edges and patterns of the tables, the box lines of the refrigerator, the vertical and horizontal lines of the walls, the textures/shadows on the table tops, and the lines of the floor. In those scenes, although the estimation of the fundamental matrix is not accurate due to the uneven distribution of features, most line segments are correctly matched due to simultaneous camera geometry estimation and line matching, with the exception of a few mismatches. Note that the end points of the matching line segments are determined by the estimated or known epipolar constraints in this section, such that the line segments can be reconstructed by recovering the end points and the degenerate configuration, where the end points of matching lines are inaccurate, is easily noticeable.

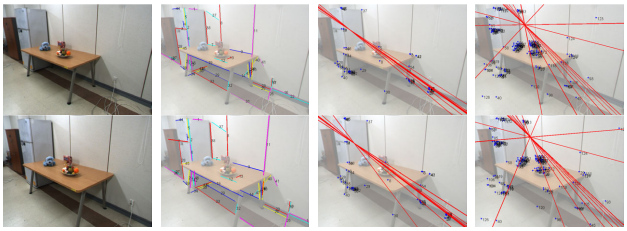


Fig. 6: Results in the poorly textured scene "lab1". Refer to Figure 4 for details.

Figure 7 is the stereo image pair that is captured by the off-the-shelf camera and rectified using the pre-calibrated camera parameters. Those scenes are acquired in a kitchen designed for the demonstration of real situation performance of personal robots, especially a robot's ability to serve foods and/or drinks to the elder. The scene "sink" includes many objects, such as a soda fountain, a tray, electronic appliances, and the sink. The general matching result is quite accurate except the mismatches of the matching line pairs $\{\#76, \#79\}$ and $\{\#46, \#91\}$. The first is from the mismatch of LICF due to similar texture areas, and the latter resulted from the failure of individual line segment matching based on relative

TABLE I: Quantitative comparison of fundamental matrix estimations.

		apt	tcorner	valbonne	kampa	lab1	lab2	table	sink	Avg (Std)
Harris	Initial	299	279	147	100	296	218	404	411	269.25 (110.95)
	Refined	202	179	99	29	126	106	229	246	152 (74.38)
	Transfer Error	0.19	0.20	0.30	3.32	0.30	0.25	0.26	0.24	0.63 (1.09)
LICF	Initial LICF	78	90	52	65	54	94	130	178	92.63 (42.78)
	Refined LICF	45	78	42	23	49	53	74	141	63.13 (36.04)
	Final line	53	70	34	34	48	49	67	107	57.75 (23.86)
	Transfer Error	0.22	0.15	0.31	0.81	0.32	0.24	0.25	0.26	0.32 (0.21)

line angle.

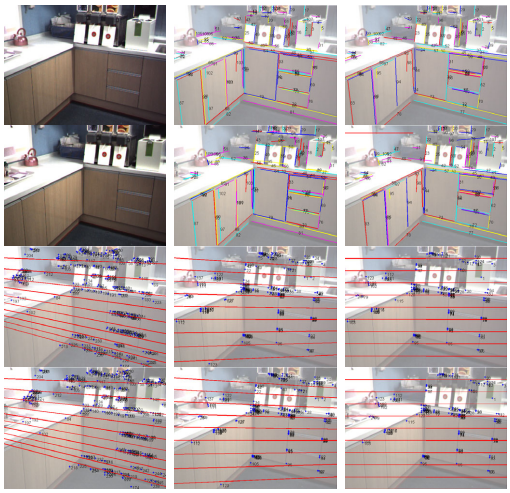


Fig. 7: Experimental results and comparison study for the scene "sink". (First and third rows) Images and results from the first view. (Second and fourth rows) Images and results from the second view. In each column of the 1st and 2nd rows are the original test images, matching lines when not using known epipolar geometry, and matching lines after rectification, in that order. In each column of the 3rd an 4th rows are matching Harris corners, the matching LICFs when not using known epipolar constraints, and matching LICFs after rectification, in that order, with their estimated epipolar constraints.

1) Experimental results of the Simplified version: Additionally, the full version, the simplified version, and the interest-point-based method are applied to the scenes in Figure 7, to compare their performance. Performance is evaluated in terms of the accuracies of the estimated camera geometry and matching results. When the full version and the interest-point-based method are applied, the known camera geometry is blinded and, instead, is estimated during the algorithms. The results show that the full version is slightly better than the simplified version in terms of the number of matched lines and fundamental matrix accuracy. The error in camera calibration seems to explain the difference. In the full version, the error does not affect the line matching results because the camera geometry, *i.e.,* the fundamental matrix, is simultaneously estimated. The full version captures more matching line segments (*e.g.*, # 6, #19, #66, and #98 in the scene "sink") than does the simplified version. For the interest-point-based method, the estimation of camera geometry is incorrect because of the poor textures in the scenes. Harris corner matching and fundamental matrix estimation are quite comparable in the scene "table," but are inferior in the scene "sink" (Figure 7) because Harris corners are not detected and matched in the bottom-left part of the image pair.

Line reconstruction: Furthermore, 3D structures are recovered from matching line segments using the calibrated camera parameters. Instead of direct line reconstruction, the end points of matching line segments are reconstructed [8], [20], as shown in Figure 8. In the scene "table", the rectangular shape of the tabletop and orthogonal structure between the tabletop and the legs of the table and the chair are modeled properly. In the scene "sink", the T-shape and the rectangular shapes of the sink are reconstructed correctly.

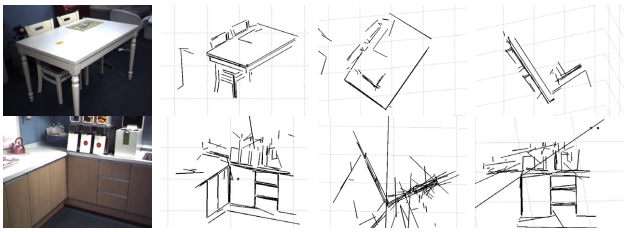


Fig. 8: Reconstructed line structure of the scenes "table" and "sink". The figures are the first view image as well the scenes viewed from the first camera, from the top, and from the side, in that order.

Speed: The simplified version of the proposed algorithm for the rectified stereo image pairs is implemented in C/C++ for fast processing in near real-time. The processing unit consists of a Pentium 4G Hz CPU and 2G Byte RAM. The line feature extraction module runs for 200 ms for both views, and the line matching module, including line pair determination, runs for 350 ms.

For the comparison purpose, color-histogram-based matching is implemented. The proposed method (350 ms) is faster than the color-histogram-based method (500 ms), conducting more numerous line matches and more accurate matching results. Although the color-histogram-based method needs more sophisticated stages to be used as a line matcher [9], already the accuracy and speed improvements by the proposed line matching method is significant. Note that Bay *et al.* [9] report that their implementation takes 8 seconds, on average, using Pentium 4 at 1.6GHz, excluding line detection.

2) Sensitivity analysis under orientation and distance variations: Finally, the matching results of LICFs and segments are presented when the orientation and distance varies between cameras and objects in order to ensure the stability and reliability of those changes. Figure 9 shows the matching results with respect to orientation and distance variation. The orientation varies from 0^0 to 90^0, with a step of 15^0, and the distance varies among $1.8m, 2.1m, 2.4m$, and $2.7m$. The results show that the matching is quite even when the orientation and the distance vary.

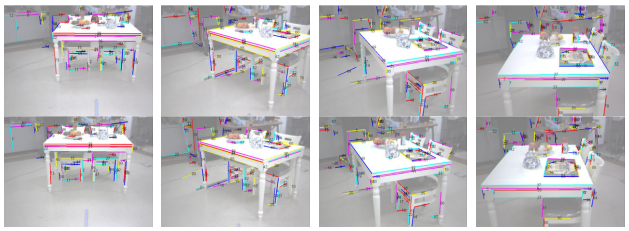
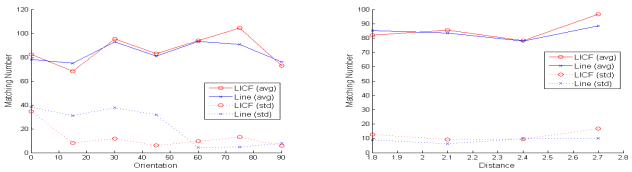


Fig. 9: Sampled line matching results with respect to varying orientation and distance between objects and cameras. In each column, the scenes with different distances and orientations between the long edge of the table and the stereo camera baseline: $\{2.7m, 0^0\}, \{2.4m, 30^0\}, \{2.1m, 60^0\}$, and $\{1.8m, 90^0\}$, in that order.

Figure 10 illustrates the numbers of matching LICFs and matching lines with respect to orientation and distance. These figures show that the matching performance is quite stable to orientation and distance variation. However, there is large variation in the standard deviation of the numbers because the background clutter is not eliminated, and the viewing volume for the table is not carefully handled.



(a) Orientation (b) Distance

Fig. 10: The number of matching LICFs and matching lines with respect to orientation and distance. The values shown are the average, and standard deviation.

VI. Concluding Remarks

In this paper, a novel line matching algorithm, based on the line intersection context feature, was presented. Experimental results showed that the performance is comparable and complimentary to that of interest-point-based matching. The proposed algorithm works well for poorly textured scenes, an area in which interest-point-based matching often fails. Simultaneous line matching and fundamental matrix estimation achieved correct line matching independent of the fundamental matrix estimation accuracy. The simplified version for rectified images using the pre-computed fundamental matrix also showed comparable results with near real-time speed.

References

[1] C. Baillard, C. Schmid, A. Zisserman, A. Fitzgibbon, and O. O. England, "Automatic line matching and 3d reconstruction of buildings from multiple views," in *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, IAPRS Vol.32, Part 3-2W5*, 1999, pp. 69–80.

[2] B. Micusik and J. Kosecka, "Piecewise planar city 3d modeling from street view panoramic sequences," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Location: Miami, FL, USA, 20-25 June 2009, pp. 2906 – 2912.

[3] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *Int. J. Comput. Vision*, vol. 37, no. 2, pp. 151–172, 2000.

[4] E. Kim, G. Medioni, and S. Lee, "Planar patch based 3d environment modeling with stereo camera," in *RO-MAN07*, Jeju island, Korea, August 26-29 2008, pp. 516–521.

[5] A. Bartoli and P. Sturm, "Multiple-view structure and motion from line correspondences," in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2003, p. 207.

[6] A. W. K. Tang, T. P. Ng, Y. S. Hung, and C. H. Leung, "Projective reconstruction from line-correspondences in multiple uncalibrated images," *Pattern Recogn.*, vol. 39, no. 5, pp. 889–896, 2006.

[7] C. Schmid and A. Zisserman, "Automatic line matching across views," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 0, p. 666, 1997.

[8] T. Werner and A. Zisserman, "New techniques for automated architectural reconstruction from photographs," in *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part II*. London, UK: Springer-Verlag, 2002, pp. 541–555.

[9] H. Bay, V. Ferrari, and L. Van Gool, "Wide-baseline stereo matching with line segments," in *CVPR05*, 2005, pp. I: 329–336.

[10] G. Kim, M. Hebert, and S.-K. Park, "Preliminary development of a line feature-based object recognition system for textureless indoor objects," in *Recent Progress in Robotics: Viable Robotic Service to Human (S. Lee, I. Hong Suh, M. Sang Kim (eds.))*. Springer-LNCIS, 2007, pp. 255–268.

[11] E. Vincent and R. Laganière, "Junction matching and fundamental matrix recovery in widely separated views," in *Proc. British Machine Vision Conf.*, London, UK, September 2004, pp. 77–86.

[12] H. Bay, A. Ess, A. Neubeck, and L. Van Gool, "3d from line segments in two poorly-textured, uncalibrated images," in *3DPVT06*, 2006, pp. 496–503.

[13] L. Wang, S. You, and U. Neumann, "Supporting range and segment-based hysteresis thresholding in edge detection," in *Proceedings of the International Conference on Image Processing (ICIP)*, San Diego, California, USA, 2008, pp. 609–612.

[14] C. Harris and M. Stephens, "A combined corner and edge detection," in *Proceedings of The Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[15] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.

[16] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[18] J. Matas, O. Chum, U. Martin, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," in *Proceedings of British Machine Vision Conference*, vol. 1, London, 2002, pp. 384–393.

[19] F. Zhao, Q. Huang, and W. Gao, "Image matching by multiscale oriented corner correlation," in *ACCV (1)*, 2006, pp. 928–937.

[20] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[21] T. Werner, "Lmatch: Matlab toolbox for matching line segments accross multiple calibrated images," 2007, `http://cmp.felk.cvut.cz/~werner/software/lmatch/`.

[22] B. Lloyd, "Computation of the fundamental matrix," 2003, `http://www.cs.unc.edu/~blloyd/comp290-089/fmatrix/`.