# Smoothly Varying Projective Transformation for Line Segment Matching

Xianwei Zheng [a], Zhuang Yuan [a], Zhen Dong[a], Mingyue Dong [a], Jianya Gong[a,b] and Hanjiang Xiong[a,*]

[a]The State Key Lab. LIESMARS, Wuhan University, Wuhan, P.R. China
[b]School of Remote Sensing and Engineering, Wuhan University, Wuhan, P.R. China

ARTICLE INFO

*Keywords*:
Line segment matching
Street-level images
Motion modeling
Projective transformation

ABSTRACT

Line segment matching is of great significance to the applications that require recovering 3D structure of objects (*e.g.,* manmade objects in street-level scenarios). However, differentiating between true and false line matches is generally hard because of the lack of strong geometric constraints for line segments. Hence, some additional constraints are forced to be used sacrificing many true line matches. In this study, we propose a robust line segment matching method based on a global projective transformation modeling. Specifically, we develop a non-parametric motion regression formulation with a specially designed direct linear transformation-based cost function, which can reformulate the piecewise smoothly varying projective transformations as a global continuous model from highly noisy point matches. The resultant model can well approximate the real underlying image transformation and derive high-quality point matches. We apply the computed model and the high-quality point matches to a point correspondence-based line matching pipeline, which provide sufficient strict geometric constraints for first generating the pair-to-pair matches and then distilling the line-to-line matches. Extensive experiments conducted on two challenging line matching datasets show that the proposed method can procure considerable correct line segment matches, outperforming the comparison methods by at least 15.5% on the benchmark dataset and 16.9% on the local dataset in terms of mean F-score.

## 1. Introduction

Matching line segments between two overlapped images has a wide spectral of applications, ranging from camera calibration (Nakano, 2021), visual localization (Shipitko et al., 2020; Yu et al., 2020), stereo matching (Jellal et al., 2017; Qin et al., 2018), image stitching (Jia et al., 2021) and urban 3D reconstruction (Wei et al., 2021; Li and Yao, 2017). As line features provide richer geometric and semantic information than point features, the matching of line segments is particularly preferred in 3D reconstruction of manmade objects from street-level images. Manmade objects possess strong regular geometry, which can be easily outlined by line features. Therefore, line correspondences can complement the general shape of objects recovered from point matches with more geometric and topological structures. However, matching line segments between images is a tough task because of the limitation in the detection and description of line segments. Line detectors commonly suffer from the localization ambiguity of edge pixels, which leads to incomplete and/or false-positive detections (*e.g.,* closely distributed parallel line segments) (Xue et al., 2021). As a result, different versions of the same 3D line detected from different views can hardly result in the same representation and are thereby difficult to be matched.

Investigations have been made to line segment matching over the past decades (Li et al., 2016a), albeit not as many as those of point feature matching (Ma et al., 2021; Jin et al., 2021; Chen et al., 2018) . Matching individual line segments in descriptor space is a straightforward extension of the idea of point feature matching in the early stages. The intensity, gradient and color information (extracted from local regions) along line segments are widely exploited for constructing line descriptors that are invariant to illumination, scale, and viewpoint (Schmid and Zisserman, 1997; Bay et al., 2005; Wang et al., 2009b). However, line segments detected even by state-of-the-art methods typically suffer from the loss of connectivity and completeness. As a result, two corresponding line segments from different views can have no or few overlap, significantly reducing their descriptor similarity. The detection inaccuracy also makes establishing point-to-point correspondence for the endpoints of line segments intractable. Accordingly, the estimation

xionghanjiang@whu.edu.cn ( Hanjiang Xiong)
ORCID(s):

and application of strict geometric constraints are difficult for line segments. Some researchers remedy this problem by adding less strict geometric or topological relation priors to assist in the association of endpoints of corresponding line segments (Schmid and Zisserman, 1997; Bay et al., 2005). Nevertheless, such priors are not always available, and matching ambiguities that resulted from weakly applied constraints remain severe. To bring additionally available constraints for matching disambiguation, an alternative way is to match line segments in groups (Wang et al., 2009a; Ok et al., 2012; Kim et al., 2014; Li et al., 2016b). The grouped line segments provide rich geometric and texture relationships between each other (*e.g.,* intersection, junction, cross angle, and region similarity) that can be excavated for establishing line-to-line correspondence. However, the grouping process is usually computationally expensive and the lack of strong geometric constraints (from point correspondences) for effectively differentiating true-false line matches remains a problem.

The complexity of matching line segments in groups can be reduced by matching in pairs. Line pairs can be easily generated and possess attractive properties for applying strict geometry constraint. Intersections of coplanar line pairs are invariant to projective transformation, saying that two projectively transformed versions of the same 3D line pair have the same representation in their intersection points. Therefore, matching line segments can be transferred to the matching of intersection points. However, directly matching intersection points is difficult, as intersection points have no descriptors like feature points extracted by dedicatedly designed feature detectors (Li et al., 2016a). Moreover, intersection points are usually very small in number compared with feature points, making them inadequate to estimate the reliable geometric transformation between two views. This inspires the use of point correspondences obtained from mature feature matching to calculate the image transformations (Fan et al., 2010; Jia et al., 2018; Wang et al., 2021). Then, the outputs of feature matching (*e.g.,* point matches and estimated transformations) are incorporated and propagated to the various steps of line segment matching, which can eliminate many false line matches. However, many existing point correspondence-based approaches only perform well when enough high-quality point matches exist around line segments, which are difficult to be satisfied in some realistic scenarios. For example, in wide baseline scenarios (particularly the street-level scenarios), existing feature matching methods usually yield point matching results with a high outlier rate. Moreover, in such scenarios, object surfaces undergo a more general transformation combining affine with projectivity. Existing feature matching methods mainly focus on easily modeled image motion aspects (*e.g.,* local similarity or affine), which are inadequate to describe the transformation for a line segment that crosses a large image region. This problem is more severe in textureless regions where no insufficient point matches assist in the modeling of local motions.

According to the above observation, we propose to compute a global Smoothly varying projective transformation model for robust Line sEgment Matching (SLEM). The key idea of SLEM is to recover a global projective transformation model that can well approximate the real underlying image transformation, which also serve as a reliable filter to derive high-quality point matches. The global projective model and the point matches are then propagated to the line segment matching pipeline for finding the correct line matches as many as possible. Specifically, a non-parametric motion regression formulation with a specially designed direct linear transformation (DLT)-based cost function is developed to model the piecewise smoothly varying projective transformation from noisy point matches. During the regression, the motion coherence is enforced and a global smoothest projective model that is consistent with the observed data is obtained via a global minimization. The computed model can filter the input point matches for true matches with very high quality. We integrate our regression technique into a point correspondence-based line matching pipeline. The high-quality point matches are utilized to compute the epipolar geometry for finding the candidate pair-to-pair line correspondences. The smoothly varying projective transformation model is used to infer the projective transformations at the intersections of line pairs, which serve as the strict geometric constraints for filtering the false intersection correspondences (i.e., the false line pair matches). The projective transformations at the intersections of line pairs are utilized to precisely map their associated line pairs from one view to another. Then mapping similarities can be computed between the line segments from corresponding pairs. Based on the measured similarities, an E-distance voting and a crosscheck are applied to obtain the final line-to-line matching results.

The proposed method has some distinctive practical advantages: (1) By enforcing motion coherence in a non-parametric regression function, we can compute a global smoothest model from highly noisy point matches. The estimated model creates a powerful separability constraint for differentiating true and false point matches. (2) We developed a DLT-based cost function that enables the modeling of smoothly varying projective transformation in a general motion regression formulation. The projective transformation can better generalize image motions than other easily modeled motion aspects (*e.g.,* similarity or affine) across large image regions. Thus, this transformation is more adaptable to line segments that have varying lengths. (3) The feature matching and the line segment matching

are loosely and globally coupled, relaxing the restrictions on the density and distribution of point correspondences. Our globally modeled projective transformation can be computed from a subset of available data at specific positions and easily extrapolated to new positions (*e.g.,* intersections of line pairs), which relieves the high requirement on the number of point matches around line segments as existing approaches (particularly those locally assembled and/or affine space-based line matching approaches).

The rest of this paper is organized as follows. Section 2 presents a brief review of related works, and Section 3 describes the details of the proposed SLEM. Section 4 and Section 5 provide the experimental results and analysis. Section 6 draws some conclusions.

## 2. Related Work

In this section, we briefly review some previous works that are relevant to this study. These works can mainly be divided into three categories based on the ways that line segments are matched.

### 2.1. Individual-based line matching

This class of methods follows the idea of point feature matching. They associate each line segment with a descriptor that is invariant to image transformations, and each line segment is then matched to its nearest neighbor in the descriptor space. Targeting at the different types of image transformation, several transformation-invariant descriptors have been proposed to exploit the photometric information of neighboring pixels (*e.g.,* pixels in a rectangle centered at the line segment). Bay et al. (2005) counted the histograms of the neighboring color profiles to construct initial line matches. Wang et al. (2009b) proposed a SIFT-like descriptor called mean–standard deviation line descriptor (MSLD) to enhance the descriptor robustness against image transformations and noises. This method counts the statistical values of gradient of pixels in the neighborhood of a line segment, which is invariant to the localization of endpoints. This idea of boosting descriptor robustness has motivated many subsequent works. Among them, Zhang and Koch (2013) and Verhagen et al. (2014) added a scale variance to descriptors. Zhang and Koch (2013) detected and described lines in multi-scale pyramids of image space, and Verhagen et al. (2014) utilized the line segments detected on the lowest scale. After matching by descriptor, geometric relations between the images are usually used to purify the matching results and retrieve the unmatched line segments. Bay et al. (2005) proposed a topological filter to remove mismatches and retrieve unmatched line segments with a consistent topological structure. Schmid and Zisserman (1997) used the epipolar geometry that are obtained from camera projection matrices or estimated from point correspondence to compute the matching score for putative line correspondences. The main disadvantages of these methods are a lack of strong geometric constraint for matching disambiguation. Under extreme scale and viewpoint changes, the descriptor similarity of corresponding line segments could be significantly reduced, leading to a large proportion of ambiguous matches that are hard to differentiate without strict geometric constraints.

### 2.2. Group-based line matching

In this category of methods, line segments are clustered into groups or pairs to form integrated features, which provide additional geometric or topological constraints for matching disambiguation. Wang et al. (2009a) adopted a line segment grouping strategy based on spatial and gradient saliency and computed the geometric attributes and inner topological relationship for each line segment group. As grouping line segments requires a certain number of neighboring line segments, many approaches focus on constructing line pairs instead. These methods found line pairs of interest from the extracted line segments in each image. Typically, numerous line pairs are first freely generated to guarantee a high recall rate and various geometric constraints are then applied to remove the redundant ones. Ok et al. (2010) investigated pairs of lines that belong to the same object by assessing their proximity, cross angle and similarity of flanking regions. Then, the correspondence of line pairs is established based on their geometric attributes, for example, topological relationship and radiometric information of their neighborhoods . Li et al. (2016b) took the advantage of the property of coplanar line pairs whose intersection points are invariant under projective transformation. By searching for coplanar line pairs, this method converts the line pair matching into a point matching problem, which can be accomplished by a scale-invariant descriptor called Line-Junction-Line (LJL). Kim and Lee (2012) introduced line intersection context features (LICFs) to capture geometrically representative structures of line pairs. In addition, the intersection points can be used to estimate image transformations, *e.g.,* global fundamental matrix (Li et al., 2016b) or local homography (Kim and Lee, 2012). Matching based on intersection points relieve the computational burden of line segment descriptor construction and eschew the issue of inaccurate localization of endpoints.

### 2.3. Point correspondence-based line matching

The point correspondence-based line matching methods leverage the feature matching results as reliable priors to guide the matching of line segments. Different from the group-based methods that use endpoints and intersections to estimate image transformation, these approaches are based on feature points which are considerable and have higher localization accuracy than the endpoints of line segments. Point matches are typically used to construct a geometric invariant or geometric constraint that guides the matching of line segments. The geometric invariant for each line segment is usually defined as the ratio of its distance to the neighboring coplanar interest points, which can be categorized according to the number of the selected interesting points. This includes the one-point affine-invariant (Fan et al., 2010), the four-point projective-invariant (Fan et al., 2012) and the multi-point characteristic number (Jia et al., 2018). The geometric invariant serves as line segment descriptors, which overcome the difficulties of directly constructing geometric relation between line segments and are robust to image transformations including rotation, scale and viewpoint changes. In comparison, a geometric constraint is usually coupled with line pairs. When there are sufficient point matches with high precision, line segments can be directly matched by an incorporation of strict homography constraint, epipolar geometry and other topological geometric relation. However, in most cases, these geometric constraints are used in the preprocessing or postprocessing procedures, *e.g.,* to construct putative line pair correspondences for further descriptor-based matching (Wang et al., 2021). Point correspondence-based methods are inevitably restricted by the point matching performance. For instance, in the wide baseline street-level scenarios, many concurrent nuisance factors (*e.g.,* low texture, repetitive pattern) undermine the feature matching. In such scenarios, existing feature matching methods typically suffer from a high outlier rate of the obtained point matches. Jia et al. (2018) proposed two-line similarity measure to improve the tolerance against the point mismatches, but a low recall rate of obtained line matches is produced. Existing feature matching methods mainly focus on the easily modeled image motion aspects (*e.g.,* similarity and affine), which are inadequate to represent the homography induced by two views of a planar surface spanning a large image region. Hence, in the regions where line segments have no sufficient point features around them, the geometric constraints are inapplicable.

In this study, we address the problems of existing point correspondence-based methods by computing a smoothly varying projective transformation model to assist in line segment matching. The model can serve as a reliable filter to remove false matches for high-quality point matches. The point matching results and the computed projective model can provide strong geometric constraints to support robust line segment matching, even in image regions with insufficient point matches.

## 3. Methodology

The workflow of the proposed SLEM consists of two major working branches: a feature matching branch and a line segment matching branch, as illustrated in Figure 1.

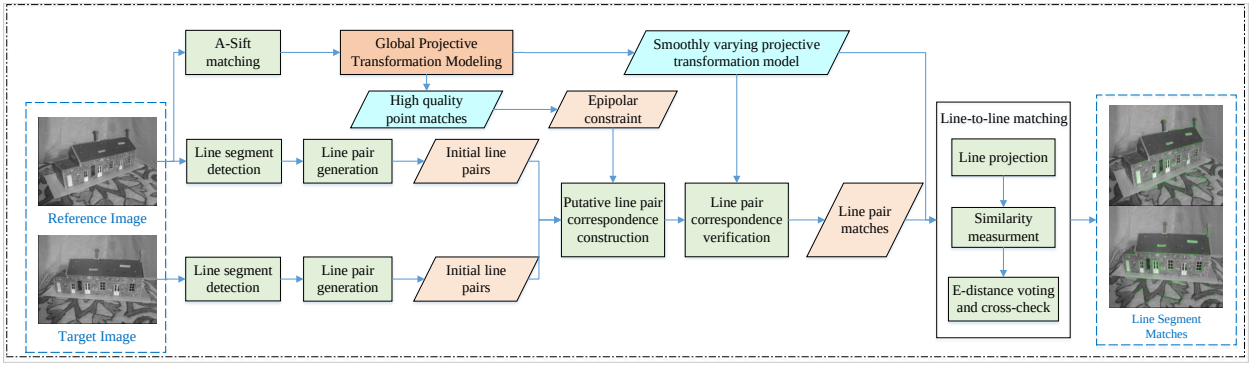



**Figure 1:** Workflow of the proposed SLEM for line segment matching.

In Figure 1, given a pair of input reference and target images, the feature matching branch first constructs a set of putative A-Sift matches (denoted as $\mathcal{M}$) via nearest neighbor matching with a preemptive ratio-test filtering (Yu and Morel, 2011). A global projective transformation modeling is then performed on the putative A-Sift match set $\mathcal{M}$, based on our coherence-based non-parametric motion regression. The outputs of this branch are a global smoothly varying projective transformation model and high-quality point matches. Meanwhile, the line segment matching branch

detects line segments from each input image and groups them into pairs. The high-quality point matches are used to compute the epipolar geometry for establishing the putative pair-to-pair line correspondences. The computed global projective transformation model is then extrapolated to the positions of corresponding intersections, which serve as a strict geometric constraint for verifying the pair-to-pair line correspondence hypotheses. The verified line pair matches are finally fed into the line-to-line matching pipeline. These line pairs are mapped from one image to another by the projective transformations recovered from their intersections. After applying the operations of similarity measurement, E-distance voting and crosscheck on the line segments and their mapped counterparts, we obtain the final line-to-line matches. In the following, we elaborate on the details of the proposed SLEM.

## 3.1. Smoothly varying projective transformation modeling

In this section, we first provide a general formulation for coherence-based image motion regression. Then, we apply our regression formulation to the global modeling of the smoothly varying projective transformation.

### *3.1.1. Generalized image motion regression*

**A. Motion description**

Let $\mathbf{X}^r = (x^r, y^r) \in \mathrm{RI}$, $\mathbf{X}^t = (x^t, y^t) \in \mathrm{TI}$ be image coordinates of two corresponding feature points of a true match, where RI and TI refer to a pair of reference and target images respectively. We can find a $3 \times 3$ transformation matrix $\mathbf{M}$ satisfying the following condition:

$$(x^t, y^t, 1)^T = \mathbf{M}(x^r, x^r, 1)^T, \tag{1}$$

The matrix $\mathbf{M}$ mathematically describes the underlying image motion induced by the two corresponding feature points, which also reveals the geometric relationship between corresponding feature points from a pair of reference and target images. In practice, the correct point correspondences are initially unknown. The geometric property of matrix (motion) $\mathbf{M}$ provides a strong constraint to verify correspondence hypothesis, making $\mathbf{M}$ an effective filter to reject the false point matches. Therefore, point feature matching can be converted into a problem of estimating the image motion from sparsely scattered putative point matches. In this study, we realize this estimation in a regression formulation by exploiting the motion coherence of correct point matches.

**B. Motion coherence**

Given a scene plane $\boldsymbol{\pi} = (\mathbf{v}^T, 1)^T$, where $\mathbf{v}^T = (a, b, c)$ is a 3D vector, any 3D point with its coordinate $\mathbf{X} = (x, y, z, 1)$ lies on the plane satisfies the condition $\pi^T\mathbf{X} = 0$. It holds that the projections of a 3D point on a pair of images TI and RI are related by a homography $\mathbf{H}$ according to the plane it lies on:

$$\mathbf{H} = \mathbf{A} - \mathbf{e}'\mathbf{v}^T, \tag{2}$$

where $\mathbf{e}'$ is the epipole from the view TI and $\mathbf{A}$ is the first three columns of the second camera matrix (Andrew, 2001). $[\mathbf{e}']_\bullet\mathbf{A} = \mathbf{F}$ is a decomposition of the fundamental matrix. $\mathbf{H}$ can be regarded as a specialization of the motion $\mathbf{M}$. Theoretically, $\mathbf{H}$ is a projective transformation that has eight degrees of freedom (DoF), but the choice of $\mathbf{H}$ is usually task-specific. To satisfy different application demands, a simplification version of $\mathbf{H}$ usually takes the form of affine (with six DoF) or similarity transformation (with four DoF). Despite variations in motion aspects, they all have an attractive property that is motion coherence, saying neighboring pixels share similar motions. The reason is that the spatial continuity of neighboring physical points that lie on the same real-world 3D plane is maintained in their projections on different views of the plane, which causes the projections of neighboring scene points to share coherent motions (*e.g.,* affine, or projective transformation). Specifically, given two points on the same physical plane, the coordinates of their projections on RI, i.e. $\mathbf{X}_1^r$, $\mathbf{X}_2^r$, and the coordinates of the corresponding points on TI, i.e. $\mathbf{X}_1^t$, $\mathbf{X}_2^t$, satisfy the condition $(\mathbf{X}_1^t, 1)^T = \mathbf{H}_1(\mathbf{X}_1^r, 1)^T$ and $(\mathbf{X}_2^t, 1)^T = \mathbf{H}_2(\mathbf{X}_2^r, 1)^T$, if $|\mathbf{X}_1^r - \mathbf{X}_2^r| < \delta$, then the motion of the two matches should be mutually substitutable (similar), *i.e.,* there exists a small $\varepsilon(\delta)$ such that $|\mathbf{H}_1(\mathbf{X}_1^r, 1)^T - \mathbf{H}_1(\mathbf{X}_2^r, 1)^T| < \varepsilon$, $|\mathbf{H}_2(\mathbf{X}_1^r, 1)^T - \mathbf{H}_2(\mathbf{X}_2^r, 1)^T| < \varepsilon$. Accordingly, we can eliminate the wrong matches whose motion varies greatly from their neighboring matches.

**C. Non-parametric motion regression**

In real-world scenarios, image transformations are complicated and could vary according to different data, which usually causes motion to be locally consistent but piecewise smooth across a large image region (Black and Anandan, 1996; Ye et al., 2003). A predefined parametric model is difficult to finely capture the underlying image motions. To

address this problem, a non-linear regression formulation (Lin et al., 2013) is adopted to seek a global smooth non-parametric model that can well generalize image motions. The formulation models a coherence cost for every possible motion and the coherence constraint is enforced by a global minimization.

Specifically, the problem is formulated as fitting a smooth function $f : \mathbf{p} \rightarrow q$, where $\mathbf{p}$ (vectors) and $q$ (scalars) are the domain and co-domain of $f(.)$. Here, $f(.)$ can be regarded as a functional representation of $\mathbf{M}$. To simplify computation, we choose a subspace of $f(.)$: span$\{f_1(.), f_2(.), \ldots, f_K(.)\}$, where $f_k$ is a specific form of smooth function, and use the projections of $f(.)$ on this subspace to formulate $f(.)$. Therefore, $f(.)$ can be formulated as a linear combination of $\{f_1(.), f_2(.), \ldots, f_K(.)\}$:

$$q = f(\mathbf{p}) = \sum_{k=1}^{K} a_k(\mathbf{p}) f_k(\mathbf{p}), \tag{3}$$

where the component $f_k(.)$ can be considered as a decomposition of the total motion $f(.)$ on the $k$th dimension and $a_k(\mathbf{p})$ defines the weight for each component. The number of $K$ usually depends on the selected motion aspects (*e.g.,* affine or projective transformation).

Considering the existence of observation error, we let $\{\mathbf{p}_j, \hat{q}_j | j = 1, 2, \ldots, N\}$ be a set of $N$ noisy observed data points (we can treat putative point matches as the noisy observed data points), where $\{\hat{q}_j\}$ are scalar values at locations $\{\mathbf{p}_j\}$. Thus, Eq. (3) can be reformulated as follows:

$$\hat{q}_j = f(\mathbf{p}_j) + n_j = \sum_{k=1}^{K} a_k(\mathbf{p}_j) f_k(\mathbf{p}_j) + n_j, \tag{4}$$

with $n_j$ denoting noise. $f_k(.)$ consists of a constant and a series term

$$f_k(\mathbf{p}) = s_k + \varphi_k(\mathbf{p}), \tag{5}$$

where $s_k$ controls the overall offset and $\varphi_k(.)$ enforces the smoothness of $f_k(.)$ while at the same time best approximate the complex transformation model induced by complicated scenarios.

For regressing the best fit function $f(.)$, we can minimize an empirical loss $E_1$ as follows

$$E_1 = \sum_{j=1}^{N} C(\hat{q}_j - f(\mathbf{p}_j)) = \sum_{j=1}^{N} C\left(\hat{q}_j - \sum_{k=1}^{K} a_k(\mathbf{p}_j) f_k(\mathbf{p}_j)\right), \tag{6}$$

where $C(.)$ can be a Huber function with a threshold $\epsilon$ (Huber, 1992) defined as

$$C(z) = Huber(z) = \begin{cases} z^2 & z \leq \varepsilon \\ 2\varepsilon|z| - \varepsilon^2 & z > \varepsilon. \end{cases} \tag{7}$$

Through the minimization of $E_1$, we can obtain a global smoothest function $f(.)$ that is consistent with the observed data. However, the minimization generally tends to increase the weight of the high-frequency term to overfit the observed data without constraint. To prevent overfitting, a regularization term $E_2$ should be introduced to punish the high-frequency term, thereby guaranteeing the global smoothness with motion coherence constraint (Yuille and Grzywacz, 1988; Myronenko et al., 2007). $E_2$ takes the form as follows:

$$E_2 = \lambda \sum_{k=1}^{K} \int_{R^D} \frac{|\bar{\varphi}_k(\omega)|^2}{\bar{g}(\omega)} d\omega, \tag{8}$$

where $\lambda$ denotes the weight of the penalty term. $\bar{\varphi}_k(.)$ and $\bar{g}(.)$ are the Fourier transform of $\varphi_k(.)$ and Gaussian function, respectively. As frequency $\omega$ becomes larger, the denominator $\bar{g}(\omega)$ decreases to penalize the high-frequency terms of $\bar{\varphi}_k(.)$, thereby globally increasing the smoothness of $\varphi_k(.)$.

With the empirical loss $E_1$ and the regularization loss $E_2$, the total energy function $E$ can be expressed as a summation of $E_1$ and $E_2$ by

$$E = E_1 + E_2 = \sum_{j=1}^{N} C(\hat{q}_j - \sum_{k=1}^{K} a_k(\mathbf{p}_j) f_k(\mathbf{p}_j)) + \lambda \sum_{k=1}^{K} \int_{R^D} \frac{|\bar{\varphi}_k(\omega)|^2}{\bar{g}(\omega)} d\omega. \tag{9}$$

According to the Euler-Lagrange equations, we have

$$\frac{\partial E}{\partial \bar{\varphi}_k(z)} = 0, \forall z \in \mathbb{R}^D, k = 1, 2 \dots K, \tag{10}$$

From Eq. (10), we can get $\bar{\varphi}_k(z)$ as follows:

$$\bar{\varphi}_k(z) = \bar{g}(-z) \sum_{j=1}^{N} \mathbf{w}_k(j) e^{-2\pi\imath <\mathbf{p}_j, \mathbf{z}>}. \tag{11}$$

By substituting Eq. (11) to Eq. (8), the regularization term $E_2$ can be rewritten in terms of $\mathbf{w}_k$ as

$$E_2 = \mathbf{w}_k^T G \mathbf{w}_k, \tag{12}$$

and meanwhile, $\varphi_k$ can be expressed according to the inverse Fourier transform of Eq. (11) as follows:

$$\varphi_k(\mathbf{p}) = \sum_{j=1}^{N} \mathbf{w}_k(j) g(\mathbf{p}, \mathbf{p}_j) = \sum_{j=1}^{N} \mathbf{w}_k(j) e^{-\frac{||\mathbf{p}-\mathbf{p}_j||}{\gamma^2}}, k = 1, 2, \dots, K \tag{13}$$

where $\mathbf{w}_k$ is an unknown $N$-dimensional vector to be estimated. $g(\mathbf{p}_i, \mathbf{p}_j)$ is a Gaussian radial basis function and $G$ in Eq. (12) is an $N \times N$ Gram matrix, *i.e.,* $G(i, j) = g(\mathbf{p}_i, \mathbf{p}_j)$. The form of $\varphi_k(.)$ indicates that the value of smooth function $f_k(\mathbf{p})$ at location $\mathbf{p}$ is relevant to the values at other locations. This form allows to globally enforce the smoothness constraint with motion coherence. Substituting Eq. (13) into Eq. (5) yields

$$f_k(\mathbf{p}) = s_k + \sum_{j=1}^{N} \mathbf{w}_k(j) g(\mathbf{p}, \mathbf{p}_j) \tag{14}$$

In practice, many points in the observed data set $\{\mathbf{p}_j\}$ are adjacent. To accelerate the computation efficiency of the minimization process, the regression function $f_k(\mathbf{p})$ can be approximated by

$$f_k(\mathbf{p}) \approx \tilde{f}_k(\mathbf{p}) = s_k + \sum_{j=1}^{M} \tilde{\mathbf{w}}_k(j) g(\mathbf{p}, \tilde{\mathbf{p}}_j) \tag{15}$$

where $\tilde{\mathbf{p}}_j$ are $M$ representative points ($M \ll N$) that can well approximate the distribution of the original observed data points $\mathbf{p}_j$. Accordingly, $\tilde{\mathbf{w}}_k$ is a $M$-dimensional vector and $s_k$ is the corresponding offset.

By substituting Eq. (12) and Eq. (15) into Eq. (9), we can finally obtain the following objective energy function

$$\begin{aligned} \arg\min E_1 + E_2 &= \arg \min_{\{f_k(\mathbf{p}_j)\}} \sum_{j=1}^{N} C\left(\hat{q}_j - \sum_{k=1}^{K} a_k(\mathbf{p}_j) f_k(\mathbf{p}_j)\right) + \lambda \sum_{k=1}^{K} \int_{R^D} \frac{|\bar{\varphi}_k(\omega)|^2}{\bar{g}(\omega)} d\omega \\ &= \arg \min_{\{\tilde{\mathbf{w}}_k, s_k\}} \sum_{j=1}^{N} C\left(\hat{q}_j - \sum_{k=1}^{K} a_k(\mathbf{p}_j)\left(s_k + \sum_{i=1}^{M} \tilde{\mathbf{w}}_k(i) g(\mathbf{p}_j, \tilde{\mathbf{p}}_i)\right)\right) + \lambda \sum_{k=1}^{K} \tilde{\mathbf{w}}_k^T G \tilde{\mathbf{w}}_k^T. \end{aligned} \tag{16}$$

In Eq. (16), as the Huber function $C(.)$ is convex and Gram matrix $G$ is positive definite, the overall energy function is also convex. Thus, the global optimal solution can be achieved by a gradient descent minimization.

### *3.1.2. Regression-based global projective transformation modeling*

Typically, affine transformation can be used to locally approximate projective transformation to simplify computation in the task of point matching (Lin et al., 2017). However, as line segments generally cross a large image region, the locality assumption can be broken and affine transformation is no longer a good approximation. This error accumulates as the length of line segment increases. In this study, we choose projective transformation as the functional form of $\mathbf{M}$, which is adaptable to the varying length of line segments and can well generalize the motion of large image regions

(from point to line segments). A projective transformation is represented by a $3 \times 3$ matrix with eight DOF as follows:

$$\mathbf{H} = \begin{pmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{v}^T & 1 \end{pmatrix}. \tag{17}$$

$\mathbf{H}$ is a composition of a non-singular linear transformation $\mathbf{A}$, translation $\mathbf{t}$ and $\mathbf{v}$ which models the vanishing point. In the following, we elaborate on how to globally model the piecewise smoothly varying projective transformation with the generalized motion regression formulation presented in Section 3.1.1. We start by a rough filtering of the putative A-sift match set $\mathcal{M}$ with a coherence probabilistic model to make the projective transformation regression feasible and efficient. Then, we propose a direct linear transformation (DLT)-based cost function to allow the optimization of energy function in Eq. (16) to be applied for projective transformation.

**A. Coherence probabilistic model**

Given the putative point match set $\mathcal{M}$, our goal is to fit a global smooth projective transformation model to $\mathcal{M}$ with the regression formulation presented in Section 3.1.1. This requires computing the motion for every match in $\mathcal{M}$. Given the aperture effect, measuring motion from a single match is unreliable (Shimojo et al., 1989). A coherent motion (estimated for a match) requires a certain number of consistent true matches to support its measurement. However, the set $\mathcal{M}$ usually contains a majority of outliers that have incoherent motions. These outliers bring heavy computation burden to the regression and confuse the estimation of coherent motions. To address this problem, we select a restricted set of well-distributed and consistent matches from $\mathcal{M}$ for subsequent fitting. This goal is accomplished through a rough filtering of unexpected matches with a coherence probabilistic model.

We define a set of $N_t$ noisy observed data points representing the $N_t$ putative matches as follows

$$\mathbf{P} = \{\mathbf{p}_j = (\mathbf{X}_j^r, \mathbf{O}_j, \mathbf{A}_j), \hat{q}_j = 1 | j = 1, 2, \dots, N_t\}, \tag{18}$$

where $\mathbf{O}_j = \mathbf{X}_j^t - \mathbf{X}_j^r$ is the displacement between two corresponding points of a match (which is a simple motion aspect that can be directly observed) and ($\mathbf{X}_j^t$ and $\mathbf{X}_j^r$) are image coordinates of corresponding points. $\mathbf{A}_j$ is a 4x1 vector representing the quasi-affine parameters of the matched feature derived from A-Sift (an approximation of affine motion). Each point match hypothesizes a likelihood '1' at location $\mathbf{p}_j$, which means that a point match is initially hypothesized to have coherent motion with other point matches. Then, we verify this coherence hypothesis through a regression-based likelihood function. Note that as the projective motion (transformation) for each match is to be estimated, it is able to roughly verify the motion coherence of point matches in the aspects of displacement ($\mathbf{O}_j$) and quasi-affine motion ($\mathbf{A}_j$) (Lin et al., 2014). We choose $K = 1$, and $a_k(\mathbf{p}_j) = 1$ in Eq. (4), and set the offset term $s_k = 0$ and $M = N_t$ in Eq. (15). We can get a simplified regression function as

$$\tilde{f}_k(\mathbf{p}) = \sum_{j=1}^{N_t} \tilde{\mathbf{w}}_k(j) g(\mathbf{p}, \tilde{\mathbf{p}}_j). \tag{19}$$

Substitute the regression function Eq. (19) into the energy function Eq. (16), we have the following likelihood function:

$$\begin{aligned} \arg\min E_1 + E_2 &= \arg\min_{\tilde{\mathbf{w}}} \sum_{j=1}^{N_t} C\left(1 - f(\mathbf{p}_j)\right) + \lambda \tilde{\mathbf{w}}^T G \tilde{\mathbf{w}} \\ &= \arg\min_{\tilde{\mathbf{w}}} \sum_{j=1}^{N_t} C\left(1 - \sum_{i=1}^{N_t} \tilde{\mathbf{w}}(i) g(\mathbf{p}_j, \tilde{\mathbf{p}}_i)\right) + \lambda \tilde{\mathbf{w}}^T G \tilde{\mathbf{w}} \end{aligned} \tag{20}$$

In Eq. (20), the penalization term $\tilde{\mathbf{w}}^T G \tilde{\mathbf{w}}$ encourages $\tilde{\mathbf{w}}$ to be zero unless forced upward by the cost $E_1$ to fit the observed data. This penalizes all motions except for the locally clustered and globally consistent motions, as the cumulative empirical cost $E_1$ for clusters of locally consistent matches is relatively small, while motions that have broad supports from coherent matches incur low smooth penalty. In $E_1$, $g(\mathbf{p}_j, \tilde{\mathbf{p}}_i)$ (i.e. elements of Gram matrix) encodes the distances between the data points (see Eq. (13)). The distances between consistent data points that have coherent motions are small. By contrast, the distances between scattered and noisy data points that have incoherent

motions are arbitrarily large. Therefore, through the minimization of Eq. (20), we can obtain a coherence probabilistic model that computes a likelihood for each hypothetical correspondence to check its consistency with other data points. By performing a thresholding step on the computed likelihoods as follows

$$\text{accept}(\mathbf{p}_j) = \text{true} \quad \text{if } (1 - f(\mathbf{p}_j)) < \epsilon_{likelihood}, \tag{21}$$

the probabilistic model filters out a majority of the incoherent matches in $\mathcal{M}$ and provide a refined point match set $\mathcal{M}_r$ for the following projective transformation modeling.

**B. DLT-based cost function**

After obtaining the refined match set $\mathcal{M}_r$ by the coherence probabilistic model, we now compute the global projective transformation model which is later used to assist in robust line segment matching. The projective transformation can be represented by a homography functional matrix as

$$\mathbf{H}(\mathbf{p}) = \begin{pmatrix} f_1(\mathbf{p}) & f_2(\mathbf{p}) & f_3(\mathbf{p}) \\ f_4(\mathbf{p}) & f_5(\mathbf{p}) & f_6(\mathbf{p}) \\ f_7(\mathbf{p}) & f_8(\mathbf{p}) & 1 \end{pmatrix}, \tag{22}$$

where elements $f_k(\mathbf{p}), k = 1, .., 8$ are non-parametric functions of $\mathbf{p}$. Before regressing the global projective transformation model, we need to take a special concern on the definition of $\mathbf{p}_j$. At the boundaries of the 3D surface, the motions induced by two views can vary abruptly. To handle this problem, we can define $\mathbf{p}_j$ the same way as Eq. (18), which enables the computation of a global continuous model from the spatially discontinuous motions in a high dimensional domain. However, to assist in line segment matching, the estimated global projective model should be able to be applied on the intersection points of line segments. Considering that the intersection points have no information representing quasi-affine parameters as the A-sift feature points, we remove $\mathbf{A}_j$ from the definition of $\mathbf{p}_j$ in Eq. (18). The goal here is to find the smooth transform $\mathbf{H}(\mathbf{p}_j)$ that maps the coordinates of point features from RI to TI with a minimum total mapping error. Accordingly, the definition of $\hat{q}_j$ is also changed. We define the $N_r$ observed data points which represent the $N_r$ refined point matches for regression as follows:

$$\mathbf{P} = \{\mathbf{p}_j = (\mathbf{X}_j^r, \mathbf{O}_j), \hat{\mathbf{q}}_\mathbf{j} = \mathbf{X}_j^t | j = 1, 2, \dots, N_r\}, \tag{23}$$

Given a feature point in RI with its coordinates as $\mathbf{X}_j^r = (x_j^r, y_j^r)$, its mapped coordinates $\hat{\mathbf{X}}_j^t = (\hat{x}_j^t, \hat{y}_j^t)$ in TI can be computed by

$$\hat{\mathbf{X}}_j^t = \mathbf{H}(\mathbf{p}_j)\mathbf{X}_j^r, \tag{24}$$

By substituting Eq. (22) into Eq. (24), we obtain

$$\begin{aligned} \hat{x}_j^t &= \frac{f_1(\mathbf{p}_j)x_j^r + f_2(\mathbf{p}_j)y_j^r + f_3(\mathbf{p}_j)}{f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1} \\ \hat{y}_j^t &= \frac{f_4(\mathbf{p}_j)x_j^r + f_5(\mathbf{p}_j)y_j^r + f_6(\mathbf{p}_j)}{f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1} \end{aligned} \tag{25}$$

According to the energy function of Eq. (16) for generalized motion modeling, the cost function for gross mapping error can be formulated as follows:

$$\begin{aligned} &\arg \min_{\{f_k(\mathbf{p}_j)\}} \sum_{j=1}^{N_r} C(x_j^t - \hat{x}_j^t) + \sum_{j=1}^{N_r} C(y_j^t - \hat{y}_j^t) + \lambda \sum_{k=1}^{8} \int_{R^D} \frac{|\bar{\varphi}_k(\omega)|^2}{\bar{g}(\omega)} d\omega \\ = &\arg \min_{\{f_k(\mathbf{p}_j)\}} \sum_{j=1}^{N_r} C(x_j^t - \frac{f_1(\mathbf{p}_j)x_j^r + f_2(\mathbf{p}_j)y_j^r + f_3(\mathbf{p}_j)}{f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1}) + \lambda \sum_{k=1}^{8} \int_{R^D} \frac{|\bar{\varphi}_k(\omega)|^2}{\bar{g}(\omega)} d\omega + \\ &\sum_{j=1}^{N_r} C(y_j^t - \frac{f_4(\mathbf{p}_j)x_j^r + f_5(\mathbf{p}_j)y_j^r + f_6(\mathbf{p}_j)}{f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1}). \end{aligned} \tag{26}$$

As the denominator contains unknown parameters $f_7(.)$ and $f_8(.)$, the convexity of Eq. (26) is unknown and direct optimization is intractable. To address this problem, we adopt the DLT algorithm (Andrew, 2001) to reformulate our cost function. Let $(\mathbf{X}_j^r, \mathbf{X}_j^t)$ be the coordinates of two corresponding points of a correct match, according to Andrew (2001), the geometric relationship between these two points can be described in the following cross product form:

$$\begin{aligned}
&① \ (\mathbf{X}_j^t, 1)^T \times \mathbf{H}(\mathbf{p}_j)(\mathbf{X}_j^r, 1)^T = 0 \iff (x_j^t, y_j^t, 1)^T \times \mathbf{H}(\mathbf{p}_j)(x_j^r, y_j^r, 1)^T = 0 \\
\iff \ &② \ (x_j^t, y_j^t, 1)^T \times \begin{pmatrix} f_1(\mathbf{p}_j)x_j^r + f_2(\mathbf{p}_j)y_j^r + f_3(\mathbf{p}_j) \\ f_4(\mathbf{p}_j)x_j^r + f_5(\mathbf{p}_j)y_j^r + f_6(\mathbf{p}_j) \\ f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1 \end{pmatrix} = (x_j^t, y_j^t, 1)^T \times \begin{pmatrix} f_x(\mathbf{p}_j) \\ f_y(\mathbf{p}_j) \\ f_z(\mathbf{p}_j) \end{pmatrix} = 0 \\
\iff \ &③ \ (x_j^t, y_j^t, 1)^T \times \mathbf{H}(\mathbf{p}_j)(x_j^r, y_j^r, 1)^T = \begin{pmatrix} f_z(\mathbf{p}_j)y_j^t - f_y(\mathbf{p}_j) \\ -f_z(\mathbf{p}_j)x_j^t + f_x(\mathbf{p}_j) \\ f_y(\mathbf{p}_j)x_j^t - f_x(\mathbf{p}_j)y_j^t \end{pmatrix} = 0
\end{aligned} \tag{27}$$

As the correctness of a given point match is unknown, the Euler distance expression of mapping error between two corresponding points of a match can then be converted into the DLT form in Eq. (27) ③ . Accordingly, the total DLT-based cost function takes the following form:

$$\begin{aligned}
E = &\sum_{j=1}^{N_r} C(f_z(\mathbf{p}_j)y_j^t - f_y(\mathbf{p}_j)) + \sum_{j=1}^{N_r} C(-f_z(\mathbf{p}_j)x_j^t + f_x(\mathbf{p}_j)) + \\
&\sum_{j=1}^{N_r} C(f_y(\mathbf{p}_j)x_j^t - f_x(\mathbf{p}_j)y_j^t) + \lambda \sum_{k=1}^{8} \tilde{\mathbf{w}}_k^T G \tilde{\mathbf{w}}_k
\end{aligned} \tag{28}$$

By substituting Eq. (15) and Eq. (27) ② to Eq. (28), we obtain the final energy function as following:

$$\begin{aligned}
\arg\min_E = \arg\min_{\{\tilde{\mathbf{w}}_k, s_k\}} &\sum_{j=1}^{N_r} C\Big(\Big(f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1\Big)y_j^t - \Big(f_4(\mathbf{p}_j)x_j^r + f_5(\mathbf{p}_j)y_j^r + f_6(\mathbf{p}_j)\Big)\Big) + \\
&\sum_{j=1}^{N_r} C\Big(-\Big(f_7(\mathbf{p}_j)x_j^r + f_8(\mathbf{p}_j)y_j^r + 1\Big)x_j^t + \Big(f_1(\mathbf{p}_j)x_j^r + f_2(\mathbf{p}_j)y_j^r + f_3(\mathbf{p}_j)\Big)\Big) + \\
&\sum_{j=1}^{N_r} C\Big(\Big(f_4(\mathbf{p}_j)x_j^r + f_5(\mathbf{p}_j)y_j^r + f_6(\mathbf{p}_j)\Big)x_j^t - \Big(f_1(\mathbf{p}_j)x_j^r + f_2(\mathbf{p}_j)y_j^r + f_3(\mathbf{p}_j)\Big)y_j^t\Big) \\
&+ \lambda \sum_{k=1}^{8} \tilde{\mathbf{w}}_k^T G \tilde{\mathbf{w}}_k; \\
&\text{where } f_k(\mathbf{p}_j) = s_k + \sum_{i=1}^{M} \tilde{\mathbf{w}}_k(i) g(\mathbf{p}_j, \tilde{\mathbf{p}}_i) | k = 1, \ldots, 8.
\end{aligned} \tag{29}$$

The optimization of Eq. (29) can be processed by gradient descent, as elaborated in Section 3.1.1. After obtaining the optimal $\tilde{\mathbf{w}}_k$ and $s_k$, we can compute the projective transformation model $\mathbf{H}(\mathbf{p})$ at any corresponding location $\{\mathbf{p}\}$ with a noisy value $\{\hat{\mathbf{q}}\}$ according to Eq. (22) and the form of $f_k(.)$. Then, the model can be used to verify the correspondence hypothesis to differentiate the true and false point matches (can be feature point matches or intersection point matches) as follows:

$$\begin{aligned}
&\text{accept}(\mathbf{p}) = \text{true} \\
&\text{if } \sqrt{((f_z(\mathbf{p})y^t - f_y(\mathbf{p}))^2 + (-f_z(\mathbf{p})x^t + f_x(\mathbf{p}))^2 + (f_y(\mathbf{p})x^t - f_x(\mathbf{p})y^t)^2} < \epsilon_v.
\end{aligned} \tag{30}$$

## 3.2. Projective-constrained line matching

### *3.2.1. Pair-to-pair matching*

Considering the endpoints of the detected line segments are usually indefinite, exploiting the geometry constraint from endpoints for line segment matching is difficult. By contrast, the intersections of line pairs have more stable

properties. For any two lines that are coplanar in physical surface, their intersection point is invariant to projective transformation in image space (Andrew, 2001) , which can mathematically described as follows:

$$(\mathbf{H}\mathbf{l}_1) \times (\mathbf{H}\mathbf{l}_2) = |\mathbf{H}|\mathbf{H}^{-T}(\mathbf{l}_1 \times \mathbf{l}_2), \tag{31}$$

where $\mathbf{l}_1$ and $\mathbf{l}_2$ are a pair of line segments in RI and $\mathbf{H}$ denotes the projective transformation induced by two views of the corresponding planar surface in image RI and TI. According to Eq. (31), we can establish the correspondence between the intersections of line pairs from RI and TI, by which the line pairs are matched. Then, we distill the line-to-line matching results from the pair-to-pair correspondences. In the following, we realize the pair-to-pair matching in three steps, namely, the line pair generation, putative line pair matching and the verification of line pair correspondence hypotheses.

**Line pair generation**: In the following, we take the generation of line pairs for RI as an example. The process for TI is the same. Let $L^r = \{l^r_1, l^r_2, \dots, l^r_{N_l}\}$ be the set of $N_l$ line segments extracted from RI, we can get the set of initial line pairs $LP^r$ by arbitrarily grouping two line segments into pairs in the form of

$$LP^r = \{P^r_k(\mathbf{I}^r_k, l^r_{i_k}, l^r_{j_k})\}, \tag{32}$$

where $\mathbf{I}^r_k$ denotes the image coordinates of intersection of $l^r_{i_k}$ and $l^r_{j_k}$. However, the set of initial line pairs contain a large number of quasi-parallel and distant line pairs. The intersections of the quasi-parallel line pairs are near the vanishing point, which are far away from the image range and susceptible to image distortion. Meanwhile, line segments that are far away from each other in image space may significantly break the coplanar assumption. These two types of line pairs will bring not only negative effects, but also heavy computational burden to the following process. Therefore, we filter out redundant line pairs with small cross angles and ensure the coplanar assumption of line pairs by restricting the distance between their endpoint and intersection. A line pair $P^r_k(\mathbf{I^r}_k, l^r_{i_k}, l^r_{j_k}) \in$ RI is selected if the following condition is satisfied:

$$\begin{cases} \min\{d(\mathbf{I}^r_k, \mathbf{a}_1), d(\mathbf{I}^r_k, \mathbf{b}_1)\} \leqslant T_d \|\mathbf{a}_1 - \mathbf{b}_1\|_2 \\ \min\{d(\mathbf{I}^r_k, \mathbf{a}_2), d(\mathbf{I}^r_k, \mathbf{b}_2)\} \leqslant T_d \|\mathbf{a}_2 - \mathbf{b}_2\|_2 \\ \theta_k \geqslant T_\theta \end{cases} \tag{33}$$

where $(\mathbf{a}_1, \mathbf{b}_1)$ and $(\mathbf{a}_2, \mathbf{b}_2)$ are endpoints of line segments $l^r_{i_k}$ and $l^r_{j_k}$ respectively, and $\theta_k$ is the cross angle of the pair of line segments at intersection $\mathbf{I}^r_k = (x^r_k, y^r_k)$.

After the removal of redundant line pairs for RI and TI, we obtain two sets of refined line pairs for each image respectively, which are denoted as follows:

$$\begin{cases} LP^r = \{P^r_k(\mathbf{I}^r_k, l^r_{i_k}, l^r_{j_k}) | k = 1, 2, \dots, N_{lp}\} \\ LP^t = \{P^t_k((\mathbf{I}^t_k, l^t_{i_k}, l^t_{j_k}) | k = 1, 2, \dots, M_{lp}\} \end{cases} \tag{34}$$

**Putative Line pair matching**: We first construct the putative matches for the obtained line pairs from RI and TI and then verify the line pair correspondence hypotheses to discover the inlier matches.

The putative line pair matches are formed by finding correspondence between intersection points of line pairs with epipolar constraint, which can reduce the search range of corresponding intersection points from 2D to 1D (Schmid and Zisserman, 1997). Let $\mathbf{X}^r$ and $\mathbf{X}^t$ be the coordinates of two corresponding points $p^r$ and $p^t$ from RI and TI respectively, the epipolar constraint can be expressed as follows:

$$(\mathbf{X}^t)^T \mathbf{F} \mathbf{X}^r = 0 \tag{35}$$

where $\mathbf{F}$ is the fundamental matrix. Let $e' = \mathbf{F}\mathbf{X}^r$, $e'$ then represents the equation of epipolar line (on TI) corresponding to point $p^r$. The epipolar constraint restricts that the corresponding point $p^t$ of $p^r$ should lie on the epipolar line corresponding to $p^r$. Based on this fact, we use the high-quality point matches obtained from a global projective transformation modeling to calculate the fundamental matrix $\mathbf{F}$. Then, for each line pair $P^r_k$ in $LP^r$, we estimate

the epipolar line corresponding to its intersection $\mathbf{I}_k^r$ by $e_k' = \mathbf{F}\mathbf{I}_k^r$ and the line pairs $P_t^t$ in $LP^t$ satisfy the following condition

$$d(e_k', \mathbf{I}_t^t) \leqslant T_e, \tag{36}$$

are selected as the candidate matches of $P_k^r$. In practice, a distance threshold $T_e$ is generally applied to tolerate the inevitable deviation of epipolar lines and corresponding intersection points. However, this tolerance can not deal with large estimation error of epipolar lines and this fact lacks consideration in previous works. In our matching pipeline, we use the high-quality point matches obtained from global projective transformation modeling to guarantee the calculation of accurate fundamental matrix, thereby mitigating the estimation error of epipolar lines as far as possible.

**Line pair correspondence verification**: After applying the epipolar constraint, we obtain the putative line pair matches. For simplicity, we use the intersection correspondences to denote the set of line pair matches as follows:

$$\mathcal{M}_{in} = \{(\mathbf{I}_k^r, \mathbf{I}_k^t) | \mathbf{I}_k^r = (x_k^r, y_k^r), \mathbf{I}_k^t = (x_k^t, y_k^t), k = 1, 2, \dots, N_{in}\}. \tag{37}$$

As epipolar constraint is a weak constraint, $\mathcal{M}_{in}$ may contain many redundant false line pair matches. To obtain good line pair matches, we use the strict geometric constraint imposed by globally modeled projective transformation to verify the line pair correspondence hypotheses. For each intersection match $\{\mathbf{I}_k^r, \mathbf{I}_k^t)\}$ in $\mathcal{M}_{in}$, we can get its proxy defined as $\{\mathbf{p}_k = (\mathbf{I}_k^r, \mathbf{O}_k), \hat{\mathbf{q}}_k = \mathbf{I}_k^t\}$, where $\mathbf{O}_k = (x_k^t - x_k^r, y_k^t - x_k^r)$. According to Eq. (22) and Eq. (29), we can compute the projective transformation $\mathbf{H}(\mathbf{p}_k)$ and the components $\{f_x(\mathbf{p}_k), f_y(\mathbf{p}_k), f_z(\mathbf{p}_k)\}$ for each intersection match. By inputting the computed parameters together with $\{\mathbf{p}_k, \hat{\mathbf{q}}_k\}$ into Eq. (30), we can finally distinguish whether an intersection match $\{(\mathbf{I}_k^r, \mathbf{I}_k^t)\}$ is a true or false match. Those false intersection matches are rejected from $\mathcal{M}_{in}$ to exclude the corresponding line pair matches for subsequent line-to-line matching.

### *3.2.2. Line-to-line matching*

In this part, we detail how to distill the exact line-to-line matches from the pair-to-pair line correspondences. Two problems need to be solved: 1) each line pair match includes undetermined number of line-to-line matches (one, two or zero); 2) a single line segment may exist in different combinations of line pairs and can be matched repeatedly, producing different matching results. To this end, we verify the possible line-to-line correspondence hypotheses for each line pair match by the computed projective transformation model, which is followed by a E-distance voting with a crosscheck to remove the redundant and false line segment matches.

We utilize the projective transformation at line pair intersection, which has been obtained in the step of line pair correspondence verification. Since we have imposed coplanarity constraint during line pair generation, the line segments in a line pair should share similar projective transformation with their intersection. Based on the estimated projective transformation at the intersection, we map each pair of line segments from RI to TI and verify the line-to-line correspondence hypotheses by the similarity between the mapped and the assumptive matched line segment. We refer to this similarity as the mapping similarity. Given a line pair $(l_i^r, l_j^r)$ in RI and its corresponding line pair $(l_m^t, l_n^t)$ in TI, we can generate four combinations to indicate the possible line-to-line matches $(l_i^r, l_m^t)$, $(l_i^r, l_n^t)$, $(l_j^r, l_m^t)$ and $(l_j^r, l_n^t)$. We express the pairwise mapping similarity between these combinations in a $2 \times 2$ similarity matrix:

$$\begin{pmatrix} sim(l_i^r, l_m^t) & sim(l_i^r, l_n^t) \\ sim(l_j^r, l_m^t) & sim(l_j^r, l_n^t) \end{pmatrix} \tag{38}$$

where $sim(.)$ denotes the mapping similarity between two line segments of a combination. The similarity is measured in two aspects: the overlapping rate and the mapping error (i.e., the distance between two line segments). An example is shown in Figure 2.

In Figure 2, $A'B'$ ($\hat{l}_i^r$) is the mapped version of $AB$ ($l_i^r$). The overlapping rate between $l_i^r$ and $l_m^t$ is $A'D'/CD$ and the mapping error is the minimum of $CC'$ and $DD'$. The combinations with a small overlapping rate or a large mapping error are considered not similar and the values in the similarity matrix are set to zero accordingly. Specifically, the value of mapping similarity between two line segments $(l_i^r, l_m^t)$ can be calculated as

$$sim(l_i^r, l_m^t) = \begin{cases} 0 & \text{if } overlapping < P_o \text{ or } \min\{CC', DD'\} > P_d \\ e^{-\min\{CC', DD'\}} & \text{else} \end{cases} \tag{39}$$

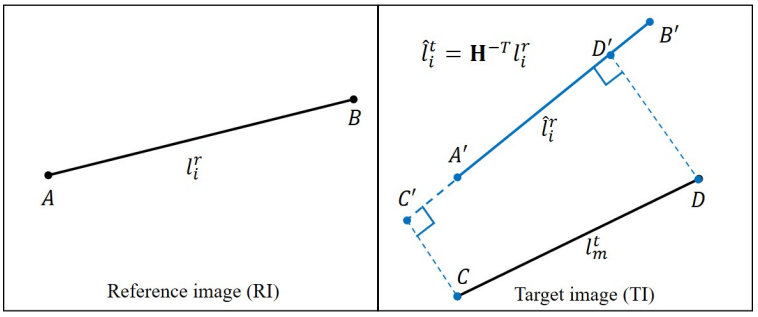



**Figure 2:** Distance and overlapping rate between the two line segments of a combination: $AB$ ($l_i^r$) and $CD$ ($l_m^t$). $A'B'$ ($\hat{l}_j^t$) is the mapped version of $AB$ in TI. The mapping similarity between $AB$ and $CD$ is measured by the distance and overlapping between $A'B'$ and $CD$. The distance is the minimum of $CC'$ and $DD'$ and the overlapping rate is the ratio $A'D'/CD$.

An illustration of the various similarity values calculated from different cases is shown in Figure 3. In Figure 3(a), the similarity value $sim(l_3^r, l_6^t)$ measured from $(l_i^r, l_m^t)$ is set to zero, as $D_{36} > P_d$. Similarly, $sim(l_3^r, l_6^t)$ in Figure 3(b) (case 2) is also set to zero. In Figure 3(c), there is no overlap between line segments from two RI and TI images, hence the values for all elements in the derived similarity matrix are set to zero.

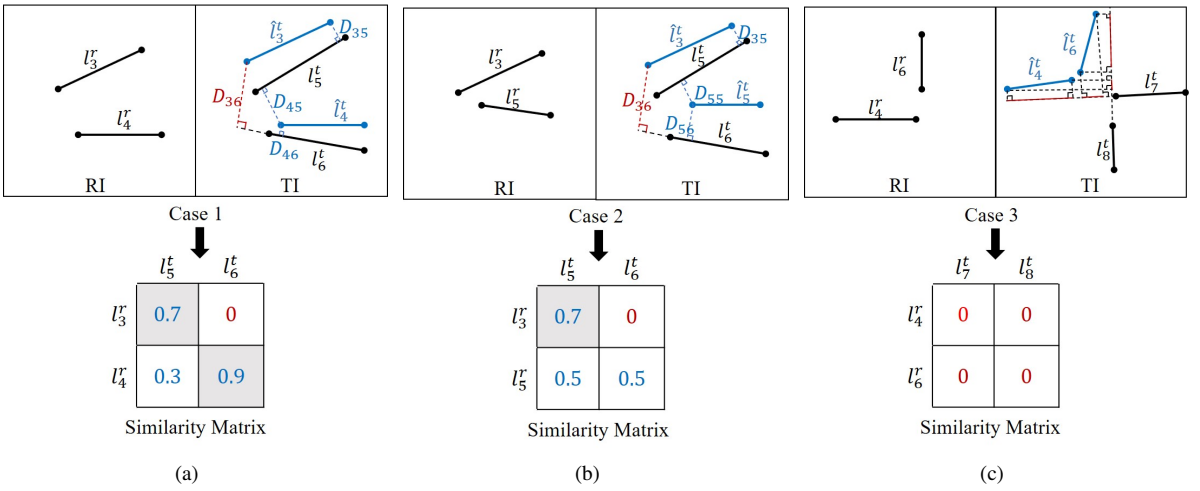



**Figure 3:** The various similarity values calculated from cases (a), (b) and (c).

From Figure 3, we can see that a line pair match can have two, one or zero line-to-line matches. To select the candidate line-to-line matches from the four combinations of a line pair match, we can use the similarity values encoded in the corresponding similarity matrix. We select the combination(s) with the maximum value on both column and row of the similarity matrix as the candidate line segment match(es). For example, combinations $(l_3^r, l_5^t)$ and $(l_4^r, l_6^t)$ in Figure 3(a), and combination $(l_3^r, l_5^t)$ in Figure 3(b) are chosen as the candidate line segment matches; whereas there are no combinations chosen as the candidate line segment matches in Figure 3(c). However, in Figure 3(a) and (b), it is possible that a line segment $(l_3^r)$ is matched repeatedly with different mapping similarities, as it may be included in different line pairs. Typically, the repeatedly matching circumstance occurs when one line segment intersects with multiple line segments. To circumvent this, we construct a vote matrix to accumulate these similarity values, as shown in Figure 4.

In Figure 4, the vote matrix $\mathbf{V}$ is an $N_l \times M_l$ matrix whose element $v_{ij}$ represents the cumulative mapping similarity between $l_i^r$ and $l_j^t$ and each element $v_{ij}$ is initialized with the value '0'. We then traverse all the obtained similarity matrices to assign the corresponding elements of vote matrix with values and only the similarity values of those selected combinations (candidate line matches) are used, as indicated by the dotted lines in Figure 4.

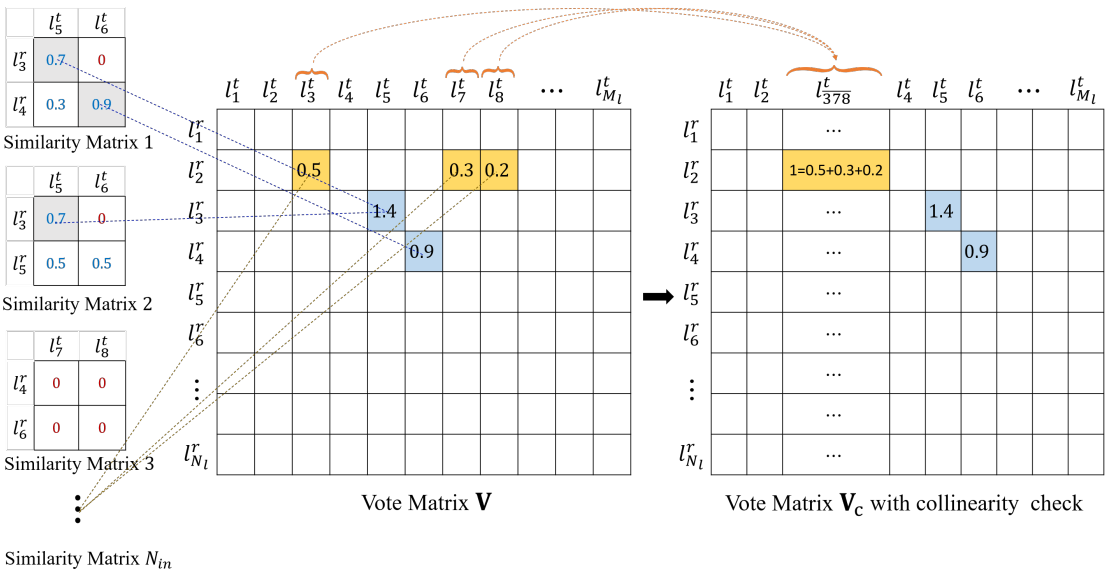



**Figure 4:** Vote matrix construction. Line combination(s) with a maximum value in both row and column of the similarity matrix are chosen and added to the vote matrix to construct the initial vote matrix $\mathbf{V}$. The colinearity check is then applied to merge the quasi-parallel segments to obtain final vote matrix $\mathbf{V}_c$.

Theoretically, the indices of the maximum mapping similarity in each row and column of $\mathbf{V}$ in Figure 4 suggest the matched line segments. However, it is possible that there exist quasi-parallel line segments from different line pairs (mainly resulted from fragmented line detection results). As a result, the vote matrix may generate one-to-many matches. We conduct a collinearity check on these matches with a distance threshold of $C_d$ and a cross angle threshold of $C_\theta$ to detect the quasi-parallel line segments. The calculation of distance and cross angle can refer to Figure 2 and Eq. (33). We merge the detected quasi-parallel line segments and accumulate their similarity values to get a new vote matrix $\mathbf{V}_c$ with the colinearity check. For example, in Figure 4, line segments $l_2^r$ and $l_3^t$, $l_7^t$ and $l_8^t$ in vote matrix $\mathbf{V}$ can possibly form one-to-many matches. We then perform the colinearity check on line segments $l_3^t$, $l_7^t$ and $l_8^t$, and if each of the two line segments have a distance smaller than $C_d$ and a cross angle smaller than $C_\theta$, their similarity values are merged as shown in $\mathbf{V}_c$. Finally, we perform a voting step on the vote matrix $\mathbf{V}_c$ so that line segment pairs with the maximum mapping similarity values on both row and column of $\mathbf{V}_c$ are voted as line segment matches. As the similarity values are mainly calculated based on the $e^x$ distance (in Eq. (39)), we term the construction of vote matrix and the voting step as the E-distance voting. To guarantee the high correctness of the obtained line segment matches, we map line segments from TI to RI, and reconduct the E-distance voting step on RI. We crosscheck the voting results, and the final line-to-line matching results are the intersection of candidate line segment matches selected by E-distance voting on both RI and TI.

## 4. Experiments

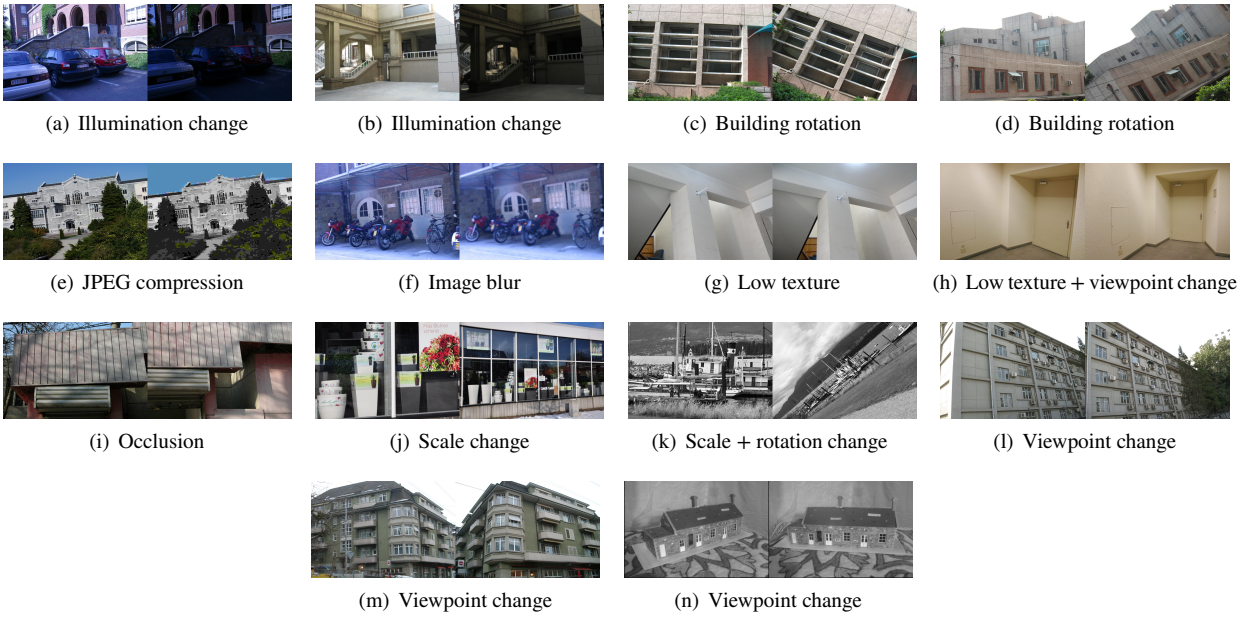
We conducted experiments on two challenging datasets, including a commonly-used benchmark dataset and a specially constructed street view dataset (termed as local dataset), to comprehensively test the capacity and generalization ability of the proposed SLEM. An in-depth analysis is also provided to reveal the advantages of our SLEM in detail. In the following subsections, we elaborate on the experimental settings, the comparison with other methods, and the experimental analysis.

### 4.1. Experimental setup

#### *4.1.1. Dataset*

**Benchmark dataset.** This is a commonly-used benckmark dataset for performance evaluation of line segment matching. The line segments of this dataset are detected by LSD detector (Von Gioi et al., 2008) and the corresponding ground truth matches are constructed by Li et al. (2016a). Following the previous works, we select fourteen pairs of images with some independent or compound representative line matching problems from the benchmark dataset for

experiment. These problems include illumination, scale, rotation and viewpoint change, image blur, occlusion, poor texture and JPEG compression, as illustrated in Figure 5.



(a) Illumination change (b) Illumination change (c) Building rotation (d) Building rotation

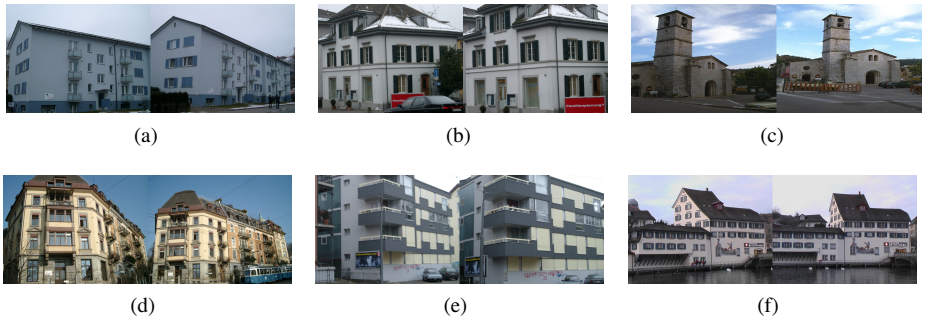
(e) JPEG compression (f) Image blur (g) Low texture (h) Low texture + viewpoint change

(i) Occlusion (j) Scale change (k) Scale + rotation change (l) Viewpoint change

(m) Viewpoint change (n) Viewpoint change

**Figure 5:** Fourteen image pairs from benchmark dataset, which cover different kinds of common problems in the line segment matching task.

**Local dataset.** As our aim is to develop a robust line matching method with a special concern on the challenging street-level scenes, we additionally select six image pairs from Zubud (Shao et al., 2003) and Valbonne Church (Schaffalitzky and Zisserman, 2002) datasets to form a local dataset for further performance evaluation. The local dataset contains images of buildings shot from different viewpoints and under different illumination. Compared with the benchmark dataset, local dataset contains more complex scenes with large areas of low texture, repetitive pattern and extreme viewpoint change. The LSD line detector is utilized to extract the line segments from each pair of images, followed by a manual check to generate the line-to-line matches, which are used as the ground truth labels.



(a) (b) (c)

(d) (e) (f)

**Figure 6:** Six street-level image pairs from local dataset.

### *4.1.2. Implementation details*

Our method begins with point feature matching and line segment detection. Following previous methods (Lin et al., 2017), we adopt A-Sift (Yu and Morel, 2011) to extract the feature points and implement nearest-neighbor matching computed by nearest neighbor distance ratio to establish the putative point matches, which are then filtered by a ratio test (Muja and Lowe, 2014). For line segment detection, we adopt the LSD detector with the mexopencv implementation to keep using the same line segment detector for dataset construction.

**Table 1**
Parameter setting.

| | Name | Explanation | Value |
|---|---|---|---|
| Parameters for point matching | $N_t$ | Number of data points in Eq. (20) | 1000 |
| | $N_r$ | Number of data points in Eq. (29) | 2000 |
| | $M$ | Number of representative data points in Eq. (29) | 300 |
| | $\lambda$ | Weight of the penalization term in Eq. (20) and Eq. (29) | 1.1 and $1.1N_r/M$ |
| | $\gamma$ | Radius of Gaussian radial basis function in Eq. (11) | 1 |
| | $\epsilon_{likelihood}$ | Threshold for coherence probabilistic model filtering in Eq. (21) | 0.01 |
| | $\epsilon_v$ | Threshold for feature(intersection) point correspondence verification in Eq. (30) | 0.01(0.05) |
| Parameters for line matching | $T_d$ | Distance threshold for line pair generation Eq. (33) | 0.2 |
| | $T_\theta$ | Cross angle threshold for line pair generation in Eq. (33) | $\pi/18$ |
| | $T_e$ | Threshold for putative line pair matching in Eq. (36) | 0.05 |
| | $P_d$ | Mapping error threshold for mapping similarity calculation in Eq. (39) | 0.1 |
| | $P_o$ | Overlapping rate threshold for mapping similarity calculation in Eq. (39) | 0.5 |
| | $C_d$ | Distance threshold for collinearity check in vote matrix construction | 0.5 |
| | $C_\theta$ | Cross angle threshold for collinearity check in vote matrix construction | $\pi/18$ |

To evaluate the robustness of the proposed method, we fixed all parameters on different datasets. Table 1 lists the setting of the used parameters. Our regression formulation allows to compute a global model only from a set of available data. To improve computational efficiency, we uniformly sample $N_t = 1000$ point matches from the original putative match set $\mathcal{M}$ for regressing the likelihood function in Eq. (20). Similarly, we also uniformly sample $N_r = 2000$ point matches from the refined match set $\mathcal{M}_r$ and select $M = 300$ representative points for global projective transformation modeling in Eq. (29). In projective transformation modeling, $\lambda$ and $\gamma$ reflect the amount of smoothness regularization (Yuille and Grzywacz, 1988). In practice, these two parameters should be adaptable to the motion type and matching scenarios with different degrees of spatial structural complexity. For complex scenes containing multiple planes, a smaller value of $\lambda$ and $\gamma$ is preferred as it facilitates fine modeling of motions, whereas a larger value prevents the model from overfitting for less complex image motions. $\lambda$ should be also correlated with the number of sampled representative data points. In this study, we set $\lambda = 1.1N_t/N_t$ in Eq. (20), $\lambda = 1.1N_r/M$ in Eq. (29) and set $\gamma = 1$. Besides, we set $\epsilon_{likelihood} = 0.01$ in Eq. (21) for filtering incoherent point matches and set $\epsilon_v = 0.01$ and 0.05 for determining the correct feature point matches and line intersection point matches respectively.

In line matching pipeline, we set $T_d = 0.2$ and $T_\theta = \pi/18$ for line pair generation, which are less strict values that help improve the overall recall of the matched line segments. For putative line pair matching, we set $T_e = 0.01$. In vote matrix construction, we set $P_d = 0.1$ and $P_o = 0.5$ for measuring mapping similarity and set $C_d = 0.5$ and $C_\theta = \pi/18$ for the collinearity check on rows and columns of the vote matrix.

#### *4.1.3. Evaluation metric*

We evaluate the performance of the proposed SLEM and the two comparison methods by three main metrics: Precision, Recall and F-score, which are calculated as follows:

$$\text{Precision} = \frac{TP}{TP+FP}; \text{Recall} = \frac{TP}{TP+FN}; \text{F-score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}, \tag{40}$$

where $TP$, $FN$, and $FP$ are the number of true positive (the correctly predicted line matches), false negative (the incorrectly predicted outlier line matches) and false positive (the incorrectly predicted inlier line matches) line matches in the final line-to-line matching results, respectively.

### 4.2. Results on benckmark dataset

To evaluate the performance of our proposed SLEM on the benchmark dataset, we choose two representative line segment matching methods with available implementation for comparison: HLPI (Jia et al., 2018) and LJL (Li et al., 2016b). Similar to SLEM, HLPI is a typical point correspondence-based line matching method, whereas LJL is mainly based on group matching. The quantitative comparison results are reported in Table 2.

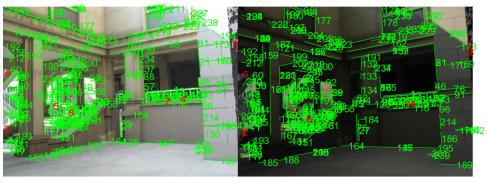
**Table 2**
Quantitative comparison of different line matching methods (HLPI (Jia et al., 2018) and LJL (Li et al., 2016b)) on the benchmark dataset (unit: %).

| Images | Precision | | | Recall | | | F-score | | | Total Number | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Our | HLPI | LJL | Our | HLPI | LJL | Our | HLPI | LJL | Our | HLPI | LJL |
| a | 93.8 | 88.6 | 88.0 | 99.8 | 85.7 | 84.8 | 96.7 | 87.1 | 86.4 | 433 | 394 | 392 |
| b | 96.8 | 92.4 | 86.5 | 100.0 | 87.3 | 70.0 | 98.8 | 89.8 | 77.4 | 247 | 224 | 192 |
| c | 96.4 | 95.9 | 95.5 | 97.6 | 84.6 | 86.8 | 97.0 | 89.9 | 90.9 | 421 | 367 | 378 |
| d | 97.2 | 93.5 | 90.5 | 100.0 | 84.1 | 82.6 | 98.6 | 88.5 | 86.4 | 355 | 310 | 315 |
| e | 90.9 | 64.6 | 67.3 | 91.2 | 55.8 | 57.1 | 91.0 | 59.9 | 61.7 | 397 | 342 | 336 |
| f | 90.2 | 72.5 | 74.4 | 94.8 | 68.2 | 67.1 | 92.5 | 70.3 | 70.6 | 387 | 346 | 332 |
| g | 92.7 | 90.5 | 80.6 | 62.3 | 31.1 | 41.0 | 74.5 | 46.3 | 54.3 | 41 | 21 | 31 |
| h | 83.3 | 95.2 | 93.1 | 49.2 | 32.8 | 44.3 | 61.9 | 48.8 | 60.0 | 36 | 21 | 29 |
| i | 91.3 | 94.7 | 94.0 | 51.7 | 70.9 | 38.9 | 66.0 | 81.1 | 55.1 | 115 | 152 | 84 |
| j | 100.0 | 67.2 | 66.7 | 83.3 | 59.7 | 44.4 | 90.9 | 63.2 | 53.3 | 60 | 64 | 48 |
| k | 80.4 | 37.8 | 47.9 | 56.7 | 25.0 | 30.4 | 66.5 | 30.1 | 37.2 | 158 | 148 | 142 |
| l | 98.9 | 97.8 | 92.3 | 95.3 | 86.5 | 80.7 | 97.1 | 91.8 | 86.1 | 854 | 784 | 776 |
| m | 35.7 | 19.3 | 31.7 | 14.2 | 7.8 | 13.9 | 20.3 | 11.1 | 19.3 | 168 | 171 | 186 |
| n | 76.7 | 57.9 | 74.1 | 51.8 | 28.9 | 20.7 | 61.8 | 38.5 | 32.3 | 206 | 152 | 85 |
| mean | **87.5** | 76.3 | 77.3 | **74.8** | 57.7 | 54.5 | **79.5** | 64.0 | 62.2 | **277.0** | 249.7 | 237.6 |

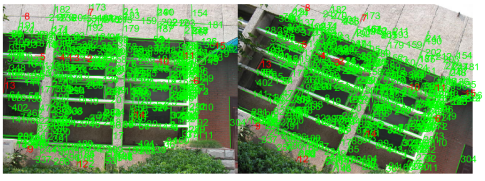
In Table 2, the proposed SLEM achieves the top performance on almost all image pairs in all metrics, obtaining an average precision, recall and F-score of 87.5%, 74.8% and 79.5%, respectively. SLEM outperforms HLPI and LJL by 11.2% and 10.2% in terms of precision. In addition, SLEM gains over HLPI and LJL by 17.1% and 20.3% in terms of recall, excelling the two comparison methods by an extremely large margin. As F-score is a comprehensive evaluation metric combining both precision and recall, the higher mean precision and recall of SLEM result in a higher mean F-score. Moreover, the proposed SLEM procures a larger number of true line segment matches than the two comparison methods. The numeric comparison results demonstrate that SLEM is more robust than the other methods for matching line segments under different matching scenarios. This excellent performance of SLEM is mainly attributed to the advantages brought by our regression-based global projective transformation modeling to the various steps of line segment matching. The high-quality matches enable the estimation of more accurate fundamental matrix, thereby yielding good putative line pair matches with epipolar constraint. Moreover, our estimated smoothly varying projective transformation model can well approximate the underlying image transformation even under complex wide baseline matching scenarios. The model provides powerful strict geometric constraint to verify the intersection correspondence hypotheses, which filters out a majority of redundant and false line pair matches. These two steps essentially reduce the matching ambiguity and errors of the following line-to-to matching. As our estimated projective transformation is reliable and highly adaptable to line segments with varying lengths, we can precisely map a line segment from one view to another to ensure the accuracy of similarity measurement, further improving the precision of SLEM. The high recall of SLEM can also be attributed to the global non-parametric regression framework, as it permits line correspondence verification in areas with few or no intersection point matches. Our method also benefits from the collinearity check on fragmented line segments in E-distance voting, which successfully handles the one-to-many matching. The qualitative matching results are visualized in Figure 7.

To better reveal the efficacy of the proposed SLEM, the numeric results in Table 2 and the visualized results in Figure 7 are incorporated to provide a more comprehensive evaluation. Following the existing works, only results of the proposed method are visualized. Image pairs shown in Figure 7(a) and (b) contain illumination changes where all methods obtain satisfying matching results. Image pairs shown in Figure 7(c), (d), (j), (k) and (l) suffer from common rotation, scale and narrow-baseline viewpoint changes. From Table 2, SLEM and HLPI that employ geometric constraints computed from point correspondences for matching perform better than LJL that relies on the LJL structure descriptor (a type of intersection descriptor) for matching. This is probably due to two reasons: (1) point features extracted by sophisticated detectors are more effective for estimating the image transformation (*e.g.,* homography) than LJL structure (or intersections) in terms of number and distribution; (2) the descriptor of point features is more invariant

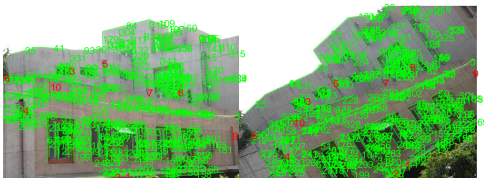
(a) Precision: 93.8%; Recall: 99.8%



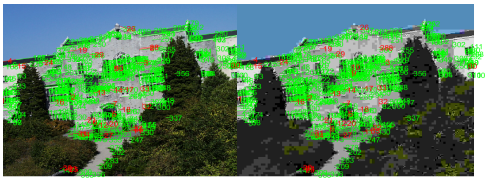
(b) Precision: 96.8%; Recall: 100%



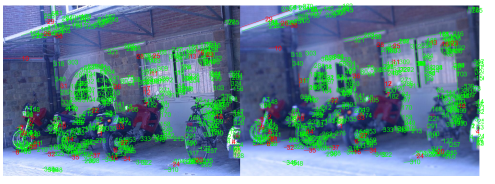
(c) Precision: 96.4%; Recall: 97.6%



(d) Precision: 97.2%; Recall: 100%

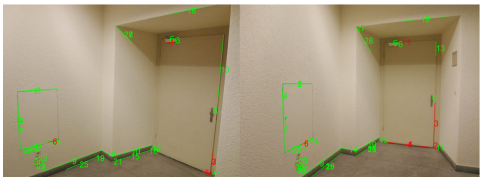


(e) Precision: 90.9%; Recall: 91.2%

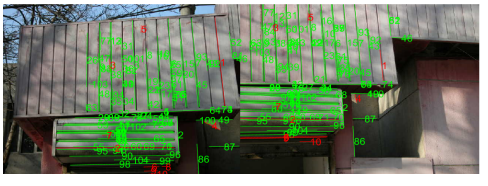


(f) Precision: 90.2%; Recall: 94.8%

(g) Precision: 92.7%; Recall: 62.3%

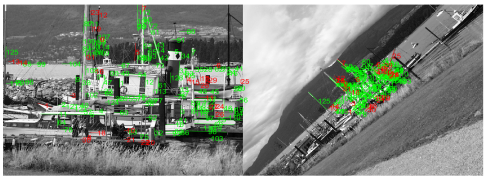


(h) Precision: 83.3%; Recall: 49.2%

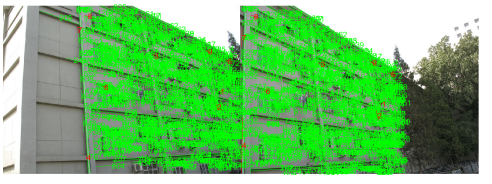


(i) Precision: 91.3%; Recall: 51.7%

(j) Precision: 100%; Recall: 83.3%

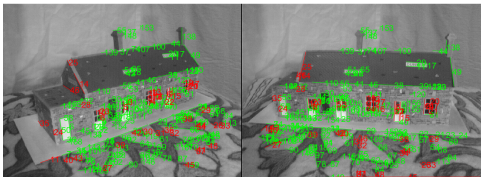


(k) Precision: 80.2%; Recall: 56.7%



(l) Precision: 98.9%; Recall: 95.3%

(m) Precision: 35.7%; Recall: 14.2%



(n) Precision: 76.7%; Recall: 51.8%

**Figure 7:** Qualitative results of the proposed method on benchmark Dataset. Correct matches are shown in green and incorrect ones are shown in red.

to various types of image changes than the LJL descriptor. However, HLPI appears to be unstable and performs worse than the proposed SLEM. The performance gap between HLPI and SLEM is more evident on image pair (j) and (k). The reason may be the lack of necessary consideration of HLPI on improving the matching quality of point features in complex scenarios, which severely limits the performance of subsequent line matching. Figure 7(e) and (f) are representatives of image blurring and compression. In these conditions, the descriptor similarity (both feature and LJL descriptors) of corresponding points is severely reduced, leading to a very high outlier rate that degrades the performance of LJL and HPLI. In comparison, SLEM is able to handle the high outlier rate via global motion regression and thus achieves stable line matching results.

Figure 7(g) and (h) are typical low-textured scenes. Given the scarcity of feature points and line segments, HLPI results in a significant decline of recall, indicating its high dependence on the density of point correspondences. In comparison, our global projective transformation modeling approach can well handle motion discontinuities and infer accurate image transformation even in regions with few point matches, thereby guaranteeing a relatively high recall in low-textured scenes. Figure 7 (m) and (n) depict the challenging wide baseline and complex street-level scenarios. All methods encounter a drastic performance drop compared with the above-mentioned image pairs, particularly for image pair (m). However, benefiting from the global projective transformation modeling, our method still achieves the top performance among the comparison methods. As street-level scenes are typical manmade scenes that have high demands on line structure reconstruction, a specially established street-view dataset is utilized for further evaluation and analysis.

### 4.3. Results on local dataset

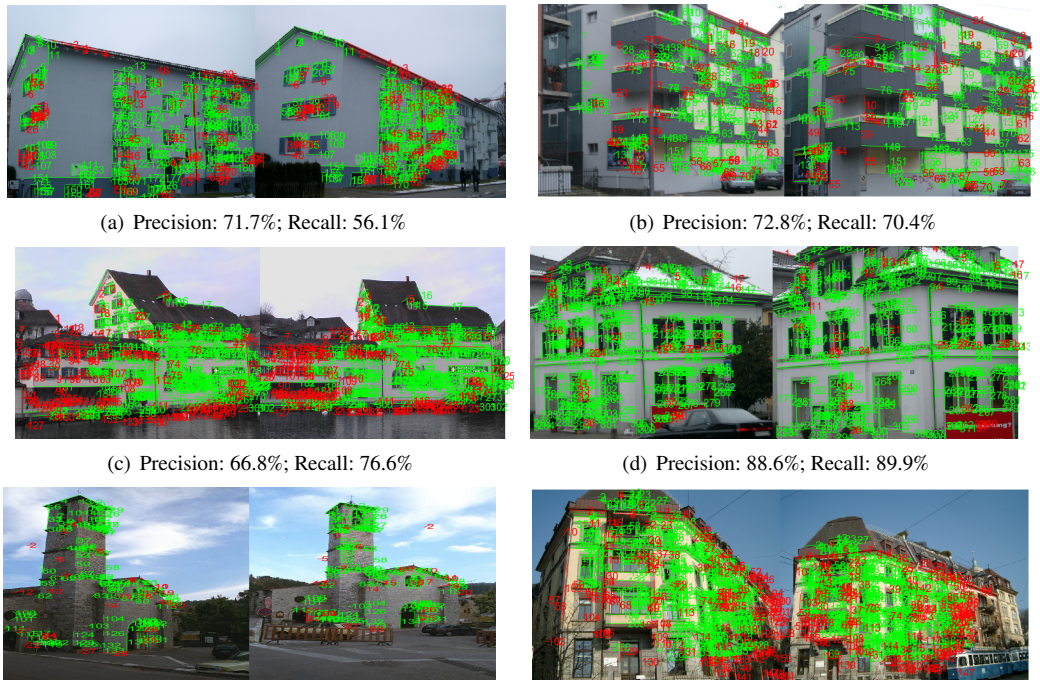
Street view images generally consist of buildings with complex facades. Although street view images provide abundant line structures, the frequently appearing low texture, repetitive pattern, complex multi-planar structure and large parallax make the line segment matching of street view images particularly challenging. To testify the efficacy of different methods in street-level scenarios, we conducted experiments on the local dataset. The quantitative results are listed in Table 3.

**Table 3**
Quantitative comparison of different methods (HLPI (Jia et al., 2018) and LJL (Li et al., 2016b)) on the local dataset (unit: %).

| Images | Precision | | | Recall | | | F-score | | | Total Number | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Our | HLPI | LJL | Our | HLPI | LJL | Our | HLPI | LJL | Our | HLPI | LJL |
| a | **71.7** | 59.0 | 48.3 | **56.1** | 38.6 | 30.2 | **58.6** | 46.7 | 37.2 | **251** | 210 | 201 |
| b | **72.8** | 46.2 | 64.8 | **70.4** | 45.6 | 68.1 | **71.6** | 45.9 | 66.4 | 261 | 266 | **284** |
| c | **66.8** | 51.2 | 55.6 | **76.6** | 54.6 | 55.1 | **71.4** | 52.8 | 55.4 | **452** | 420 | 390 |
| d | **88.6** | 76.9 | 75.9 | **89.9** | 77.8 | 66.3 | **89.3** | 77.4 | 70.8 | **352** | 351 | 303 |
| e | **82.7** | 47.5 | 60.5 | **65.0** | 36.9 | 47.6 | **72.8** | 41.5 | 53.3 | **162** | 160 | **162** |
| f | **68.9** | 23.7 | 43.3 | **54.0** | 21.0 | 36.9 | **60.6** | 22.2 | 39.9 | 486 | **549** | 529 |
| mean | **74.8** | 50.8 | 58.1 | **67.9** | 45.7 | 50.7 | **70.7** | 47.8 | 53.8 | 325.3 | **326.0** | 311.5 |

The overall situation in Table 3 is similar to that in Table 2, where our SLEM achieves the best performance on almost all image pairs in all metrics. SLEM gains over HLPI and LJL by 24.0% and 16.7% in mean precision, by 22.2% and 17.2% in mean recall, and by 22.9% and 16.9% in mean F-score, respectively. Although the two comparison methods can procure more number of total line matches in some image pairs, they also bring larger number of false line matches manifesting by a low precision. In general, the performance gain of the proposed SLEM over HLPI and LJL are indeed impressive, which strongly demonstrates its robustness on the challenging street-level scenarios. To better reveal the effectiveness of our method, the qualitative results are also visualized in Figure 8.

From Figure 8, the low texture, repetitive pattern, and complex multi-planar structure are common in street-level scenarios, which severely degrade the performances of the two comparison methods. HLPI uses RANSAC to compute homography locally to assist in the line matching, which will be ineffective in textureless regions. This problem becomes more severe when large area repetitive elements exist, which cause consistently wrong point matches that are hard to distinguish at a local scale. Moreover, RANSAC can not deal with motion discontinuities incurred by the multi-planar structure. As shown in Table 3, when the matching scenarios become more difficult in Figure 8(d) to (f)



(a) Precision: 71.7%; Recall: 56.1%

(b) Precision: 72.8%; Recall: 70.4%

(c) Precision: 66.8%; Recall: 76.6%

(d) Precision: 88.6%; Recall: 89.9%

(e) Precision: 82.7%; Recall: 65.0%

(f) Precision: 68.9%; Recall: 54.0%

**Figure 8:** Visualized matching results of the proposed SLEM on local dataset. Correct line matches are shown in green and incorrect ones are shown in red.

, the performance of HLPI degrades much more rapidly than the other methods. Specifically, the additional nuisance factors (the drastic scale and viewpoint changes) in Figure 8(f) further undermine the point matching and homography estimation of HLPI, resulting in a poorer performance than on other image pairs. LJL relies on group matching with the specially designed LJL structure descriptor that seems to be more stable than HLPI on matching line segments in street view images. However, the overall performance is still unsatisfactory, because the concurrent nuisance factors in street scenes can also bring ambiguities to LJL descriptor and undermine the local homography estimation from LJL matches.

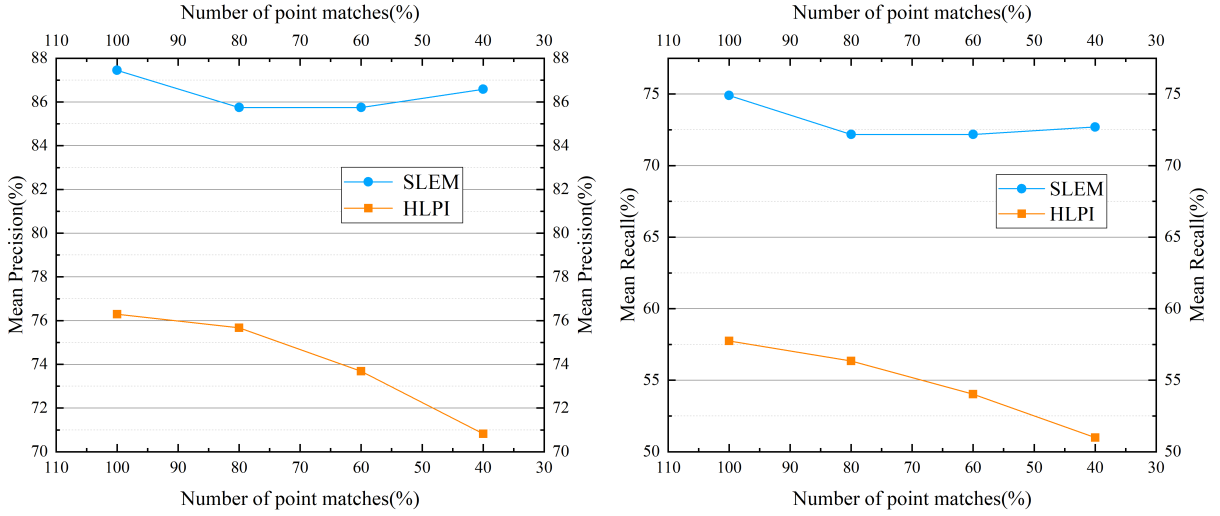
From the results in Figure 8 and Table 3, the proposed SLEM manifests its strong ability in handling the above nuisance factors. This strong performance can be explained in three aspects. First, our global projective transformation model can be computed from a subset of available data and extrapolate to new positions. This enables the robust estimation of underlying image transformation in low texture regions with insufficient point features. Second, the proposed non-parametric motion regression is able to incorporate information from across the whole image. Hence, the resultant global model can effectively discern the consistent false point matches caused by repetitive patterns, preventing the error information from propagating to line matching. Finally, our regression formulation can model the piecewise smoothly varying projective transformation as a global continuous model, which addresses the motion boundary issues incurred by multi-planar structure of building facades. In addition, the projective transformation can best represent the more general image transformation (combining affine with projectivity) in complex street-level scenarios. These properties together make our SLEM a robust line segment matcher that can achieve acceptable results even in the presence of notable viewpoint and scale changes, as shown in Figure 8 (f).

## 5. Analysis

As mentioned in the introduction, two major aspects affect the performance of point correspondence-based line matching methods: 1) the sufficiency of point matches; and 2) the reliability of the estimated image transformation. In this section, these two aspects are further analyzed with ablation experiments.

## 5.1. Robustness to the quantity of point matches

We first analyze the influence of the quantity of point matches on different point correspondence-based methods. We randomly select 100%, 80%, 60%, and 40% of the putative point matches (nearest neighbor A-sift matches after ratio-test) and respectively input them into the point correspondence-based matching pipeline of our SLEM and HLPI. The final line matching results of the two methods in terms of mean precision and recall are reported in Figure 9.




(a) Mean precision (b) Mean recall

**Figure 9:** Statistical results of line segment matching performance of the proposed SLEM and HLPI with varying number of putative point matches. (a) Mean precision. (b) Mean recall.

As shown in Figure 9(a), SLEM and HLPI achieve a mean precision of 87.5% and 76.3% respectively, when using all of the putative point matches for line segment matching. SLEM outperforms HLPI by 11.2% in precision. With the number of point matches reduced from 100% to 40% of the original putative matches, the maximum drop in mean precision for SLEM and HLPI are 1.7% and 5.5%. Compared with SLEM, HLPI undergoes a more significant drop in precision with the reduction of point matches. From Figure 9(b), as the number of point matches decreases, the change of recall for the two methods presents similar trends as that of precision. When the number of point matches decreases from 100% to 40%, the recall of SLEM and HILP witness a maximum decline of approximately 2.7% and 6.8% respectively. From these results, we can observe that our SLEM demonstrates a higher robustness to the number of point matches than HLPI. This reveals the advantages of the global projective modeling of SLEM, which enables the transformation model to be computed from a small available subset of the point matches and extrapolated to the other positions. In particular, the motion coherence of correct matches is enforced via the global optimization of our regression function, making our SLEM robust to outliers. However, a better spatial distribution of point matches and ratio between the inlier and outlier point matches are helpful to model estimation. This also explains why the precision and recall of the proposed SLEM increase when the number of point matches is reduced from 60% to 40%, since the sampling of putative point matches leads to a certain randomness in the distribution of inlier and outlier point matches. In comparison, HLPI uses RANSAC to estimate the homography for each local region around the line segments, putting a high requirement on the number of point matches around each of the line segment. In complex scenarios, the putative point matches are typically highly noisy and scattered. RANSAC can hardly estimate the correct homography, thereby propagating large errors to the line segment matching procedure, resulting in performance degradation.

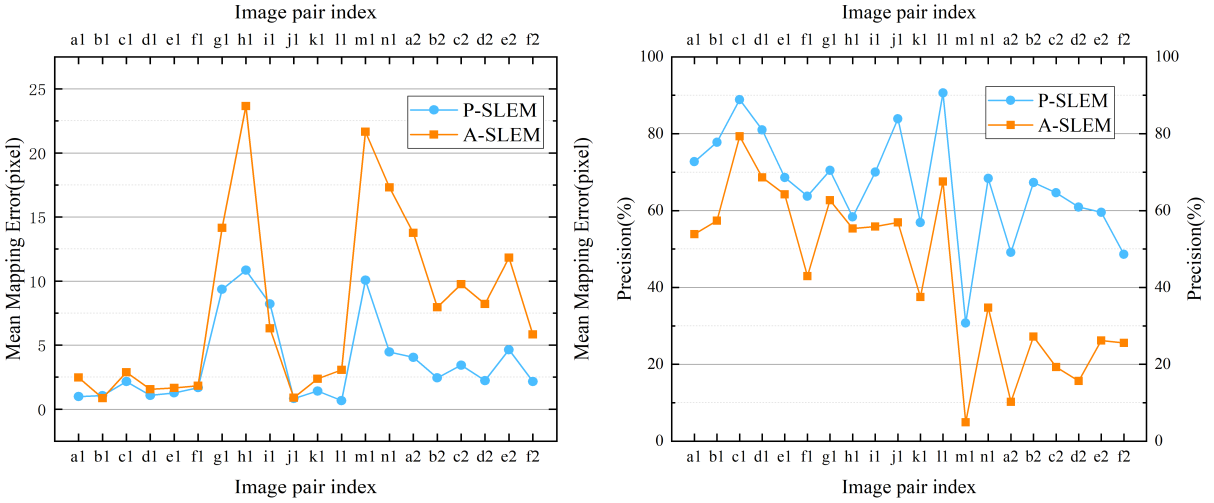
## 5.2. Necessity of using projective transformation

Affine transformation is typically used to approximate local image motions (Li et al., 2019; Cavalli et al., 2020). However, in real world scenarios, lines of varying length may undergo a more general projective transformation combining affine with projectivity. To verify whether it is necessary to use projective transformation for line segment mapping, we compare it with the commonly-used affine transformation (Lin et al., 2017). We replace the projective transformation with affine transformation in the regression function and estimate a global smoothly varying affine

model to assist in the line segment matching. We also remove the crosscheck operation so that the line-to-line matching results is purely determined by the transformation model that is selected. For convenience, we term the two solutions of using projective and affine transformation for line matching as projective SLEM (P-SLEM) and affine SLEM (A-SLEM), respectively. We apply P-SLEM and A-SLEM to the total 20 image pairs from the benchmark and local dataset. The performances of the two solutions are evaluated by the mapping error and the precision of the line segment matching results. The mapping error is defined as the mapping distance between two line segments, as shown in Figure 2. For evaluation, the mapping error between the mapped and the ground truth corresponding line segment is used. For each image pair, we calculate a mean mapping error for all the obtained line matches. The quantitative results are reported in Table 4.

**Table 4**
Comparison of A-SLEM and P-SLEM on the two datasets (unit: pixel for Mean mapping error; % for Precision).

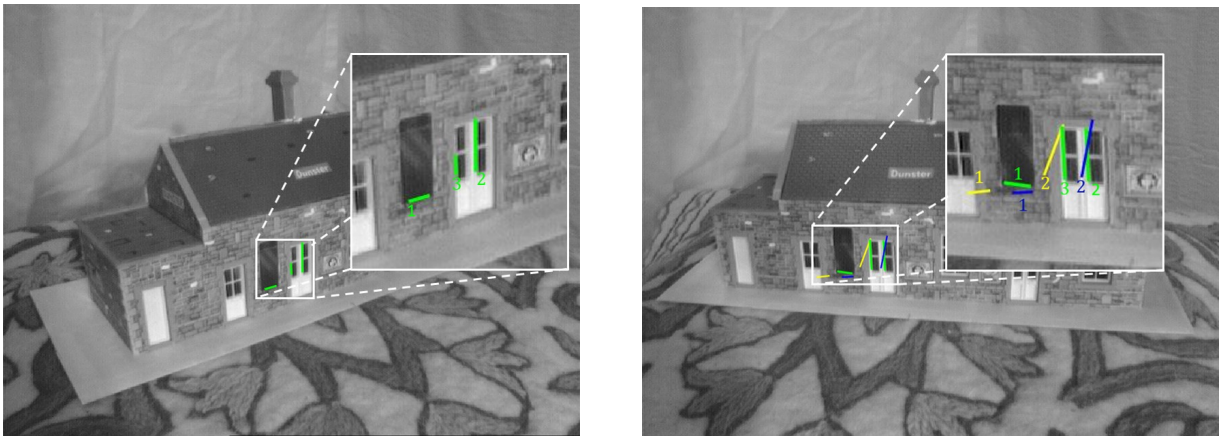
| Benchmark dataset | Mean mapping error | | Precision | | Local dataset | Mean mapping error | | Precision | |
|---|---|---|---|---|---|---|---|---|---|
| | P-SLEM | A-SLEM | P-SLEM | A-SLEM | | P-SLEM | A-SLEM | P-SLEM | A-SLEM |
| a | **0.982** | 2.469 | **72.7** | 53.9 | | | | | |
| b | 1.060 | **0.854** | **77.8** | 57.4 | a | **4.049** | 13.765 | **49.1** | 10.2 |
| c | **2.155** | 2.879 | **88.9** | 79.3 | | | | | |
| d | **1.075** | 1.555 | **81.0** | 68.7 | b | **2.443** | 7.951 | **67.3** | 27.2 |
| e | **1.267** | 1.662 | **68.6** | 64.2 | | | | | |
| f | **1.680** | 1.830 | **63.7** | 42.9 | c | **3.432** | 9.754 | **64.6** | 19.3 |
| g | **9.349** | 14.157 | **70.5** | 62.7 | | | | | |
| h | **10.847** | 23.661 | **58.4** | 55.3 | d | **2.227** | 8.215 | **60.9** | 15.6 |
| i | 8.218 | **6.299** | **70.0** | 55.8 | | | | | |
| j | **0.826** | 0.887 | **83.9** | 56.9 | e | **4.641** | 11.821 | **59.6** | 26.2 |
| k | **1.410** | 2.360 | **56.9** | 37.5 | | | | | |
| l | **0.669** | 3.064 | **90.6** | 67.6 | f | **2.160** | 5.831 | **48.6** | 25.6 |
| m | **10.062** | 21.652 | **30.7** | 4.9 | | | | | |
| n | **4.463** | 17.310 | **68.4** | 34.7 | | | | | |
| | | | | | Mean | **3.651** | 7.899 | **66.6** | 43.3 |




(a) Mean mapping error (b) Precision

**Figure 10:** Comparison of A-SLEM and P-SLEM on the two datasets. (a) Mean mapping error. (b) Precision. Image pairs a1-n1 are from benchmark dataset, while image pairs a2-f2 are from local dataset.

In Table 4, P-SLEM achieves the top performance on almost all image pairs in all metrics, obtaining a mean mapping error and a precision of 3.651 (in pixel) and 66.6%, respectively. P-SLEM reduces the mean mapping error of A-SLEM by 4.248 pixel and gains over A-SLEM by 23.3% in terms of precision. Particularly, P-SLEM largely reduces the mapping error of A-SLEM in some image pairs that are difficult to match (*e.g.,* image pair h and n from the benchmark dataset). The results reveal that the projective transformation adopted in P-SLEM is more adaptable to line segments of varying length than affine transformation deployed in A-SLEM, which can better approximate the real image transformation across large image region. This merit of projective transformation is critical to the performance of the final line segment matching, as indicated by the precision results of the two solutions. The mapping error reduction by P-SLEM brings considerable precision improvements for most of image pairs, resulting in significant mean precision boost. The improvements in both mapping error and precision for different image pairs are more clearly visible in Figure 10. Figure 11 shows an example of mapping line segments from RI to TI by different transformation models.



(a) Reference image (RI) (b) Target image (TI)

**Figure 11:** Effect of mapping line segments from RI to TI by different transformations. The green lines labeled with same number in (a) RI and (b) TI are ground truth line matches. The yellow and blue lines are obtained by mapping the green lines from RI to TI via affine and projective model, respectively. The mapping error of yellow lines is clearly visible.

In Figure 11, the green lines labeled with the same number in RI (Figure 11(a)) and TI (Figure 11(b)) are ground truth line matches. As an example, we map the line segments labeled as '1' and '2' from RI to TI by the affine (estimated by A-SLEM) and projective (estimated by P-SLEM) models, respectively. As shown in Figure 11(b), the yellow lines are obtained by the affine model, while the blue lines are obtained by the projective model. It is clear that the yellow lines deviate far from the ground truth line segments. Especially, the yellow line '2' is closer to the green line '3' than its corresponding line (green line '2'). In this case, the yellow line '2' will mislead its original line (green line '2' in RI) to be wrongly matched to the green line '3' in TI. In contrast, both the blue lines '1' and '2' approach the ground truth line segments very well, and thus their original lines in RI can potentially be correctly matched to green lines '1' and '2' in TI. The visual results in Figure 11 indicate that the projective model can more precisely map line segments from one view to another than affine model. This phenomenon also explains why the P-SLEM yields better line matching performance than A-SLEM, as the projective model can greatly reduce the potential false line matches resulted from inaccurate line mapping.

## 6. Conclusion

This study presented a method called SLEM for line segment matching based on a global modeling of piecewise smoothly varying projective transformation. The experimental results show that the proposed SLEM can achieve excellent performance consistently on two challenging datasets. The quantitative and quality comparison results also show that the proposed SLEM can outperform the existing methods by large margins, demonstrating its high robustness to the various types of scenes (*e.g.,* scenes with poor texture, repetitive pattern, and large viewpoint change). The

experimental analysis further verifies that SLEM is robust to the number of point matches applied for line segment matching. Moreover, the estimated projective transformation can more precisely map a line segment from one view to another than other transformations (*e.g.,* affine). The results indicate that SLEM is more capable of utilizing the strict geometric constraint for matching disambiguation than the group-based line matching method (*e.g.,* LJL). Compared with the point correspondence-based method (*e.g.,* HLPI), SLEM can robustly match line segments even in the regions with insufficient or highly noisy point matches, by taking advantages of global projective transformation modeling.

In the future, we will exploit the effectiveness of the proposed SLEM on the 3D reconstruction of manmade objects in street-level scenes. We will also focus on improving the matching performance in some scenarios that are extremely difficult to match, for example, the ground to aerial line matching, where image pairs undergo abrupt changes in viewpoint, scale, and appearance.

## References


Andrew, A.M., 2001. Multiple view geometry in computer vision. Kybernetes .

Bay, H., Ferraris, V., Van Gool, L., 2005. Wide-baseline stereo matching with line segments, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), IEEE. pp. 329–336.

Black, M.J., Anandan, P., 1996. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. Computer vision and image understanding 63, 75–104.

Cavalli, L., Larsson, V., Oswald, M.R., Sattler, T., Pollefeys, M., 2020. Handcrafted outlier detection revisited, in: European Conference on Computer Vision, Springer. pp. 770–787.

Chen, M., Qin, R., He, H., Zhu, Q., Wang, X., 2018. A local distinctive features matching method for remote sensing images with repetitive patterns. Photogrammetric Engineering & Remote Sensing 84, 513–524.

Fan, B., Wu, F., Hu, Z., 2010. Line matching leveraged by point correspondences, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE. pp. 390–397.

Fan, B., Wu, F., Hu, Z., 2012. Robust line matching through line–point invariants. Pattern Recognition 45, 794–805.

Huber, P.J., 1992. Robust estimation of a location parameter, in: Breakthroughs in statistics. Springer, pp. 492–518.

Jellal, R.A., Lange, M., Wassermann, B., Schilling, A., Zell, A., 2017. Ls-elas: Line segment based efficient large scale stereo matching, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE. pp. 146–152.

Jia, Q., Fan, X., Gao, X., Yu, M., Li, H., Luo, Z., 2018. Line matching based on line-points invariant and local homography. Pattern Recognition 81, 471–483.

Jia, Q., Li, Z., Fan, X., Zhao, H., Teng, S., Ye, X., Latecki, L.J., 2021. Leveraging line-point consistence to preserve structures for wide parallax image stitching, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12186–12195.

Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K.M., Trulls, E., 2021. Image matching across wide baselines: From paper to practice. International Journal of Computer Vision 129, 517–547.

Kim, H., Lee, S., 2012. Simultaneous line matching and epipolar geometry estimation based on the intersection context of coplanar line pairs. Pattern Recognition Letters 33, 1349–1363.

Kim, H., Lee, S., Lee, Y., 2014. Wide-baseline stereo matching based on the line intersection context for real-time workspace modeling. JOSA A 31, 421–435.

Li, J., Hu, Q., Ai, M., 2019. Lam: Locality affine-invariant feature matching. ISPRS Journal of Photogrammetry and Remote Sensing 154, 28–40.

Li, K., Yao, J., 2017. Line segment matching and reconstruction via exploiting coplanar cues. ISPRS Journal of Photogrammetry and Remote Sensing 125, 33–49.

Li, K., Yao, J., Lu, M., Heng, Y., Wu, T., Li, Y., 2016a. Line segment matching: a benchmark, in: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. pp. 1–9.

Li, K., Yao, J., Lu, X., Li, L., Zhang, Z., 2016b. Hierarchical line matching based on line–junction–line structure descriptor and local homography estimation. Neurocomputing 184, 207–220.

Lin, W.Y., Cheng, M.M., Zheng, S., Lu, J., Crook, N., 2013. Robust non-parametric data fitting for correspondence modeling, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 2376–2383.

Lin, W.Y., Wang, F., Cheng, M.M., Yeung, S.K., Torr, P.H., Do, M.N., Lu, J., 2017. Code: Coherence based decision boundaries for feature correspondence. IEEE transactions on pattern analysis and machine intelligence 40, 34–47.

Lin, W.Y.D., Cheng, M.M., Lu, J., Yang, H., Do, M.N., Torr, P., 2014. Bilateral functions for global motion modeling, in: European Conference on Computer Vision, Springer. pp. 341–356.

Ma, J., Jiang, X., Fan, A., Jiang, J., Yan, J., 2021. Image matching from handcrafted to deep features: A survey. International Journal of Computer Vision 129, 23–79.

Muja, M., Lowe, D.G., 2014. Scalable nearest neighbor algorithms for high dimensional data. IEEE transactions on pattern analysis and machine intelligence 36, 2227–2240.

Myronenko, A., Song, X., Carreira-Perpinán, M.A., et al., 2007. Non-rigid point set registration: Coherent point drift. Advances in neural information processing systems 19, 1009.

Nakano, G., 2021. Camera calibration using parallel line segments, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE. pp. 1505–1512.

Ok, A., Wegner, J.D., Heipke, C., Rottensteiner, F., Sörgel, U., Toprak, V., 2010. A stereo line matching technique for aerial images based on a

pair-wise relation approach. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences-ISPRS Archives 38 (2010), Nr. 1W17 38.

Ok, A.O., Wegner, J.D., Heipke, C., Rottensteiner, F., Soergel, U., Toprak, V., 2012. Matching of straight line segments from aerial stereo images of urban areas. ISPRS Journal of Photogrammetry and Remote Sensing 74, 133–152.

Qin, R., Chen, M., Huang, X., Hu, K., 2018. Disparity refinement in depth discontinuity using robustly matched straight lines for digital surface model generation. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12, 174–185.

Schaffalitzky, F., Zisserman, A., 2002. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?", in: European conference on computer vision, Springer. pp. 414–431.

Schmid, C., Zisserman, A., 1997. Automatic line matching across views, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE. pp. 666–671.

Shao, H., Svoboda, T., Van Gool, L., 2003. Zubud-zurich buildings database for image based recognition. Computer Vision Lab, Swiss Federal Institute of Technology, Switzerland, Tech. Rep 260, 6.

Shimojo, S., Silverman, G.H., Nakayama, K., 1989. Occlusion and the solution to the aperture problem for motion. Vision research 29, 619–626.

Shipitko, O., Kibalov, V., Abramov, M., 2020. Linear features observation model for autonomous vehicle localization, in: 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), IEEE. pp. 1360–1365.

Verhagen, B., Timofte, R., Van Gool, L., 2014. Scale-invariant line descriptors for wide baseline matching, in: IEEE Winter Conference on Applications of Computer Vision, IEEE. pp. 493–500.

Von Gioi, R.G., Jakubowicz, J., Morel, J.M., Randall, G., 2008. Lsd: A fast line segment detector with a false detection control. IEEE transactions on pattern analysis and machine intelligence 32, 722–732.

Wang, J., Zhu, Q., Liu, S., Wang, W., 2021. Robust line feature matching based on pair-wise geometric constraints and matching redundancy. ISPRS Journal of Photogrammetry and Remote Sensing 172, 41–58.

Wang, L., Neumann, U., You, S., 2009a. Wide-baseline image matching using line signatures, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE. pp. 1311–1318.

Wang, Z., Wu, F., Hu, Z., 2009b. Msld: A robust descriptor for line matching. Pattern Recognition 42, 941–953.

Wei, D., Zhang, Y., Liu, X., Li, C., Li, Z., 2021. Robust line segment matching across views via ranking the line-point graph. ISPRS Journal of Photogrammetry and Remote Sensing 171, 49–62.

Xue, N., Bai, S., Wang, F.D., Xia, G.S., Wu, T., Zhang, L., Torr, P.H., 2021. Learning regional attraction for line segment detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 43, 1998–2013. doi:10.1109/TPAMI.2019.2958642.

Ye, M., Haralick, R.M., Shapiro, L.G., 2003. Estimating piecewise-smooth optical flow with global matching and graduated optimization. IEEE transactions on pattern analysis and machine intelligence 25, 1625–1630.

Yu, G., Morel, J.M., 2011. Asift: An algorithm for fully affine invariant comparison. Image Processing On Line 1, 11–38.

Yu, H., Zhen, W., Yang, W., Zhang, J., Scherer, S., 2020. Monocular camera localization in prior lidar maps with 2d-3d line correspondences, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 4588–4594.

Yuille, A.L., Grzywacz, N.M., 1988. The motion coherence theory, in: ICCV.

Zhang, L., Koch, R., 2013. An efficient and robust line segment matching approach based on lbd descriptor and pairwise geometric consistency. Journal of Visual Communication and Image Representation 24, 794–805.