Novel Coplanar Line-Points Invariants for Robust Line Matching Across Views

Qi Jia, Xinkai Gao, Xin Fan[(✉)], Zhongxuan Luo, Haojie Li, and Ziyao Chen

School of Software, Dalian University of Technology, Dalian, China
{jiaqi,xin.fan,zxluo,hjli}@dlut.edu.cn,
{gaoxinkai,2013czy}@mail.dlut.edu.cn

Abstract. Robust line matching across wide-baseline views is a challenging task in computer vision. Most of the existing methods highly depend on the positional relationships between lines and the associated textures. These cues are sensitive to various image transformations especially perspective deformations, and likely to fail in the scenarios where few texture present. In this paper, we construct a new coplanar line-points invariant upon a newly developed projective invariant, named characteristic number, and propose a line matching algorithm using the invariant. The construction of this invariant uses intersections of coplanar lines instead of endpoints, rendering more robust matching across views. Additionally, a series of line-points invariant values generate the similarity metric for matching that is less affected by mismatched interest points than traditional approaches. Accurate homography recovered from the invariant allows all lines, even those without interest points around them, a chance to be matched. Extensive comparisons with the state-of-the-art validate the matching accuracy and robustness of the proposed method to projective transformations. The method also performs well for image pairs with few textures and similar textures.

Keywords: Line matching · Projective invariant · Characteristic number

1 Introduction

Feature matching is such a fundamental task in computer vision that it has found wide applications in photogrammetry, image mosaicking, and object tracking *etc.* [2,7]. Points and lines are prone to be mismatched due to illumination and viewpoint changes. In the last two decades, point matching methods have been well studied [11,14], while lines are not so popular as points due to the higher geometric complexity. Lines usually incorporate more semantic and structural information than points, and thus it is quite important to match lines in the scenarios where lines are abundant. The scenarios include 3D modeling and robot navigation in manmade scenes [6,13].

Most of existing line matching methods use texture information near lines as descriptors. Wang *et al.* [17] proposed a SIFT-like descriptor, the mean-standard

deviation line descriptor (MSLD). In many images, the textures in the vicinity of line segments are not rich enough to assemble an effective descriptor. These textures are also quite similar, possibly generating a less distinctive descriptor. Moreover, MSLD is sensitive to scale changes that is quite common in feature matching. Zhang *et al.* [18] utilize both local appearance and geometric attributes of lines in order to construct the line band descriptor (LBD). This method requires a global rotation angle between images, which is not always accurate. Texture based methods are typically sensitive to various image transformations, and may fail on images of low texture images and similar textures.

Some methods match a group of lines instead of giving each line a descriptor in order to obtain better performance in low texture images. In [15], line groups are matched through a feature named line signature (LS), and a multi-scale scheme is used to enhance the performance under scale changes. Unfortunately, this process is computationally expensive. López *et al.* [9] combined geometric properties with local appearance of a pair of lines and the structural context of their neighboring segments. Nevertheless, both methods highly rely on the endpoints of line segments. These endpoints are prone to be mismatched when their locations are not accurate due to various image transformations and partial occlusions.

Different from generating descriptors for lines or line groups, researchers resort to epipolar constraints or geometric invariants for line matching. Similar textures and inaccurate endpoints have less effects on these constraints or invariants. Lourakis *et al.* [10] used two lines and two points to build a projective invariant to match lines. However, this method can only work on images with a single plane. Al-Shahri *et al.* [1] exploited the epipolar geometry and coplanarity constraints between pairs of lines. This method performs well in wide-baseline images, but needs to estimate the fundamental matrix via the interest point correspondences. This estimation relies on the accuracy of the point correspondences, resulting in the chicken-and-egg dilemma. Fan *et al.* [4,5] provide two kinds of invariants based on the distance between matched feature points and lines. This method performs well under various image transformations. Again, it highly depends on the accuracy of the matched interest points.

In this paper, we propose a novel line matching method based on a newly developed projective invariant, named characteristic number (CN) [12]. Figure 1 sketches the work flow of the algorithm. As Fig. 2 shows, the line-points geometries in the neighborhood of each line construct our new line-points invariant upon CN robust to projective transformations. Hence, we are able to obtain well-matched neighborhoods as well as the homography between these neighborhoods. Finally, we incorporate more matched line pairs using this homography for line matching. The main contributions are:

(1) a new line-points projective invariant constructed on the intersections of coplanar lines that are more robust than those interest points matched upon textural information;
(2) a similarity metric between line neighborhoods given by a series of line-points invariant values less affected by mis-matched interest points;
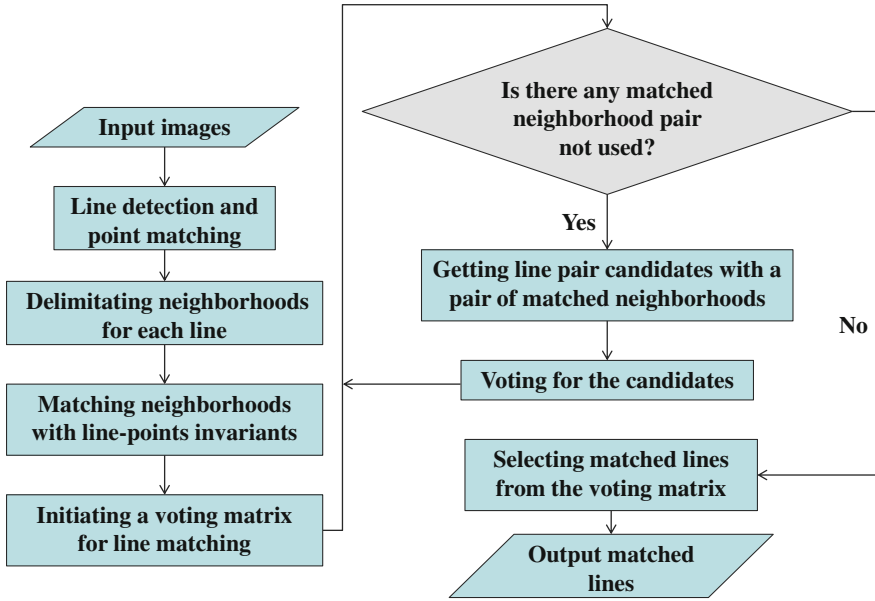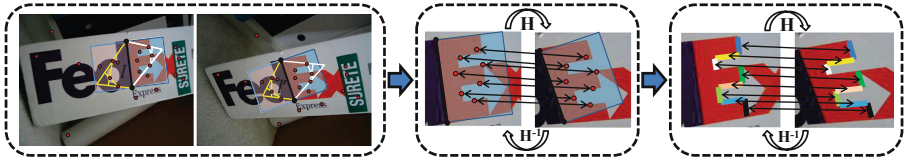
Fig. 1. Architecture of the proposed method

Fig. 2. Overview of matching neighborhoods and selecting line pair candidates. l and l' are a pair of lines; red dots represent the interest points (Color figure online)

(3) accurate homography between matched line neighborhoods recovered by the intrinsic coplanar attributes of the new line-points invariant. All the lines in image pairs have a chance to be matched. Thus, this strategy makes it possible to exploit more potential matched line pairs without interest points around them.

The rest of this paper is organized as follows. Section 2 introduces the projective invariant of characteristic numbers and from which the line-points invariant is derived. The line matching method is deliberated in Sect. 3. Experiments and results are reported in Sect. 4. Section 5 concludes the paper.

2 Line-Points Projective Invariant

In this section, a newly developed projective invariant characteristic number (CN) [12] is introduced into the construction of line-points invariant due to its geometric flexibility, and a new line-points invariant is constructed based on it.

2.1   Characteristic Number

Cross ratio [8] is one of the fundamental invariants in projective geometry and is widely used in computer vision applications [3,13]. The characteristic number (CN) extends the cross ratio in various respects, and reflects the intrinsic geometry of given points. Different from cross ratio, these points do not necessarily lie on lines. This property provides the flexibility to describe the underlying geometries by the points on and out of lines. We give the definition of CN below.

Definition 1. *Let $\mathbb{K}$ be a field and $\mathbb{P}^m$ be m-dimension projective space over $\mathbb{K}$, and $P_1, P_2, \ldots, P_r$ be r distinct points in $\mathbb{P}^m(\mathbb{K})$ that construct a close loop ($P_{r+1} = P_1$). There are n distinct points $Q_i^{(1)}, Q_i^{(2)}, \ldots, Q_i^{(n)}$ on the line segment $P_i P_{i+1}, i = 1, 2, \ldots, r$. Each point $Q_i^{(j)}$ can be linearly represented by P_i and P_{i+1} as*

$$Q_i^{(j)} = a_i^{(j)} P_i + b_i^{(j)} P_{i+1} \tag{1}$$

Let $\mathcal{P} = \{P_i\}_{i=1}^r$ and $\mathcal{Q} = \{Q_i^{(j)}\}_{i=1,\ldots,r}^{j=1,\ldots,n}$, the quantity

$$CN(\mathcal{P}, \mathcal{Q}) = \prod_{i=1}^{r} \left(\prod_{j=1}^{n} \frac{a_i^{(j)}}{b_i^{(j)}} \right) \tag{2}$$

is called the characteristic number of $\mathcal{P}$ and $\mathcal{Q}$.

2.2   Construction of Line-Points Projective Invariant

Interest points and lines are used to construct the line-points projective invariant. Generally, the most representative points of a line are the endpoints, but due to various changes between images, the line extraction methods usually cannot provide accurate endpoints. However, if two lines are located on the same plane, the location of their intersection to the object remains unchanged under projective transformation. We make a rough hypothesis that if the intersection of two lines is very close to one of the endpoints, the two lines are likely to be coplanar. Given a line l, suppose e is one of its endpoints. For all the intersections on line l, if o is the nearest intersection to e and the distance between o and e is smaller than $0.1 * length(l)$, o is chosen to substitute e as a key point of l. Only in the case that there is no such intersection available near an endpoint, the endpoint itself will be a key point. Further, in order to reduce pseudo intersections produced by collinear and parallel lines, we set a threshold for the angle between two lines. In our experiment, if the angle is not greater than $\pi/8$, their intersection will be abandoned. We also define the gradient of a line as the average gradient of all points on it. As shown in Fig. 3, two black arrows illustrate the gradient directions of lines a and b respectively. In order to keep rotation invariant, for line a, the area directed by the line gradient is denoted as $Right(a)$, and the other side is $Left(a)$. In a clockwise direction, the key point on the edge from $Left(a)$ to $Right(a)$ is denoted as KP_a^1 and the other is KP_a^2.

Fig. 3. The gradient directions and sides of lines
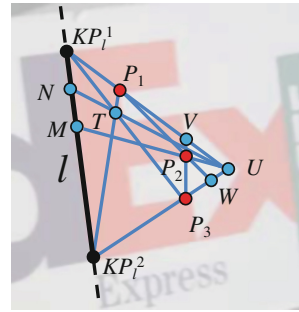
Fig. 4. Construction of line-points invariant.

We use five points to construct the line-points projective invariant. As shown in Fig. 4, KP_l^1 and KP_l^2 are two key points on line l, P_1, P_2, and P_3 are three non-collinear interest points on the same side of l. We denote the line through two points, P_i and P_j, as $\overline{P_iP_j}$ and the intersection of two lines, $\overline{P_iP_j}$, and $\overline{P_kP_m}$, as $<\overline{P_iP_j}, \overline{P_kP_m}>$. We can obtain several intersection points (blue dots), including $U = <\overline{KP_l^1P_1}, \overline{KP_l^2P_3}>$, $V = <\overline{KP_l^1P_1}, \overline{P_3P_2}>$, $W = <\overline{P_1P_2}, \overline{KP_l^2P_3}>$, $T = <\overline{KP_l^1P_3}, \overline{P_1KP_l^2}>$, $M = <\overline{KP_l^1KP_l^2}, \overline{UP_2}>$, and $N = <\overline{KP_l^1KP_l^2}, \overline{UT}>$. Thus we have $\triangle KP_l^1UKP_l^2$ with two points on every side. Thereafter, we are able to calculate CN with $\mathcal{P} = \{KP_l^1, U, KP_l^2\}$ and $\mathcal{Q} = \{P_1, V, W, P_3, M, N\}$. We denote the CN constructed in this way as $FCN(KP_l^1, KP_l^2, P_1, P_2, P_3)$.

3   Line Matching

In this section, a two-stage line matching algorithm is designed to obtain as many matched line pairs as possible with high accuracy. In the first stage, the similarities between line neighborhoods are calculated. In the second stage, the homography transformations between matched coplanar neighborhoods are calculated, creating some bases on which more matched line pairs can be exploited.

3.1   Similarity Between Line Neighborhoods

Neighborhood Definition. Line neighborhoods provide structural information around each line. In this paper, the neighborhood is determined by the length of the line to keep invariant to scale changes. As shown in Fig. 5, in the neighborhood of line l, the distance from any interest point to l is less than $\alpha * length(l)$ and less than $\beta * length(l)$ to the perpendicular bisector line. If point p is in the neighborhood of l, it is denoted as $p \in \mathcal{LPS}_l$. In our experiments, α is set as 2.0 while β is set as 0.5.

As many lines are formed by the intersection of planes, points located on different sides of a line may not be coplanar. Hence, the neighborhood is split
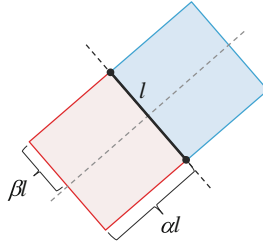
Fig. 5. The neighborhood of a line (Color figure online)

into left one and right one according to the gradient direction, which is detailed in Sect. 2.2 and Fig. 5. The left neighborhood is denoted as $\mathcal{LPS}_l^L$ and the right one is $\mathcal{LPS}_l^R$, which are represented in Fig. 5 in red and blue, respectively.

Neighborhood-to-Neighborhood Similarity Measure. Suppose there are two images, I and I', of the same subject from different views. The set of lines detected in each image is denoted as $\mathcal{L} = \{a_1, a_2, \ldots, a_n\}$ and $\mathcal{L}' = \{b_1, b_2, \ldots, b_m\}$. The matched interest points set in the two images is denoted as $\mathcal{C} = \{(x_i, y_i), i = 1, 2, \ldots\}$(some matches are not correct), where x_i and y_i are matched interest points in I and I' respectively.

The similarity between line neighborhoods is measured by the line-points invariant with matched interest points in the neighborhood. For line l, $\mathcal{LPS}_l^L$ and $\mathcal{LPS}_l^R$ are evaluated separately. We take $\mathcal{LPS}_l^R$ as the example in the following steps.

Given a pair of lines $a \in \mathcal{L}$ and $b \in \mathcal{L}'$, the matched interest points in $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$ compose a set: $\{(x_i, y_i)|x_i \in \mathcal{LPS}_a^R, y_i \in \mathcal{LPS}_b^R, (x_i, y_i) \in \mathcal{C}, i = 1, 2, \ldots, N\}$, where N is the number of matched interest points. If $N < 5$, we set the similarity between the two neighborhoods as 0. Otherwise, we select one pair of points (x_i, y_i) as the ith base point pair each time, giving us N base point pairs. For each base point pair, another two pairs (x_j, y_j) and (x_k, y_k) in the remaining $N-1$ pairs can be used to calculate $FCN(KP_a^1, KP_a^2, x_i, x_j, x_k)$ and $FCN(KP_b^1, KP_b^2, y_i, y_j, y_k)$. We have C_{N-1}^2 choices, which means we have C_{N-1}^2 FCN values to represent the relationships between each base point and the line. The rth ($r = 1, 2, \ldots, C_{N-1}^2$) FCN value for the ith base point pair is denoted as $FCN_i^a(r)$ and $FCN_i^b(r)$, respectively, and the similarity between the two values is calculated by:

$$S(r) = e^{-||FCN_i^a(r) - FCN_i^b(r)||}.$$

Then we can get C_{N-1}^2 similarities for the ith base point pair, and the median value is used as the similarity of the ith base point pair to reduce the effect of mismatched points.

$$SIM(x_i, y_i) = median\{S(r)\}, r = 1, 2, \ldots, C_{N-1}^2.$$

Finally, the similarity of $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$ is denoted as the base point pair with the max similarity:

$$SIM_R(a,b) = max\{SIM(x_i, y_i)\}, i = 1, 2, \ldots, N.$$

Finally, if in image I', for all the neighborhoods of lines on the right side, $\mathcal{LPS}_b^R$ has the max similarity with $\mathcal{LPS}_a^R$, and in image I, $\mathcal{LPS}_a^R$ has the max similarity with $\mathcal{LPS}_b^R$ as well, then we take $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$ as a pair of matched line neighborhoods on the right side. The same method can be used to get matched line neighborhoods on the left side.

3.2   Matching Lines by Homography Transformation

The property of line-points invariant indicates that if the similarity between two neighborhoods of lines is very high, most of the interest points in the neighborhoods are very likely to locate on the same plane area. Thus, the homography $\mathbf{H}$ between the two neighborhoods can be calculated by matched points with Random Sample Consensus (RANSAC). Then for each line $a \in \mathcal{L}$ in image I and $b \in \mathcal{L}'$ in image I', we can map a to a' via $a' = \mathbf{H}a$, and map b to b' via $b' = \mathbf{H}^{-1}b$.

We then use two constrains to screen the potential matching lines, taking the line a and the mapped line b' in image I for example.

1. **Vertical distance constraint:** As illustrated in Fig. 6(a), d_1 and d_3 are the distances from the endpoints of line a to line b', while d_2 and d_4 are the distances from the endpoints of line b' to line a. The distance between line a and line b' is denoted as $d_v(a, b') = max(d_1, d_2, d_3, d_4)$. If $d_v(a, b') < \gamma$, then a and b' satisfy the vertical distance constraint. This constraint ensures two lines are vertically near to each other, where γ is set to 3 pixels in our experiment.
2. **Horizontal distance constraint:** As illustrated in Fig. 6(b), the distance between the midpoint of line a and line b' is denoted as d_h. If $d_h < (length(a) + length(b'))/2$, then the two lines satisfy the horizontal constraint. This constraint ensures two lines are horizontally near to each other.

If a and b' satisfy both constraints while the corresponding lines a' and b also satisfy both constraints, then line a and b are regarded as a pair of candidates. In practice, there may be one line in one image satisfying the constraints with several lines in another image, and the candidates calculated from different matched neighborhoods may also be different. In order to pick out the best-matched line pairs, a weighted voting strategy is used.

We construct a voting matrix $\mathbf{V}$ with size $n * m$, where n and m are the number of lines in images I and I', respectively. All elements are initialized by 0, which will be updated by the matched neighborhoods.

For a pair of matched neighborhoods $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$, we suppose they are well matched with similarity $SIM_R(a, b)$. As the accuracy of the candidate selection is affected by the accuracy of $\mathbf{H}$, and $\mathbf{H}$ is calculated by the matched

(a) Vertical constraint   (b) Horizontal constraint
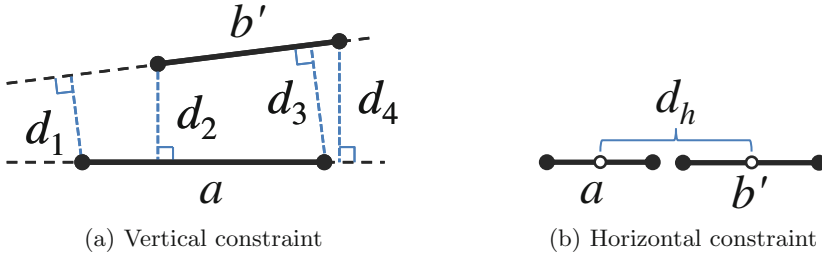
Fig. 6. Two constraints to obtain matching lines

points in $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$, we take the similarity between two matched neighborhoods into account. For example, if $a_i \in \mathcal{L}$ and $b_j \in \mathcal{L}'$ are regarded to be a pair of candidates based on $\mathbf{H}$ as calculated from $\mathcal{LPS}_a^R$ and $\mathcal{LPS}_b^R$, then $\mathbf{V}$ is updated by $\mathbf{V}_{i,j} = \mathbf{V}_{i,j} + SIM_R(a,b)$.

After all the matching neighborhoods in image I and I' are used to update $\mathbf{V}$, if $\mathbf{V}_{i,j}$ is greater than 0.9 and is the maximum between both the ith row and the jth column, then line a_i and line b_j are regarded as matched lines.

4   Experiments

To evaluate the performance of our method, another two state-of-the-art line matching methods are used for comparison: LP [4,5] and CA [9]. Both implementations are provided by their authors, and they are selected due to their good performance dealing with a wide range of image transformations. In order to follow the same protocol, we use the line detection method LSD [16] when comparing with LP, while taking the detector used in [9] when comparing with CA. In addition, the interest points used in our method and in LP are detected and matched by SIFT [11]. We test the proposed method in four conditions to verify its robustness to different changes: rotation, scaling, occlusion, and viewpoint changing. Most of the images used in our experiments are the same as [4,5]. In the viewpoint change experiment, we test the proposed method on both low texture images and high but similar texture images to illustrate the robustness of our methods to interest points changing on number and quantity.[1]

The results of the proposed method are shown in Figs. 7, 8, 9, 10, 11 and 12, where the matched lines are labeled in red with the same number. The statistical results are listed in Tables 1 and 2. The first column is the label of image pairs, and the second column is the number of lines detected from the image pair. The last two columns show the correct matched lines/total matched lines, and the correct rates of our method and the compared method. Besides the correct rates, the count of total matches also weighs the performance as more candidates of matched lines are likely to render robustness to subsequent processing such

---

[1] The source code of the proposed method is available at https://github.com/dlut-dimt/LineMatching.

(a)   (b)

Fig. 7. Results under rotation changes (Color figure online)

as stereo reconstruction and panorama stitching. Also, the more total matches there are, the more difficult it is to generate correct matches.

Rotation changes: In rotation transformation, the length of lines, angles and relative positions between lines are kept. Our result is shown in Fig. 7, and the details are shown in the first row of Tables 1 and 2. We can see that the matching precision of the three methods are all 100 % for all the methods are rotation invariant. However, the proposed method gets 4 more correct matched pairs than LP, and 53 more correct matched pairs than CA.

Scale changes: As shown in Fig. 8, the length of lines are changed in scale changes, resulting in some lines disappearing. Both LP and our method perform well with the number of correct matched lines are 121 and 131, respectively. The accuracy for both the proposed method and LP are more than 98 %, while the accuracy for CA is less than 70 % for CA depends on the position relationship between lines. The lines may disappear in large scale changes which badly affects the result.

Further, we test our method on the images with both rotation and scale changes, as shown in Fig. 9. As shown in Tables 1 and 2, our method outperforms other methods with accuracy of 93 %, while the accuracy of LP drops to 88.9 % and CA cannot get any correct matches for most lines disappear, as shown in Fig. 9. The results indicate that our method is robust in sever image transformations with the number of lines and interest points changing.

Occlusion: In the condition of occlusion, the endpoints and the length of lines are greatly changed, and methods based on such attributes may fail. We give our result in Fig. 10 and the 4*th* rows of Tables 1 and 2. The proposed method has a matching precision of 100 %, while LP and CA are 98.6 % and 95.8 %, respectively. The proposed method gets 9 and 24 more correct matches than LP and CA, respectively. In particular, the correct matched lines are twice that of CA. This result validates the robustness of our method as regards endpoints changing.
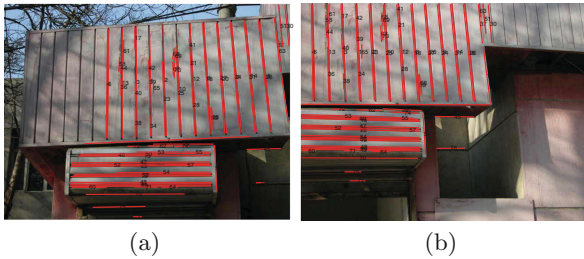
Viewpoint changes: Viewpoint changes are very common in reality. We test two groups with three images in each group. In each group, the first image is regarded as the reference image and the other two are query images, thus
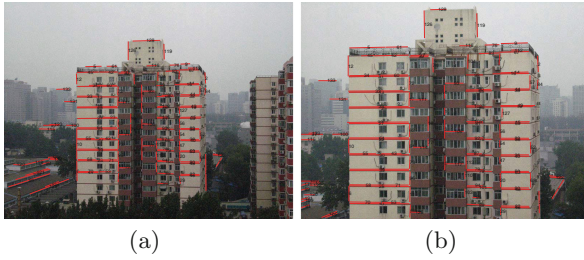
**Fig. 8.** Results under scale changes (Color figure online)
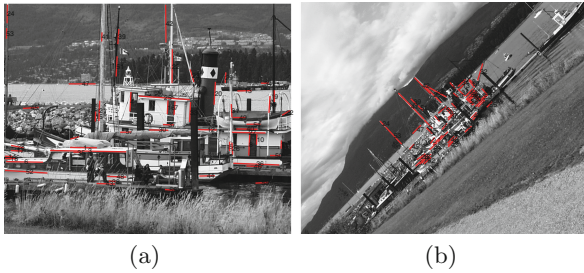
Fig. 9. Results under scale changes plus rotation (Color figure online)

Fig. 10. Results under occlusion (Color figure online)

providing us two pairs of images for each group. The first group contains three low texture images, while the second group has high texture images, and many local parts of which are similar. Both groups are very challenging.

Our results on the first group are shown in Fig. 11 and are detailed in the 5th and 6th rows of Tables 1 and 2. The proposed method and LP get all the matches correct, but CA gets only 91.1 % in the first pair and failed to get any matches again in the second pair under larger viewpoint changes. We also get 26 and 16 more correct pairs than LP, and 19 and 22 more pairs than CA. Our results on the second group are shown in Fig. 12 and are detailed in the last two rows of Tables 1 and 2. The matching precision of LP and CA are about the

(a)   (b)

(c)   (d)
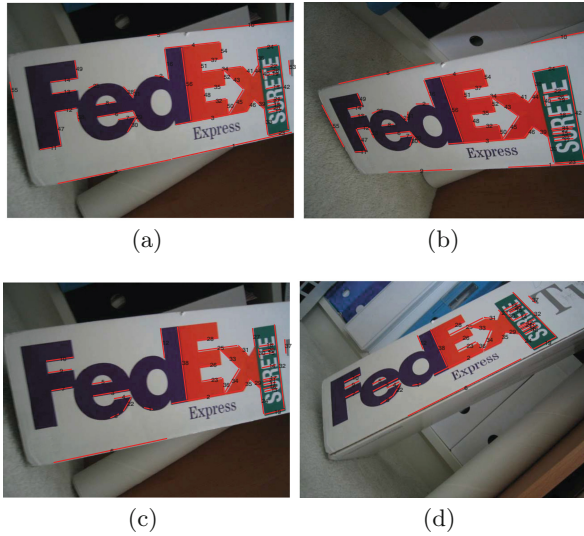
Fig. 11. Results on low texture images with viewpoint changes (Color figure online)

same as the proposed method, however, we get 25 and 28 more correct matches than LP and get 35 and 38 more correct matches than CA.

The experiments show that the proposed method is robust in projective transformation. In addition, LP [4,5] used an affine invariant with 1 line and 2 points in [4], while [5] used an additional projective invariant with 1 line and 4 points. Let N is the number of lines and M is the number of points in both images, the complexity of LP is $O(N^2M^2)$ to be invariant to affine transformations and is $O(N^2M^4)$ to be invariant to projective transformations. The proposed method is invariant to projective transformations with complexity $O(N^2M^3)$ only. Although LP can also be projective invariant, it depends on the accuracy of matched interest points. When there are few interest points in low texture

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

Fig. 12. Results on high texture images with viewpoint changes (Color figure online)

Table 1. Comparison of the proposed method and LP

Images	Extracted	Proposed method	LP
Figure 7(a)–(b)	$(272, 237)$	$(194/194, 100\%)$	$(190/190, 100\%)$
Figure 8(a)–(b)	$(315, 225)$	$(131/133, 98.5\%)$	$(121/122, 99.2\%)$
Figure 9(a)–(b)	$(540, 202)$	$(53/57, 93.0\%)$	$(40/45, 88.9\%)$
Figure 10(a)–(b)	$(210, 153)$	$(77/77, 100\%)$	$(68/69, 98.6\%)$
Figure 11(a)–(b)	$(94, 78)$	$(56/56, 100\%)$	$(30/30, 100\%)$
Figure 11(c)–(d)	$(94, 101)$	$(38/38, 100\%)$	$(22/22, 100\%)$
Figure 12(a)–(b)	$(247, 234)$	$(148/154, 96.1\%)$	$(123/126, 97.6\%)$
Figure 12(c)–(d)	$(247, 233)$	$(126/141, 89.3\%)$	$(98/112, 87.5\%)$

Table 2. Comparison of the proposed method and CA

Images	Extracted	Proposed method	CA
Figure 7(a)–(b)	$(196, 181)$	$(137/137, 100\%)$	$(84/84, 100\%)$
Figure 8(a)–(b)	$(241, 202)$	$(97/101, 96.0\%)$	$(41/59, 69.5\%)$
Figure 9(a)–(b)	$(386, 123)$	$(18/20, 90.0\%)$	$(0/0, 0\%)$
Figure 10(a)–(b)	$(169, 134)$	$(47/47, 100\%)$	$(23/24, 95.8\%)$
Figure 11(a)–(b)	$(92, 87)$	$(50/50, 100\%)$	$(41/45, 91.1\%)$
Figure 11(c)–(d)	$(92, 101)$	$(22/22, 100\%)$	$(0/0, 0\%)$
Figure 12(a)–(b)	$(217, 219)$	$(93/99, 93.9\%)$	$(58/62, 93.5\%)$
Figure 12(c)–(d)	$(217, 209)$	$(91/95, 95.8\%)$	$(53/55, 96.4\%)$

images or a large number of mismatched interest points in high texture images with similar parts, the accuracy significantly declines.

5   Conclusion

In this paper, a new line-points projective invariant is proposed, which is used to compute the similarities between line neighborhoods. An extending matching strategy is also adopted to exploit more potential matching lines. Experiment results show that the proposed method is robust against many distortions, and can achieve better performance than some existing state-of-the-art methods, especially in images with low texture and large viewpoint changes. Moreover, the proposed method is based on the planar projective invariant, so the performance for non-planar scenes may not be as good as planar scenes.

References

1. Al-Shahri, M., Yilmaz, A.: Line matching in wide-baseline stereo: a top-down approach. IEEE Trans. Image Process. **23**(9), 4199–4210 (2014)
2. Albarelli, A., Rodolà, E., Torsello, A.: Loosely distinctive features for robust surface alignment. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 519–532. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15555-0_38
3. Bergamasco, F., Albarelli, A., Torsello, A.: Pi-Tag: a fast image-space marker design based on projective invariants. Mach. Vis. Appl. **24**(6), 1295–1310 (2013)
4. Fan, B., Wu, F., Hu, Z.: Line matching leveraged by point correspondences. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 390–397. IEEE (2010)
5. Fan, B., Wu, F., Hu, Z.: Robust line matching through line-point invariants. Pattern Recogn. **45**(2), 794–805 (2012)
6. Jain, A., Kurz, C., Thormahlen, T., Seidel, H.P.: Exploiting global connectivity constraints for reconstruction of 3D line segments from images. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1586–1593. IEEE (2010)
7. Lemaire, T., Lacroix, S.: Monocular-vision based SLAM using line segments. In: 2007 IEEE International Conference on Robotics and Automation, pp. 2791–2796. IEEE (2007)
8. Li, L., Tan, C.: Recognizing planar symbols with severe perspective deformation. IEEE Trans. Pattern Anal. Mach. Intell. **32**(4), 755–762 (2010)
9. López, J., Santos, R., Fdez-Vidal, X.R., Pardo, X.M.: Two-view line matching algorithm based on context and appearance in low-textured images. Pattern Recogn. **48**(7), 2164–2184 (2015)
10. Lourakis, M.I., Halkidis, S.T., Orphanoudakis, S.C.: Matching disparate views of planar surfaces using projective invariants. Image Vis. Comput. **18**(9), 673–683 (2000)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
12. Luo, Z., Zhou, X., Gu, D.X.: From a projective invariant to some new properties of algebraic hypersurfaces. Sci. Chin. Math. **57**(11), 2273–2284 (2014)
13. Ramalingam, S., Antunes, M., Snow, D., Lee, G.H., Pillai, S.: Line-sweep: cross-ratio for wide-baseline matching and 3D reconstruction. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1238–1246, June 2015
14. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564–2571. IEEE (2011)
15. Wang, L., Neumann, U., You, S.: Wide-baseline image matching using line signatures. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1311–1318. IEEE (2009)
16. Wang, L., You, S., Neumann, U.: Supporting range and segment-based hysteresis thresholding in edge detection. In: 15th IEEE International Conference on Image Processing, ICIP 2008, pp. 609–612. IEEE (2008)
17. Wang, Z., Wu, F., Hu, Z.: MSLD: a robust descriptor for line matching. Pattern Recogn. **42**(5), 941–953 (2009)
18. Zhang, L., Koch, R.: An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency. J. Vis. Commun. Image Represent. **24**(7), 794–805 (2013)