# A Scale, Rotation, and Affine Invariant Line Detection and Matching Algorithm for 3D Reconstruction of Infrastructure

H. Fathi[1] and I. Brilakis[2]

[1]PhD, School of Civil & Environmental Engineering, Georgia Institute of Technology, Atlanta, GA; Email: ha_fathi@gatech.edu
[2]Laing O'Rourke Lecturer of Construction Engineering, Department of Engineering, University of Cambridge, UK; Email: ib340@cam.ac.uk

## ABSTRACT

Structure from Motion (SfM) is a well-known approach for extracting the 3D geometry of a structure from 2D images. Despite being inexpensive, simple, and automatic, it has not been widely used in the construction industry mainly due to challenges in detecting and matching visual features. In built infrastructure scenes, most areas lack distinctive points due to the prevalence of poorly-textured surfaces which obstructs the successful use of SfM. In contrast, an abundant number of straight lines are visible but cannot be fully exploited because of the low performance of existing line matching algorithms. When applied to infrastructure scenes, such algorithms produce many mismatches because of inaccurate locations of line end-points, fragmented lines, lack of geometric constraints, and lack of distinctive texture in the local neighborhood. This paper presents a novel method for tackling this problem. The key innovations of the proposed method are the joint use of local and global information in the scene, and the canonical form representation of the support region of each feature. Using the scale-space theory, lines that are stable with respect to scale variations are first extracted. A dynamic pixel support region is then assigned to each line based on the Laplacian function. Each region is described by an affine-invariant vector. Finally, the Euclidean distance between vectors and global geometric constraints in the scene are applied to determine corresponding pairs. Experimental analyses have shown performance improvement compared to the state-of-the-art algorithms.

## INTRODUCTION

There is an increasing need in the AEC/FM industry to capture the 3D geometry of a structure. Recent advancements in the field of image-based 3D reconstruction have led to the introduction of the SfM techniques as a solution for such a need. These techniques are inexpensive, easy to use, safe, and automatic. They could produce 3D point clouds or line sets with cm level accuracies which satisfy the practical requirements for several tasks such as 3D visualization, archiving, and area/volume estimation (Dai et al., 2013). However, there are a number of challenges that hinder their widespread use in the industry. Accurate feature matching between views under changes in illumination, viewpoint, and background is one of the key challenges. Most of the existing matching methods work based on local point or region features which are deficient for poorly textured scenes (Wang et al., 2009-a).

Point matching has been extensively studied in the literature. Invariant local feature points and descriptors such as SIFT (Lowe, 2004) and SURF (Bay et al., 2008) have shown reliable performance in challenging scenarios (i.e., scale and viewpoint changes). However, these approaches require sufficient texture information in the scene to be used for extracting and matching feature points. In real world applications in the AEC/FM industry, it is often the case that the target scene lacks distinctive points due to the prevalence of poorly textured surfaces (e.g., smooth surface of a concrete wall). Line features are arguably the best alternative in such cases considering the fact that an abundant number of lines are visible in man-made structures that also define topological information of the existing objects (Kim and Lee, 2012). Despite this advantage, less progress has been made in line-based 3D reconstruction compared to its point-based counterpart and line matching still remains a challenging task. The state-of-the-art line matching algorithms use different concepts such as epipolar beam, color and/or gradient histogram, and intersection context to improve the matching performance. However, most of these algorithms fail in matching lines at occluding boundaries because of their assumption for rectangular pixel support region.

This paper aims to address the aforementioned problem by proposing a scale, rotation, and affine invariant line descriptor that is calculated for a dynamic pixel support region. The shape and geometry of this region is defined based on the local texture information and zero-crossings of the image Laplacian function. Canonical image representation is also used to compensate for the image distortion due to viewpoint changes. The method is tested on several infrastructure scenes and results show an average of 4% increase in recall and 5% increase in precision compared to MSLD (Wang et al., 2009-b) as the most state-of-the-art generic line matching algorithm.

## BACKGROUND

Compared to point and region matching, line matching is still a very challenging task due to several reasons: a) inaccuracy in the location of line endpoints; b) unavailability of strong disambiguating geometric constraints; and c) lack of rich textures in the local neighborhood of a line. Because of these inherent difficulties, several approaches have been proposed during the past years to achieve robust line matching. Schmid and Zisserman (2000) used the epipolar geometry for line endpoints in short baseline matching, and one parameter family of plane homographies in wide baseline matching. The limitation of this method is the need for known geometrical relations between images in advance. Aiming to remove this limitation, Bay et al. (2005) proposed a method for line matching in color images, where an initial set of line correspondences are generated using color histograms; then, a topological filter is used to iteratively increase possible matches. This method heavily relies on color rather than the texture around the line and it may fail in the case where color is not distinctive.

In another category of line matching methods, researchers have tried to construct a multi-dimensional descriptor vector for each line segment and use the vector difference for locating good matches. For example, Wang et al. (2009-a) clustered line segments into local groups according to their spatial proximity and assigned a descriptor to each group. The similarity measure of group pairs is based on the location of line end-points, orientation of the lines, and their intersection angle; this allows the method to be affine invariant. The methods, however, cannot handle general camera motions and relies on the availability of several lines in a close proximity. In another study, Wang et al. (2009-b) presented a SIFT-like descriptor called MSLD which does not need any prior knowledge for line matching. It is purely image content-based and applicable to general scenes. Although this method demonstrates superior performance compared to the other algorithms, it provides poor matching results for line segments that are located in object boundaries, when the background of the object changes in two views. There also exist some other studies that use line-point invariants (Fan et al., 2012) or intersection context of coplanar line pairs (Kim and Lee, 2012) for robust line matching; but these methods heavily rely on the existence of some predefined structures which certainly limits their applicability.

As can be inferred from the previous discussion, most of the existing algorithms fail to provide a highly distinctive descriptor for line matching under rotation, change of illumination, image blur, change of viewpoint, and partial occlusion. The primary reason is the following: these algorithms (e.g., MSLD) assign a rectangular pixel support region to every line segment with predefined dimensions disregarding the length of the line or the visual texture in the local neighborhood; therefore, for a specific line segment at object boundaries, half of the information may completely change in two different views. This makes the matching process very difficult if not impossible.

The research objective of this paper is to test whether assigning a pixel support region with dynamic shape and geometry can enhance the distinctiveness of the generated descriptor vectors for each line segment. Due to its superior performance in generic scenes, the MSLD algorithm (Wang et al., 2009-b) is considered here as the basis and the proposed concept is applied to modify/enhance this algorithm. The key research question that will be answered is: how can we define the boundaries of a pixel support region based on the local texture information in the scene and avoid error prone matches at occluding boundaries?

## METHODOLOGY

Prior to construct multi-dimensional descriptor vectors for line segments, a number of pre-processing steps need be implemented. For a pair of images or video frames, straight lines are initially detected in each view using the exiting state-of-the-art methods that work based on the scale-space theory (e.g., SILT that is proposed by Khaleghi et al., 2009). Collinear line segments are then merged to generate long lines. Since line segments are often not fully extracted and split into several smaller (more or less) collinear line fragments, the merging process is necessary to avoid mismatches because of gaps in the edge response. The output of these two primary steps is used as the input for the proposed line matching algorithm.

A very important issue that should be dealt with in the beginning is the problem of occluding object boundaries. In such boundaries, a change in viewpoint will cause inconsistency with the image in one side of the occlusion and make the descriptor vector inaccurate. A very simple solution for this problem is to separate the pixel support region into two parts, one on either side of the line segment, and generate a descriptor for each of these sides (side 1 and side 2). Since only one of the two descriptors will be on the side of the occluding boundary, the information from the other one can be used for locating robust matches. An issue that arises with this strategy is the question that which descriptors should be used to compare line segment A with B (i.e., A-side-1 with B-side-1 or B-side-2 and vice versa). The use of the epipolar geometry is proposed here to address this problem. For this purpose, line segments in each view are converted into directed lines. The two end-points of a line segment are randomly labeled as $s$ and $e$; the directed line is therefore $\overrightarrow{se}$. The side in the clockwise direction is called $side1$ and the other side is labeled as $side2$. Now, the epipolar lines corresponding to the points $s_1$ and $e_1$ in view 1 are found using the fundamental matrix. If the epipolar lines are not parallel to the line in view 2, two scenarios could happen:

- Epipolar line for $s_1$ is closer to $s_2$ and the one for $e_1$ is closer to $e_2$; in this case, side 1 in view 1 should be compared with side 1 in view 2 and side 2 in view 1 should be compared with side 2 in view 2
- Epipolar line for $s_1$ is closer to $e_2$ and the one for $e_1$ is closer to $s_2$; in this case, side 1 in view 1 should be compared with side 2 in view 2 and side 2 in view 1 should be compared with side 1 in view 2

However, if the epipolar lines are more or less parallel to the line in view 2, the

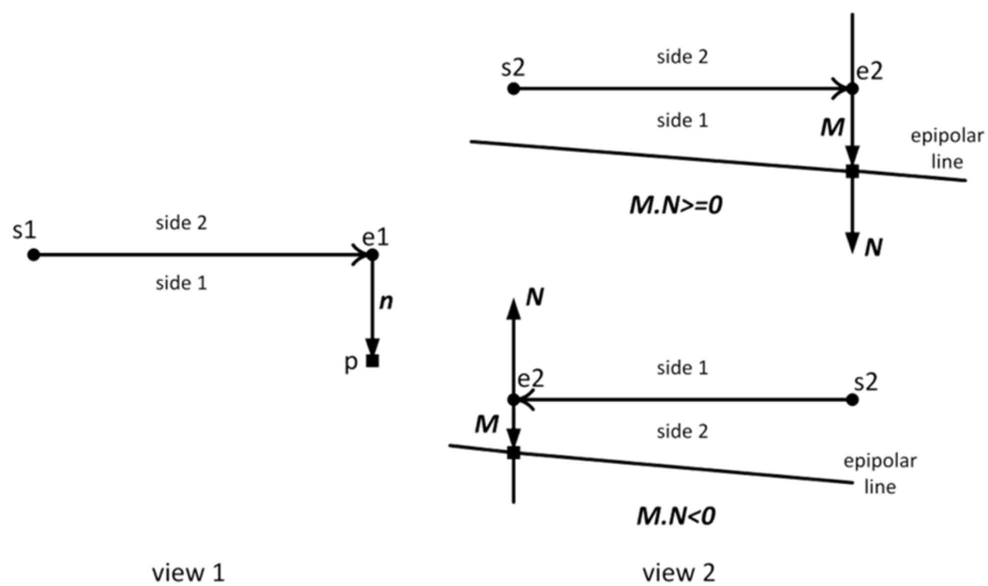


**Figure 1: Determining the side correspondence using epipolar geometry**

abovementioned approach cannot be used. In this case, the unit normal vector of the directed line in view 1 ($n$) is calculated in the clockwise direction and added to $e_1$ to achieve point $p$. The fundamental matrix between two views is used to calculate the epipolar line corresponding to $p$. As shown in Fig. 1, the dot product between the normal vector of the line in view 2 ($N$) and the vector from $e_2$ to the intersection of the epipolar line and normal vector can determine the corresponding sides. If the dot product is greater than or equal zero, the sides with same numbers should be compared to each other and vice versa.

After addressing the problem of occluding boundaries, a support region is determined for each side of a line segment. An algorithm which works based on locating zero-crossings in the Laplacian function is proposed here. It has been shown that a region enclosed by the zero-crossings of the Laplacian operator is scale-invariant and also insensitive to a wide range of viewpoint transformations (Lindeberg, 1998). In order to locate such a region, a seed point should to be given first. The proposed algorithm uses the two end-points of a line segment as well as its middle point for this purpose. Starting from each seed point, the Laplacian operator is used on the pixels along rays emanating from the seed point (Fig. 2). The zero-crossing on each ray is marked as a boundary point (the zero-crossings are typically located at places that image gradient changes rapidly). The enclosed region that is generated by connecting these boundary points is the region of interest.

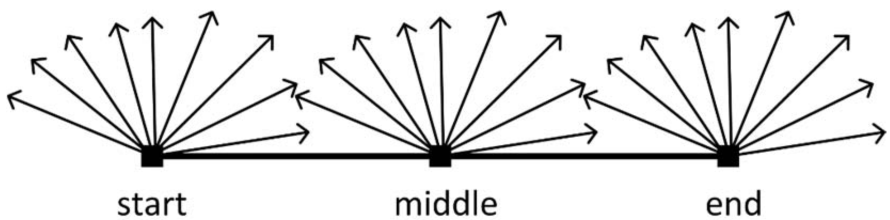


**Figure 2: Laplacian along "rays" emanating from start, middle, and end points of a line segment**

The generated support region has the following features: a) the shape and size of the region is more or less the same despite the potential inaccuracies in locating the end-points of a physical line segment in an image (Fig. 3); b) in poorly-textured areas, the region expands until edge-like points are detected; hence, the region always consists of points that are distinctive; c) the shape and size of the region is more or less the same in two views with viewpoint changes or distortions.


**Figure 3: Shape and size of the support region despite inaccuracies in locating the end-points**

Once a pixel support region is assigned to each line segment, a multi-dimensional descriptor vector needs to be constructed for each region based on the local image gradients of the enclosed pixels. The direct use of MSLD in this case would not result in desirable and distinctive descriptors because of the viewpoint changes, image rotations, and distortions. This paper proposes the canonical representation of the regions before assigning the descriptors. It is known that two directions are required for canonical representation. The direction of the line segment is used as the first direction. In order to determine the second direction, the histogram of the angle of gradients for pixels inside the region should be found first. The peak of this histogram is used as the second direction. An example is demonstrated in Fig. 4. Finally, MSLD is applied on the canonical representation of the regions and its similarity measure is used to match line segments in different views.

## DESIGN OF EXPERIMENTS

The performance of the proposed algorithm is tested on photographs or image pairs extracted from video streams. Two primary criteria are used in this evaluation: ratio of correct matches CM to total number of lines that are visible in both views TL (i.e., recall), and ratio of correct matches CM to total number of matches TM (i.e., precision). CM, TM, and TL are determined manually and via visual inspection. To achieve a realistic comparison, all thresholds and decision making parameters are set to the values that have been recommended for MSLD (Wang et al., 2009-b). The same matching criteria have been also applied in the experiments. For example, the dimension of the descriptor vectors is set to 72. The NNDR (nearest/next ratio) ratio and the global threshold are also set to 0.8 and 0.55, respectively.

In order to achieve 95% confidence and ±5% confidence interval, 400 image pairs are extracted from video streams or taken with a digital camera. This data have been collected from different environments (e.g., buildings, roof structures, bridges, façades, etc.). The image pairs have different resolutions and are captured with different cameras and lens specifications (e.g., 8, 5, and/or 3 megapixel resolution + 8, 16, and/or 25mm focal length). The data set is categorized into five groups based on the kind of transformation/change that exist between the two views (each group consists of 80 samples): rotation, scale, image blur, illumination, and viewpoint

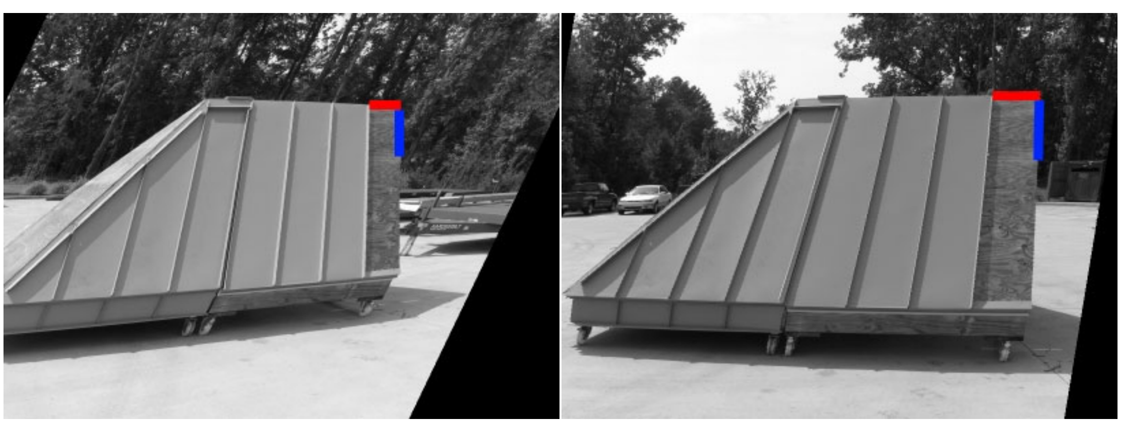

**Figure 4: Canonical representation of support regions (horizontal: line segment; vertical: direction of the peak of the histogram of angle of gradients**

change. It needs to be mentioned that other than illumination which has been changed using image editing software programs, all other cases are extracted directly from the collected data with no modification/editing. In all experiments, line segments are detected using the LSD algorithm proposed in Von Gioi et al. (2010) which is a parmeterless algorithm and does not any parameter tuning.

## IMPLEMENTATION AND RESULTS

Video streams and photographs were captured from different environments introduced in the design of experiment section with different camera and lens configurations. Image/frame pairs were then extracted from the data such that they cover a wide range of changes such as rotation, scale, image blur, illumination, and viewpoint changes. Scale-space representations of the images were first generated and then the LSD method was applied to detect line features at the local extrema. For each detected line segment, two multi-dimensional descriptor vectors were found using the original MSLD algorithm and the proposed method. Line segments were then matched by comparing those descriptors. The results of this comparison are presented in Table 1 according to two metrics: recall and precision.

As can be inferred from the numerical comparison in Table 1, the proposed algorithm performs better in terms of rotation, scale, and viewpoint changes. On the other hand, the performance of MSLD and proposed algorithm is more or less the same in illumination and blur changes. In general, the proposed method outperforms MSLD in most cases (+4% increase in recall and +5% increase in precision) which is due to assigning dynamic pixel support regions and converting the regions to the canonical form.

**Table 1: Average recall and precision for MSLD and the proposed algorithm**

| Change Scenario | Recall | | Precision | |
|---|---|---|---|---|
| | MSLD | Proposed | MSLD | Proposed |
| **Rotation** | 0.69 | 0.74 | 0.82 | 0.88 |
| **Scale** | 0.57 | 0.62 | 0.79 | 0.84 |
| **Illumination** | 0.72 | 0.74 | 0.89 | 0.90 |
| **Blur** | 0.56 | 0.56 | 0.77 | 0.80 |
| **Viewpoint** | 0.60 | 0.65 | 0.72 | 0.78 |

## SUMMARY AND CONCLUSION

It has been shown that the performance of the histogram-based algorithms in feature matching is superior to the other approaches. These algorithms typically consider a predefined rectangular shape pixel support region for constructing a descriptor vector for the target feature. In the case of line matching, this approach leads to poor performances at occluding boundaries because at least half of the information in the support region could entirely change in two different views. In order to address this limitation, this paper presented an algorithm for assigning a pixel support region that has a dynamic shape and geometry. The region boundaries are determined based on the zero-crossings of the Laplacian function. The proposed algorithm constructs two scale, rotation, and affine invariant descriptor vectors for each line segment in an image and uses the Euclidean distance to match them. The use of the epipolar geometry was proposed for selecting the vectors that need to be compared. On the other hand, canonical representation was used to compensate for image distortions due to the change of viewpoint.

A database including 400 image pairs, that were extracted from video streams or captured using a digital camera, was generated to evaluate the performance of the proposed algorithm. Five different scenarios were considered: rotation, scale, illumination, and viewpoint changes in addition to image blur. The recall and precision at each scenario were calculated using the results of the algorithm and visual inspection. The performance was also compared to MSLD as the most-state-of-the-art generic line matching algorithm. In average, the proposed algorithm outperformed MSLD by 4% in recall and 5% in precision.

## REFERENCES

Bay, H., Ess, A., Tuytelaars, T., and Gool, L. (2008) "Speeded-Up Robust Features (SURF)" *Computer Vision and Image Understanding*, 110(3), 346-359.

Bay, H., Ferrari, V., and von Gool, L. (2005) "Wide-baseline stereo matching with line segments" *In proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*.

Dai, F., Rashidi, A., Brilakis, I., and Vela, P. (2013) "Comparison of image- and time-of-flight-based technologies for 3D reconstruction of infrastructure" *Journal of Construction Engineering and Management*, 139(1), 69-79.

Fan, B., Wu, F., and Hu, Z. (2012) "Robust line matching through line–point invariants" *Pattern Recognition*, 45, 794-805.

Khaleghi, B., Baklouti, M., and Karray, F. (2009) "SILT: Scale-invariant line transform" *In proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation*.

Kim, H., and 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.

Lindeberg, T. (1998) "Feature detection with automatic scale selection" *International Journal of Computer Vision*, 30(2), 77-116.

Lowe, D. (2004) "Distinctive image features from scale-invariant key points" *International Journal of Computer Vision*, 2(60), 91-110.

Schmid, C., and Zisserman, A. (2000) "The geometry and matching of lines and curves over multiple views" *International Journal of Computer Vision*, 40(3), 199-233.

Von Gioi, R., Jakubowicz, J., Morel, J., and Randall, G. (2010) "LSD: a fast line segment detector with a false detection control" *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 32(4), 722-732.

Wang, L., Neumann, U., and You, S. (2009-a) "Wide-baseline image matching using line signatures" *In Proceedings of ICCV*, 1311–1318.

Wang, Z., Wu, F., and Hu, Z. (2009-b). "MSLD: a robust descriptor for line matching" *Pattern Recognition*, 42, 941-953.