IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 23, 2020, accepted February 19, 2020, date of publication February 28, 2020, date of current version March 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2977119

PLSD: A Perceptually Accurate Line Segment Detection Approach

QIDA YU[1], GUILI XU[1,2], YUEHUA CHENG[1], AND ZHENG H. ZHU[3], (Senior Member, IEEE)

[1]College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[2]Nondestructive Detection and Monitoring Technology for High Speed Transportation Facilities, Key Laboratory of Ministry of Industry and Information Technology, Nanjing 211100, China
[3]Department of Mechanical Engineering, York University, Toronto, ON M3J 1P3, Canada

Corresponding authors: Guili Xu (guilixu@nuaa.edu.cn) and Zheng H. Zhu (gzhu@yorku.ca)

This work was supported in part by the National Natural Science Foundation of China under Grant 61473148, and in part by the China Scholarship Council under Grant 201906830092.

ABSTRACT Most existing line segment detection methods suffer from the over-segmentation phenomenon. An improved line segment detection method is presented in this work, which can generate more and longer line segments, yet still accurately reflect the structural details of the image. Line segment grouping, line segment validation and a multiscale framework are adopted to reach this end. Specifically, smart grouping rules are introduced to locate potential homologous line segments (derived from the same boundaries). Novel merging criteria based on Helmholtz principle is then used to evaluate the meaningfulness between separate line segments and their merged ones. The improved multiscale framework facilitates line segments merging in detection and post-detection processes, yielding more high-quality line segments. Finally, the proposed method is compared with four leading methods on the famous public dataset, YorkUrban-LineSegment. Experimental results demonstrate that the method has good continuity and outperforms the leading methods in F-measure.

INDEX TERMS *A contrario* approach, grouping rules, line segment detection, line segment validation, multiscale.

I. INTRODUCTION

A line segment (LS) is a common and important geometric primitive in a digital image as most objects in man-made scenes are structured and can be outlined easily by LSs. As such, LSs are widely used as low-level cues for many vision tasks including object detection [1], shape identification [2], vanishing point estimation [3], stereo matching [4], [5], and 3D reconstruction [6]. As a fundamental technique, LS detection has been studied extensively and remains an active field within image processing research [7]. At present, LS detection algorithms can be used as an assistant tool for generating image proposals that contain salient straight lines and junctions, then feeding these image proposals to convolutional neural network (CNN) for further extracting wireframes of images [8]. This process achieves desirable results and indicates that traditional image processing techniques can work in collaboration with advancing deep learning methods. Ideally, an LS detection method should be capable of detecting LSs as accurately as human vision systems. However, this goal remains out of reach. The two main challenges of LS detection are that previous methods tend to produce several broken LSs rather than a complete LS, and that automatically tuning the thresholds for LS detection under varying conditions is difficult. Furthermore, online efficiency requirements of most real-world applications make these problems increasingly challenging.

The associate editor coordinating the review of this manuscript and approving it for publication was Ke Gu.

A. RELATED WORK

Existing methods for LS detection can be roughly classified into two categories: 1) Hough Transform; 2) Perceptual grouping.

Hough transform (HT) introduced over half a century ago, is an ingenious technique for feature extraction, which converts a global pattern detection problem in the image domain into an efficient peak detection problem in the parameter space [9]. Generally, a standard Hough-based LS detector commences by taking a binary edge map from the input image, using an edge operator, e.g., the famous Canny operator [10], and then applying HT to the extracted map for

finding all candidate lines, which are afterwards cut off into LSs according to predefined gap and length criteria. This type of methods enjoy the advantage of the global nature of HT, and thus show a good robustness to image noise and degraded configurations (e.g., occlusion), but there are some drawbacks such as complicated threshold setting problems, false detection on well-textured regions (reporting false positives) and low contrast area (reporting false negatives), heavy computation load and excessive memory consumption. To solve the aforementioned problems, every aspect of HT has been scrutinized for decades, and many improved or modified methods are put forward [11]. Early attempts mainly focus on improving the efficiency of HT [12]–[15]. Kiryati *et al.* [13] proposed a method called Probabilistic Hough Transform (PHT), where only a small subset (as low as 2%) of the edge points are selected randomly as input to HT rather than exhaustively. Within the same probabilistic framework, Matas *et al.* [14] tried to use a false detection mechanism to automate the setting of the detection thresholds. However, this method is still not fast enough in practice and may result in bad accuracy or false negatives due to the lack of scalable termination criteria. Recently, intrinsic properties of LSs are exploited into HT to enhance the detection performance [16], [17]. For example, Guerreiro and Aguiar [17] suggested imposing the connectivity between adjacent lines in the HT voting scheme via accounting for the contributions of edge points lying in increasingly larger neighborhoods and whose position and angle information agree with potential LSs. This mechanism can link several short LSs into a long one, therefore providing a more complete description of objects in real scenes. More recently, some researchers turn to analyze HT butterflies to determine the true endpoints of LSs [18]–[21], as the resulting butterfly from every single pixel of a line is capable of providing highly accurate LS parameters. In spite of all these improvements, Hough-based methods suffer a common disadvantage that their performance hinges on the quality of edge detection due to the utilization of an edge map as input and not the original image.

Perceptual grouping (PG) look at the problem from a rather different perspective. Methods in this class describe an LS as connected area where exist sufficient similar and collinear components. Hence, the key idea is to grow together components, one by one, into a so-called line support region, followed by an estimation of that area by using a validation method. Burns *et al.* [22] first proposed to detect LSs based on gradient orientation. In their method, an LS is detected as a straight image region in which inner pixels (components) share roughly the same orientation (similarities). This method is superior to HT in terms of accuracy and false detection rate. However, it needs to adjust a series of parameters empirically and manually, providing no satisfactory solution on how to choose a proper criterion for LS detection. Besides, the exhaustive search strategy used makes it unsuitable for practice. Later in [23], Von Gioi *et al.* thoroughly addressed the two issues mentioned above by introducing the Helmholtz principle and a heuristic algorithm, and then developed a linear-time Line Segment Detector (LSD) which has become a baseline in LS detection algorithms. By taking the advantage of Helmholtz principle, no parameter tuning is needed and false positives are well controlled in LSD. In fact, Helmholtz principle which was introduced by Desolneux *et al.* [24], [25], is an efficient technique for automatic computation of thresholds and has been applied to many detection problems where a detection can be modeled as an aligned structure. To allow a detection method parameter free, this principle converts the detection threshold to an easy-to-set quantity, i.e., Number of False Alarms (or NFA for short) by adopting the *a contrario* uniform random assumption. A much faster version of LSD can be found in Akinlar and Topal [26], termed as EDlines. It detects LSs from edge pixels directly, and therefore runs up to eleven times faster than LSD when processing the same image, which makes it an ideal candidate for real-time applications. However, EDlines neglects the role of points around an LS to its accurate localization, thereby leading to a poor accuracy when facing substantial noise.

All in all, compared to HT-based methods, PG approaches are computationally simple and easy to implement. Therefore, most existing LS detectors are based on local decisions, which however, also makes them prone to generate broken LSs.

B. PROPOSED METHOD

To alleviate the over-segmentation effect of PG approaches, Salaün *et al.* [27] proposed a multiscale extension of the prominent LS detector, LSD. However, their merging criterion at each scale, as well as their brute force grouping algorithm are not fully satisfactory. Hamid *et al.* [28] presented a merging method for LSD based on angular and spatial proximity, however, this technique will merge close parallel LSs into a single one, and their results are unreliable due to a lack of a validation step. In this paper, advantages from both methods are utilized and their drawbacks are excluded. A perceptually accurate line segment detection approach (PLSD) is proposed that can significantly reduce the over-segmentation phenomenon and is also robust to low contrast. Due to advantages of speed and being parameterless, the *a contrario* framework used in LSD is employed to automatically determine which LSs should be detected and merged.

Using the PLSD approach, an image pyramid is first built based on the specific size of the original image. At each scale, information from the previous scale is then explored to refine detected LSs and standard LSD is used to locate extra LSs that fail to be detected at coarser scales. Next, all LSs are grouped adaptively according to spatial and angular criteria at the finest scale. Finally, the meaningfulness of separated segments in each group is compared with their merged segment through a novel NFA score, then the passed candidate will be tested again using a density criterion.

C. NOVELTIES OF THE PROPOSED METHOD

The main contributions of this work are summarized as follows:

- Unlike most existing LS detectors which are prone to over-segmentation, a high-quality LS detection method is proposed, which tends to generate long LSs.
- An efficient grouping algorithm is presented to greedily cluster LSs as merged candidates.
- A novel merging criterion based on Helmholtz principle is put forward to determine whether the LSs in each group should be merged or not.
- More and longer LSs are detected under a novel multi-scale framework, especially for high resolution images where conventional methods do not work.

D. OUTLINE OF THE PAPER

The remainder of the paper is organized as follows. Section II revisits the principle of LSD and analyzes its fragmentation effect. Section III details the grouping rules used to propose LS merging candidates. Section IV describes the validation process, introducing a novel NFA for determining whether or not separate LSs should be merged. A multiscale framework designed to detect more and longer LSs is presented in Section V, along with a complexity analysis. The experimental results are provided in Section VI,and finally, Section VII concludes the work.

II. ANALYSIS OF THE OVER-SEGMENTATION PHENOMENON OF LSD

The proposed method is a novel extension of LSD which is discussed briefly in this section. The LSD algorithm was originally proposed in [23] and also introduced in [29] by the same author. It follows a three-stage detection framework consisting of region growing, rectangular approximation and LS validation. The process is outlined as follows.

Stage 1: In the region growing stage, the gradient magnitude map and the gradient orientation map (level-line angle (*LLA*) field) is determined using a difference-based method. These pixels are then re-sorted in a descending order according to gradient magnitude. Finally, the region growing procedure is started from the pixel with the highest gradient magnitude. The result is that the ordered pixels are recursively assigned into different line support regions.

Stage 2: In the rectangular approximation stage, every line support region is approximated with a rectangle, the endpoints and width of which uniquely position an LS. The orientation of the rectangle is defined as the angle between the positive horizontal axis and the first inertia axis of itself. The value of length and width are determined by the minimums permitting a complete covering of the line support region. Afterwards, these rectangles are regarded as LS candidates.

Stage 3: In the LS validation stage, all LS candidates are tested based on the *a contrario* statistical framework, and only those that cannot be explained by the background are retained as detections. Desolneux *et al.* [24] argued that the expectation of a searched structure under a non-structed model should be very small. Based on this theory, two critical factors are required in the *a contrario* statistical approach: a model for the background to describe the un-structured data and a model for the object to measure the structuredness. In LSD, an input image x following the background model $\mathcal{H}_0$ (null hypothesis) is a random image X, in which *LLAs* are independent random variables distributed over $[0, 2\pi]$, while the occurrence of an LS s corresponds to the situation that sufficient aligned points have appeared in a rectangle under $\mathcal{H}_0$. In the subsequent work in this paper, a pixel in s is considered to be aligned only if its *LLA* is tangent to the rectangle up to a given tolerance ζ. Specially, if $\zeta = 2p\pi$, then the probability that a pixel is aligned equals to p. Based on this, the NFA of s with $n(s)$ pixels and $k(s, x)$ aligned points can be calculated to efficiently determine whether or not s is a true detection:

$$\mathrm{NFA}(s, x) = N_L \cdot \mathbb{P}(k(s, X) \geq k(s, x)) \tag{1}$$

where N_L denotes the number of possible rectangles in X and $\mathbb{P}(k(s, X) \geq k(s, x))$ is the probability of the event that s can be observed under $\mathcal{H}_0$, which follows a binomial distribution:

$$\mathbb{P}(k(s, X) \geq k(s, x)) = \mathcal{B}(n(s), k(s, x), p) \tag{2}$$

where

$$\mathcal{B}(n(s), k(s, x), p) = \sum_{i=k(s,x)}^{n(s)} \binom{n(s)}{i} p^i (1-p)^{n(s)-i}$$

is the tail of the binomial distribution used to calculate the probability that at least $k(s, x)$ out of $n(s)$ pixels in s are aligned. In practice, if $\mathrm{NFA}(s, x) \leq \varepsilon$, with a small ε (ε is usually set to 1), the segment s is considered ε-meaningful and should therefore be detected.

While LSD is a state-of-the-art LS detector which provides complete definitions about what an LS is, how to detect and validate an LS, it yields too many fractured LSs. The over-segmentation phenomenon of LSD is illustrated in Fig. 1, in which the edges of desks are detected as multiple shorter LSs rather than a single long LS. In practical use, these short LSs are not expected. From the perception of human visual systems, short LSs are not as salient as long ones. It is therefore much easier to screen a long LS than a short one. In addition, the localization accuracy of long LSs is generally more accurate than that of short ones as long LSs have a higher tolerance to inaccurate endpoints. Breaking up a complete LS into fragments will also increase the number

(a) Input image (b) Ground Truth (c) LSD

FIGURE 1. Over-segmentation phenomenon of LSD.

of LSs detected in an image, leading to extra computational burden in subsequent applications like LS matching and 3D reconstruction. In principle, this over-segmentation weakness arises from the fact that a greedy mechanism is utilized to form the line support region. In LSD, line support regions are defined as connected pixels sharing similar *LLAs*. However, this similarity criterion is not transitive. Therefore, when image noise or lack of contrast occurs, disruptive pixels will terminate the growing of the current line support region and the remaining pixels that belong to an original LS then generate another region or several line support regions. This means that LSD is prone to fragment a long LS into several short ones. The process is detailed below.

In *Stage 1*, the gradient information of each pixel is calculated by a 2×2 mask. This procedure can be formulated as:

$$g_x(x, y) = \frac{i(x+1, y) + i(x+1, y+1) - i(x, y) - i(x, y+1)}{2} \quad (3)$$

and

$$g_y(x, y) = \frac{i(x, y+1) + i(x+1, y+1) - i(x, y) - i(x+1, y)}{2} \quad (4)$$

where $i(x, y)$ denotes the gray-level value of a pixel at (x,y).

Correspondingly, *LLA* can be computed as:

$$LLA = arctan(\frac{g_x(x, y)}{-g_y(x, y)}) \quad (5)$$

while the gradient magnitude $G(x, y)$ is obtained according to:

$$G(x, y) = \sqrt{g_x^2(x, y) + g_y^2(x, y)} \quad (6)$$

Additionally, the angle of an aligned point cluster is set to:

$$arctan(\frac{\Sigma_j sin(LLA_j)}{\Sigma_j cos(LLA_j)}) \quad (7)$$

where the index j goes through the pixels in a cluster.

Starting from a seed pixel, the line support region is enlarged gradually by adding the adjacent (eight-neighborhood) aligned points into it according to the heuristic principle. This method means that each pixel will be visited once at most in LSD and thus provides a very fast speed. However, its local nature prevents LSD from generating long LSs. As shown in Fig. 2, ideally, there should be only one LS detected in the image. However, if one of the aligned points (in blue) exhibits a mutation due to noise (labeled by yellow), then the *LLA* may disagree with its neighborhoods, thus breaking the current growth of the region. As a consequence, a complete cluster is divided into two small ones, namely, $R1$ (in green) and $R2$ (in orange), which explains the over-segmentation phenomenon of LSD.

The designer of LSD noticed this problem, and applied a GaussianBlur filter to reduce the effect of noisy pixels before further detection. However, the promotion is limited,

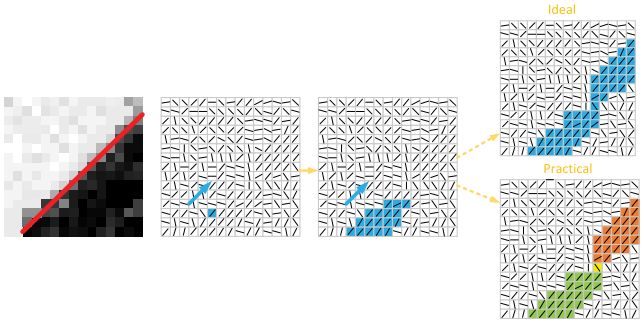


FIGURE 2. Over-segmentation phenomenon caused by disruptive pixels.

because the appearance of fractured LSs is related to not only image noise but also the information of an image. For example, the photosensitivity difference among the charge coupled device (CCD) of a camera will lead to a non-uniform alteration of gradient information in some certain pixels, where a filter does not work. A more powerful method is thus required to enhance the detection performance. As previously mentioned, long LSs are more meaningful than short ones. Based on this fact, a merging step can be added to refine the output of LSD.

In the following two sections, efficient grouping rules are determined to provide potential homologous LSs and a powerful validation method is established to determine whether the separated LS should be merged or not.

III. LINE SEGMENT GROUPING RULES

Considering that LSD detects LSs with uncertainty in the endpoint location, caused by blurring, glitters, shadows and other imperfections, merging separated segments into a single LS should satisfy the following five conditions:

1) Segments should be close enough;
2) The direction of segments should be similar enough;
3) The ratio of the overlap between them should not be more than a certain threshold;
4) The density of the merged LS should not exceed a certain threshold;
5) The gradient information of pixels in the LSs to be merged should be similar enough.

The first three rules are geometric constraints, while the last two are gradient constraints. In terms of frequency, the geometric constraints are of high frequency and are discriminating enough to distinguish homologous LSs from separated LSs. Therefore, once these constraints are guaranteed, LSs can be easily merged. However, this is not easy when noisy LSs are present. Alternatively, the gradient constraints are of low frequency. They are robust to noisy LSs and can be adopted to build a strong probabilistic model such as the Helmholtz principle, but are not as discriminating as geometric constraints. The limitations of some existing LS merging techniques [28], [30], [31] can be understood in relation to these constrains as such algorithms use part or all of the geometric constraints without an efficient validation step

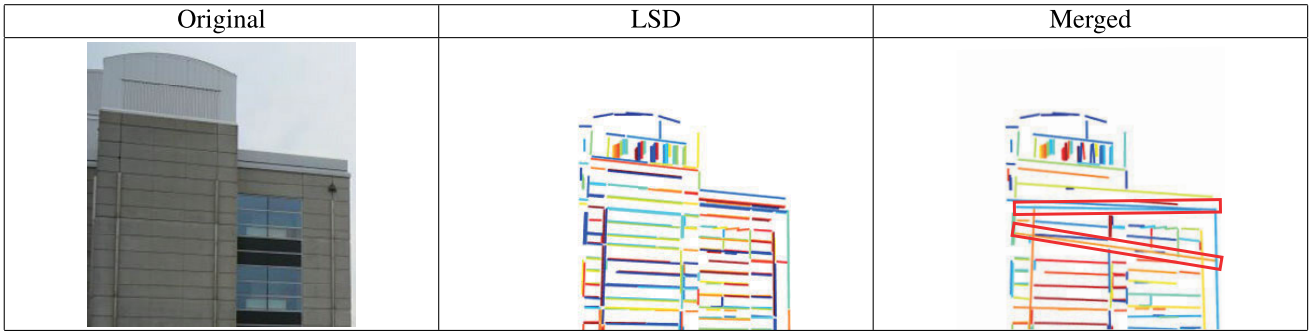


FIGURE 3. Illustration of incorrect mergings. Close parallel LSs are merged incorrectly. Incorrect mergings can be seen in the area labeled by red rectangles.

(constraint 4 and 5), leading to incorrect merging, as shown in Fig. 3.

In this paper, the first three constraints are used to group LSs and the last two are employed to validate a true detection. The grouping rules are updated from those established in [28], and are detailed as follows. The symbol $\mathcal{S}$ represents a set of LSs detected in an image by LSD and s_i is the *ith* LS in $\mathcal{S}$ ($i = 1, \ldots, \#\mathcal{S}$). Subscript $|s_i|$ denotes the length of s_i. For constraint 1 and 2, the grouping criteria with regard to angle and spatial proximity are proportional to the length of LSs to be merged. Thus, they are set adaptively based on length. For efficiency, all LSs are sorted by length beforehand, and a histogram of LS orientations is computed and employed to reduce the searching space. The grouping procedure proceeds from the longest to the shortest LS in $\mathcal{S}$, and the histogram is utilized to pick out a set of similar angle LSs for the current LS rather than exhaustively. Thus, the efficiency is increased.

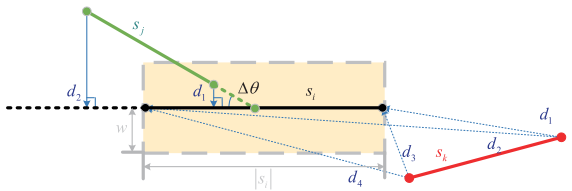


FIGURE 4. The grouping geometry for angle and spatial proximity.

As illustrated in Fig. 4, given an LS s_i and its neighbors s_j and s_k, the affect region of s_i is defined as a rectangle, the width and length of which are $2w$ and $|s_i|$, respectively. If any LS drops inside the affect region (e.g., s_j), the vertical distance from the endpoints of s_j to s_i is determined and the smaller is selected as the distance between l_i and l_j, which is denoted by d. If any LS falls outside that region (e.g., s_k), the Euclidean distance from the endpoints of s_i to those of s_k is computed and the smallest value is selected as the distance between the two segments. If d is greater than the spatial proximity criteria σ_s, the two segments should not be grouped. Otherwise, the angle proximity between them is inspected. The adaptive criteria σ_s is set as:

$$\sigma_s = \xi_s \, |s_i| \tag{8}$$

where $0 < \xi_s < 1$ is a user-specified parameter.

Next, the angle difference between two LSs is checked. The angle difference between s_i and s_j is denoted as $\Delta\theta$. If $\Delta\theta$ is smaller than the angle proximity criteria σ_a, these two segments are selected to be grouped. The adaptive criterion σ_a is modeled by a logistic function

$$\sigma_a = \left(1 - \frac{1}{1 + e^{a(\lambda + b)}}\right) \alpha \tag{9}$$

where a, b and α are user-specified parameters, and λ is a combined normalized coefficient

$$\lambda = \frac{s_j}{s_i} + \frac{d}{\sigma_s} \tag{10}$$

Finally, the overlap between s_i and s_j is examined to avoid two parallel LSs from being grouped. As illustrated in Fig. 5, p_i and p_j represent the projections of s_i and s_j onto the $y-$axis ($x-$axis) if the slope of s_i is more than (less than or equal to) one. Then, $o(p_i, p_j)$ denotes the length of overlap between p_i and p_j, the ratio of overlap $r(s_i, s_j)$ between s_i and s_j can be defined by

$$r(s_i, s_j) = \frac{o(p_i, p_j)}{|p_j|} \tag{11}$$

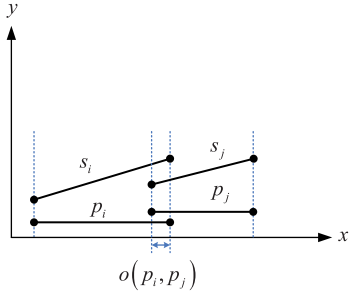
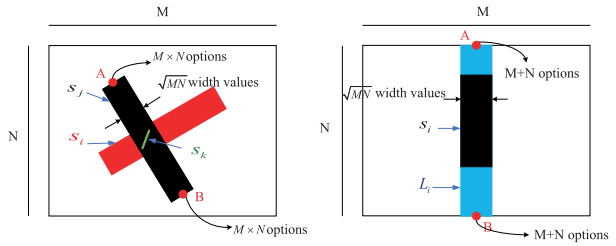


FIGURE 5. The grouping geometry for overlap.

s_i and s_j are grouped, if their overlap ratio is less than a certain threshold σ_o.

IV. LINE SEGMENT GROUPING VALIDATION

LS grouping provides many LS clusters, with each cluster considered as a merged LS hypothesis. Subsequently, a probabilistic criterion is used to accept a cluster as a true



(a) MLSD's strategy (b) Our strategy

FIGURE 6. The comparison of strategies of estimating the number of tests.

detection, using the concept detailed in [27], [32]. In [32], Cao *et al.* explored the issues of validity, stopping criterion and merging in clustering analysis based on *acontrario* statistical framework. They argue that the appropriate way to determine whether or not two disjoint clusters C_1 and C_2 should be merged into a union C (a minimal cluster containing C_1 and C_2) is to compare the joint NFA of C_1 and C_2 against the NFA of C rather than to contrast the NFA of C_1 (resp. C_2) against the NFA of C. Later, Salaün *et al.* created a multiscale extension of LSD, called multiscale line segment detector (MLSD) [27]. However, the issue encountered in [27] is that many repetitive LS configurations are counted into the number of tests, which is one of the key coefficients in NFA calculation, resulting in a wrong joint NFA for separate LSs. As shown in Fig. 6(a), s_k can be found in both s_i and s_j, while the exhaustive choice for the number of tests is to take all searched structures (rectangles in this paper) without repetitive configurations. To obtain a correct value of the number of tests for LSs in an image, a novel strategy (first a 2D line then 2D LSs) is proposed in this work which is different from those used in LSD (finding possible endpoints of an LS) and MLSD (first a 2D LS then 2D LSs). The new process is given as follows: A line is defined as an LS joining two borders of the image. Thus, there are $(N+M)\times(N+M)\times\sqrt{NM}$ different lines in a $N\times M$ image ($\sqrt{NM}$ different width values for each line is also considered). The lines are labeled baselines on which all LSs can be searched. Given a baseline L_i, an LS with support in L_i is an LS contained in L_i. Further, to guarantee an LS with support in a unique baseline, in each baseline only LSs whose width and angle are the same (or opposite) as this baseline are selected. Then, the total number of LSs with support in L_i equals to $\binom{\lceil |L_i| \rceil}{2}$. Obviously, using the first 2D lines then 2D LSs strategy can ensure that the LSs with support in L_i will not be searched on another baseline.

To calculate the joint NFA of separated LS and the NFA of their merged one, the same background and object models used in LSD are employed, and their expression can be defined according to:

Definition 1: For a merged segment s (the smallest rectangle containing the rectangles associated to s_1 and s_2) with support in L, its $\mathrm{NFA}_{\mathcal{P}}(s)$ *is defined as*

$$\mathrm{NFA}_{\mathcal{P}}(s) = \#\mathcal{L}\binom{\lceil |L| \rceil}{2}(n(s)+1)\mathcal{B}(n(s), k(s,x), p) \quad (12)$$

where $\#\mathcal{L}$ denotes the number of different possible lines in the image.

The joint NFA of separate segments can be determined in the same way as the merged segment case.

Definition 2: For two separate segments s_1 and s_2 with support in L_1 and L_2 respectively, their joint $\mathrm{NFA}_{\mathcal{P}}(s_1, s_2)$ *is defined as*

$$\#\mathcal{L}(\#\mathcal{L}-1)\prod_{i=1}^{2}\binom{\lceil |L_i| \rceil}{2}(n(s_i)+1)\mathcal{B}(n(s_i), k(s_i,x), p) \quad (13)$$

Note that $k(s,x) \neq k(s_1,x) + k(s_2,x)$, and the angle of s is different from those of s_1 and s_2, they cannot directly be added together. Instead, the p-aligned points in s must be recounted. As the value of these NFAs is considerably small, for simplicity, it is a general practice to take logarithm of NFAs. Thus, the merging function can be defined as

$$\mathcal{F} = \log\left(\frac{\mathrm{NFA}_{\mathcal{P}}(s)}{\mathrm{NFA}_{\mathcal{P}}(s_1, s_2)}\right) \quad (14)$$

If $\mathcal{F} < 0$, the separate segments are considered less meaningful than their merged one, and thus should be merged. This step corresponds to constraint 5.

For a single segment, the proposed $\mathrm{NFA}_{\mathcal{P}}(s)$ is slightly different from $\mathrm{NFA}(s,x)$ in LSD. Grompone von Gioi et al. argued that this type of difference is only technical (more details is provided in [33]).

Finally, the passed candidates are tested again according to the density of aligned points in a rectangle. For a merged LS s, its aligned point density can be expressed as

$$\mathcal{D}_s = \frac{k(s,x)}{\textit{the length of } s \times \textit{the widht of } s} \quad (15)$$

If $\mathcal{D}_s \geq 0.7$, the merged LS s is regarded as a final detection. This step corresponds to constraint 4.

V. MULTISCALE FRAMEWORK AND COMPLEXITY ANALYSIS

Considering that the image with smaller size usually generates fewer yet proportionally longer LSs, whereas the original image might produce more broken LSs, a mulitscale framework is adopted to further alleviate the fractured effect of LSD. In what follows, the scale of a picture is represented by an upper index k, which varies from 1 to K. Specifically, x^K corresponds to the original picture, while x^k is computed by the product of a coefficient $2^{2(k-K)}$ and the size of x^K. Each picture x^k is also smoothed by a GaussianBlur filter before further detection.

This framework begins from $k = 1$. First, standard LSD is employed to detect LSs on x^1, and at subsequent scale, cues from the previous scale are explored to refine detected LSs. For example, given a detected LS s_i^{k-1} with orientation $\theta(s_i^{k-1})$, let $\mathcal{R}_i^k$ denote the zoomed rectangular region of s_i^{k-1} in x^k, and $\mathcal{I}_i^k$ be the set of aligned pixels in $\mathcal{R}_i^k$

$$\mathcal{I}_i^k = \left\{q \parallel q \in \mathcal{R}_i^k \textit{ and } |\theta(q) - \theta(s_i^{k-1})|_{mod\pi} < \pi p\right\} \quad (16)$$

A set $\mathcal{C}_i^k$ of connected clusters can then be formed from $\mathcal{I}_i^k$ by an eight-neighborhood strategy. As $\mathcal{C}_i^k$ may contain a high number of tentative LSs, it is not possible to exhaustively test all fusions. Therefore, a greedy method is employed to search potential LSs to merge. Starting with the cluster of the longest length, clusters that satisfy the grouping rules defined in Section III are searched, then the merging function (Eq. (14)) is computed for each cluster. If it is negative and the density criterion is also satisfied, the current clusters are replaced by their merged version as a new cluster. This process is repeated until no more clusters can be merged.

Afterwards, only the clusters that pass the standard NFA (Eq. (1)) are retained as LSs, then LSD is used to locate extra LSs at current scale. For a detected LS s_i^{k-1}, a situation in which no derived LS can be detected on x^k may occur due to noise or lack of contrast. In this case, the original coarse LS with scale information is retained, but there are no attempts to refine it at the same location at finer scales.

Finally, all LSs detected at the finest scale are tested to be merged according to the directions in Section III and Section IV to further ensure their completeness.

Algorithm 1 PLSD

Input: The gray image x
Output: A set of more meaningful LSs $\mathcal{S}_\mathcal{P}$
REM *Step1: compute the scale pyramid consisting of K images by down-sampling*
for $k = 1$ *to* K **do**
REM *Step2: initialize LS set at current scale by up-sampling the LS set from previous scale*
if $k == 1$ **then**
$\mathcal{S}_\mathcal{P}^k \leftarrow \emptyset$
else
$\mathcal{S}_\mathcal{P}^k \leftarrow \text{Upscale}(\mathcal{S}_\mathcal{P}^{k-1})$
end
REM *Step3: refine LSs at the same locations as LSs from previous scale*
$\mathcal{S}_\mathcal{P}^k \leftarrow \text{Refine}(\mathcal{S}_\mathcal{P}^k)$
REM *Step4: add extra LSs detected by LSD at current scale*
$\mathcal{S}_\mathcal{P}^k \leftarrow \mathcal{S}_\mathcal{P}^k \cup \text{LSD}(x^k)$
end
REM *Step5: merge LSs in the whole image at scale K*
$\mathcal{S}_\mathcal{P}^K \leftarrow \text{Merge}(\mathcal{S}_\mathcal{P}^K)$
return $\mathcal{S}_\mathcal{P}^K$

The complete procedure is outlined in Algorithm 1. Note that the proposed multiscale framework is different from [27]. As Helmholtz principle is a statistical model, which may yield incorrect merging by accident when the number of pixels is not sufficient, LSs are only refined in coarser scales.

As a mulitscale framework is employed, when $k > 1$, it is difficult to calculate the whole complexity of the proposed method in a few terms. Thus, the computational cost when $k = 1$ is analyzed. In the LS detection step, the computational cost is $O(N_{pixel})$, where N_{pixel} is the number of pixels in an image. In the procedure of LS grouping, the computational complexity equals $O(N_{line}^2)$, where N_{line} stands for the number of LSs. Validating merged candidates N_{can} has an complexity of $O(N_{can})$. Hence, the entire complexity of the proposed method in one scale is $O(N_{pixel}+N_{line}^2+N_{can})$. It is linear with respect to the number of pixels. The proposed method takes approximately $0.4s$ to process a 640×480 image, indicating that the proposed method can be used for online applications.

VI. EXPERIMENTAL RESULTS

In this section, extensive and detailed experiments are performed to demonstrate the high-quality LS detection performance of the proposed method compared to existing leading methods.

A. EXPERIMENT SETUP

1) DATASET

The YorkUrban-LineSegment dataset [34] is a new benchmark for LS detection based on the YorkUrban dataset [35] and was used to test the LS methods in this paper. It is comprised of 45 indoor and 57 outdoor images with a 640×480 resolution, containing about 70k LS instances.

2) COMPARED METHODS

The proposed method is compared with four state-of-the-art LS detection methods including LSD [29], EDlines [26], LSM [28], and MLSD [27]. To ensure a fair comparison, the source codes of the compared methods are obtained online, and they are execute with default parameters. Apart from LSM which is run in MATLAB, all methods are implemented using the C++ programming language. The experiments are carried out on a laptop (Intel Core i5-3317U, 1.7GHz CPU and 4GB memory) with Unbuntu16.04.

3) EVALUATION METRICS

To reasonably and comprehensively evaluate the performance of the proposed method, the metrics employed are: the average precision (AP), the average recall (AR), F-measure, processing time (in seconds). Based on the number of correctly detected LSs (true positives (TP)), the number of incorrectly detected LSs (false positives (FP)), and the number of LSs that exist in the image but are not detected by the algorithm(false negatives (FN)), precision, recall, and F-measure can be defined as:

$$Precision = \frac{TP}{TP + FP}, \tag{17}$$

$$Recall = \frac{TP}{TP + FN}, \tag{18}$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{19}$$

A detected LS s is regarded as a TP based on the overlap ratio as follows:

TABLE 1. Experimental results of different methods on the YorkUrban-LineSegment dataset.

Method	AP	AR	F-measure	Time/s
LSD [23]	0.3280	0.3300	0.3239	11.7
EDlines [26]	0.3055	0.3373	0.3206	**4.7**
MLSD [27]	0.2976	0.3738	0.3314	65.6
LSM [28]	0.3072	0.2719	0.2885	721.3
PLSD	**0.3378**	**0.3843**	**0.3596**	41.3

a) For each ground truth s_g, a set of detections $\{s\}$ are searched which meet the following criteria:

$$d_p(s_g, s) \leq T_{dist}, \tag{20}$$

$$d_a(\theta_g, \theta_d) \leq T_{ang}, \tag{21}$$

where θ_g and θ_d are the direction of s_g and s respectively. $d_d(\cdot, \cdot)$ represents the distance from the midpoint of s_g to s in the direction orthogonal to θ_g. $d_a(\cdot, \cdot)$ returns the angular difference between two LSs.

b) The $\{s\}$ is regarded as true when its intersection over the ground truth s_g is equal or superior to T_{len}:

$$\frac{s_g \bigcap \{s\}}{s_g} \geq T_{len}. \tag{22}$$

We set $T_{dist} = 1$, $T_{ang} = \frac{\pi}{36}$ and $T_{len} = 0.75$.

4) INVOLVED PARAMETERS

The default parameter settings that are generally used in LSD are adopted along with seven new parameters which are discussed as follows: w controls the size of the affect region of an LS, which is used to classify LSs before calculating spatial proximity. The w is fixed at 20, which works well in experiments, and ξ_s is a coefficient of Eq. (8), which is used to evaluate the distance deviation of two LSs. The ξ_s is set to 0.08 empirically as this amount yields the optimum results in the experiments. Additionally, a, b and α are three coefficients of Eq. (9). They can be set to -2, -1.5, and 22.5°, respectively. Overlap tolerance σ_o is used to filter out parallel LSs and is set to 0.1, and K can be calculated automatically based on the size of original image:

$$K = min\left\{k \parallel k \in \mathbb{N} \; and \; 2^k l_{size} \geq max(w', h')\right\} \tag{23}$$

where w' and h' are the width and height of original image, respectively, l_{size} is set to 500 in this paper as over-segmentation phenomenon is not obvious for images with the size smaller than 500×500.

B. QUANTITATIVE COMPARISON

The LS detection results of different methods on the YorkUrban-LineSegment dataset are provided in Table 1. As illustrated by the results, the proposed method is superior to other methods for AP, AR and F-measure. Owing to the excellent ability of Helmholtz principle in pruning false positives, LSD, LSM, MLSD, and PLSD all display a comparable performance in terms of AP. The proposed method slightly outperforms LSD by 3% in AP because some short LSs regarded as FP in LSD are merged into long LSs in PLSD. When the length of these LSs increases, the angle difference

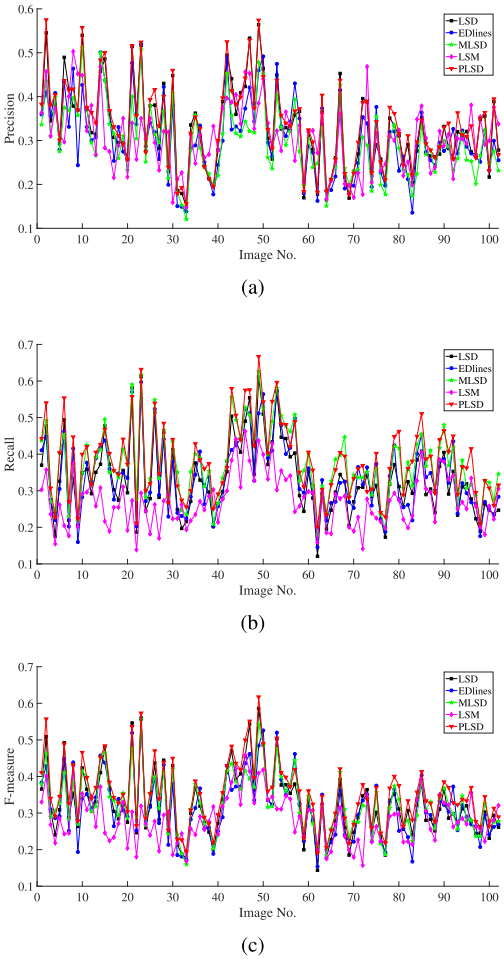


FIGURE 7. Detailed detection performance of the proposed method against other methods on the YorkUrban-LineSegment dataset, (a) Precision, (b) Recall, (c) F-measure.

TABLE 2. The number of images with the highest score for each algorithm in terms of precision, recall and F-measure on the YorkUrban-LineSegment dataset.

Index	LSD	EDlines	MLSD	LSM	PLSD
Numbers(Precision)	21	13	2	30	**36**
Numbers(Recall)	0	7	35	2	**58**
Numbers(F-measure)	4	10	18	9	**61**

TABLE 3. The average overlap ratio of correctly detected LSs on the YorkUrban-LineSegment dataset.

Method	Average overlap ratio
LSD [23]	0.9419
EDlines [26]	0.9305
MLSD [27]	0.9471
LSM [28]	0.9410
PLSD	**0.9511**

TABLE 4. The statistical results on the 28th, 48th and the whole images of YorkUrban-LineSegment dataset.

Image	Index	LSD	EDlines	LSM	MLSD	PLSD
28th	TL	21399	22356	19139	27576	24130
	N	475	503	471	490	509
	AL	45.0	44.4	40.6	56.3	47.4
48th	TL	11723	12220	13107	17275	15917
	N	440	516	570	587	583
	AL	29.2	23.7	23.0	29.4	27.3
all	TL	1793172	1970757	1707054	2444231	2079385
	N	56496	64670	60278	64076	63095
	AL	31.7	30.5	28.3	38.1	33.0

between the ground truth and detected LSs is decreased to an allowable tolerance. Both LSM and MLSD are derived from LSD and fail to increase the AP of LSD. For LSM, this is because it fails to prevent close parallel LSs from being merged, and the endpoints of the new merged LS are determined empirically by the original LSs, thus reducing its localization accuracy. More importantly, LSM lacks an accurate validation method for merging (as discussed in Section 3). MLSD fails to increase the AP of LSD because the grouping step is not adequate and neglect the uncertainty of endpoints of LSs detected by LSD. A key coefficient in the NFA calculation is also incorrect in the MLSD method. The method proposed in this paper does not encounter such problems. Due to the merging step and multiscale nature, MLSD and PLSD achieve higher AR than other methods (The AR of PLSD is 16% high compared to that of LSD). EDlines has a comparable performance to LSD as well as the fastest processing time, which is more suitable for real-time applications.

Detailed precision, recall and F-measure on the YorkUrban-LineSegment dataset with respect to the image numbers are provided in Fig. 7 and the number of images with the highest score in precision, recall and F-measure for each algorithm is listed in Table 2. Again, the proposed method displays the best performance over other methods on most images of the dataset, which implies that it is a high-quality LS detection approach.

To comprehensively assess the detection performance of the competitive LS methods, the intersection ratio T_{len} is varied from 0.55 to 0.9 with the step of 0.05. The corresponding results are provided in Fig. 8. As illustrated, the F-measure curve of the proposed method is above those of compared methods on the dataset.

Localization accuracy is essential for line-based problems. while accurate localization of LSs can ensure high accuracy for subsequent missions, localization accuracy also helps a detector distinguish close LSs. Thus, the localization accuracy of different LS detection methods are further tested by computing the average overlap ratio (AOR) of correctly detected LSs on the YorkUrban-LineSegment dataset. The results show that the proposed method achieves a slightly better performance than the compared methods in terms of AOR, as shown in Table 3. In this category, line support region based methods generally obtain a high score as more points are used to accurately locate an LS compared to other methods.

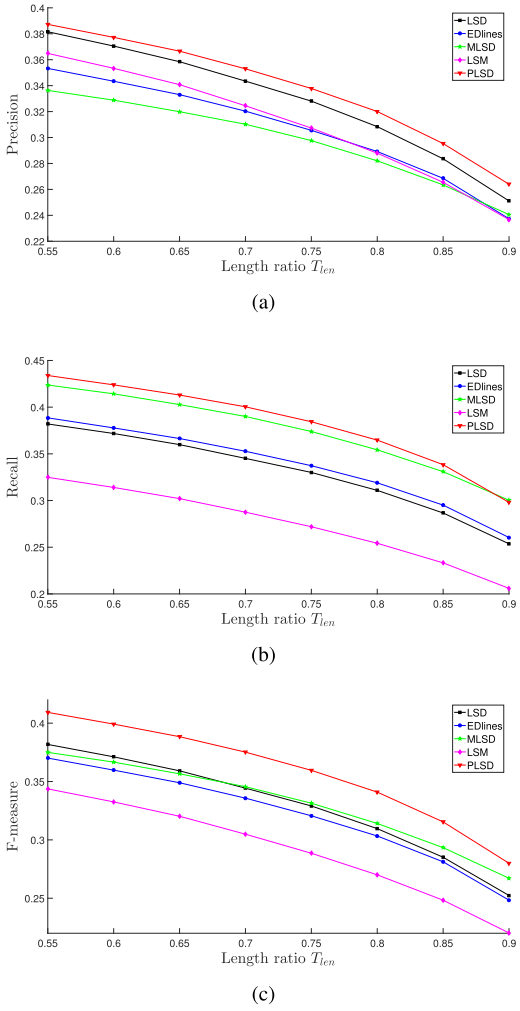
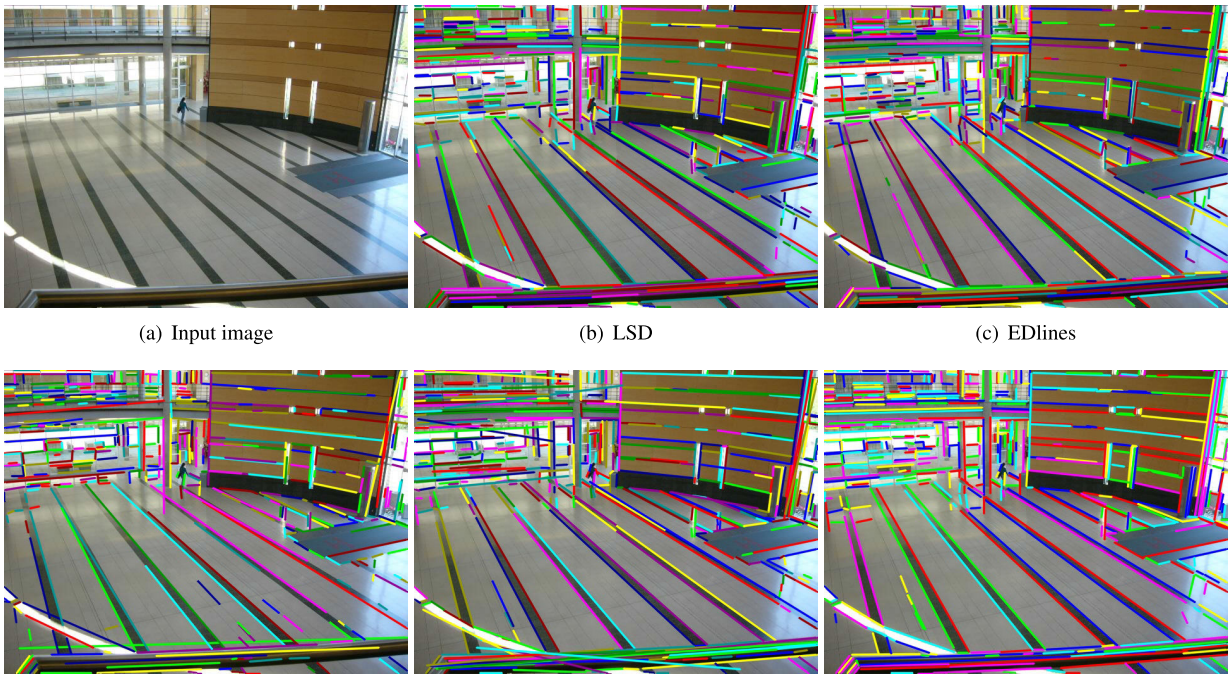


FIGURE 8. The LS detection performance of the proposed method against other methods by varying intersection length ratio T_{len} from 0.55 to 0.9 on the YorkUrban-LineSegment dataset.

C. QUALITATIVE COMPARISON

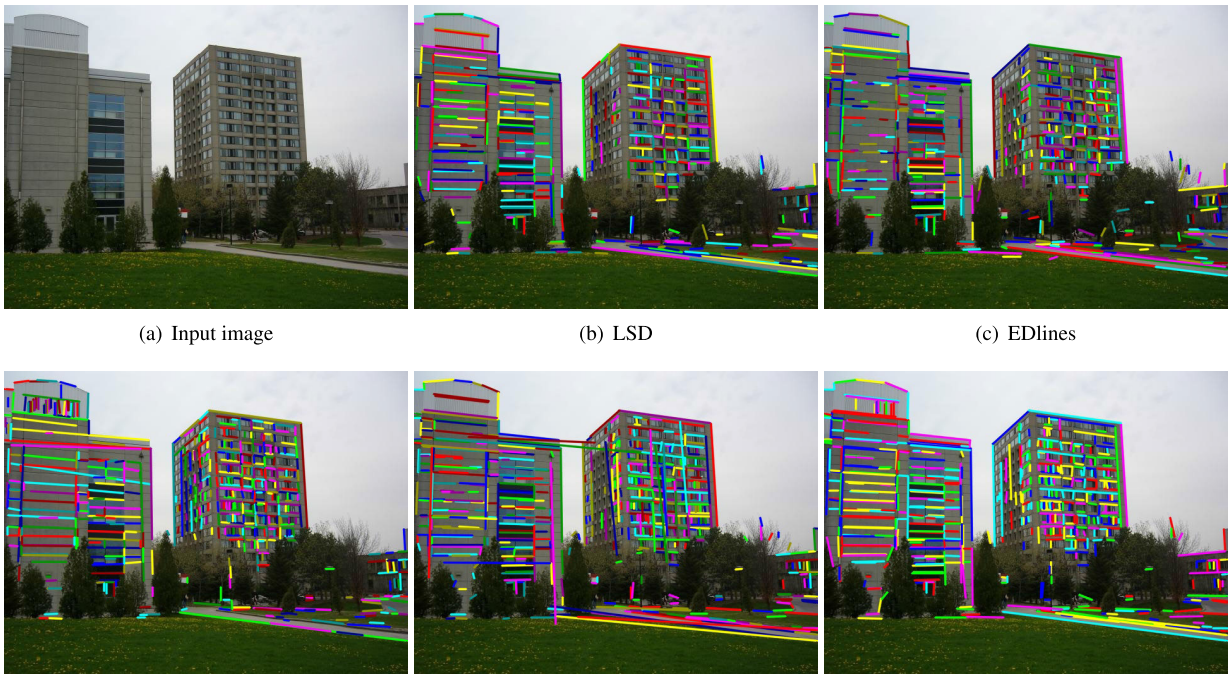
Fig. 9 and Fig. 10 show the visual results on the 28th and 48th images of the YorkUrban-LineSegment dataset using



(a) Input image (b) LSD (c) EDlines

(d) LSM (e) MLSD (f) PLSD

FIGURE 9. The visual results of different LS detection methods on the 28th image of the dataset.

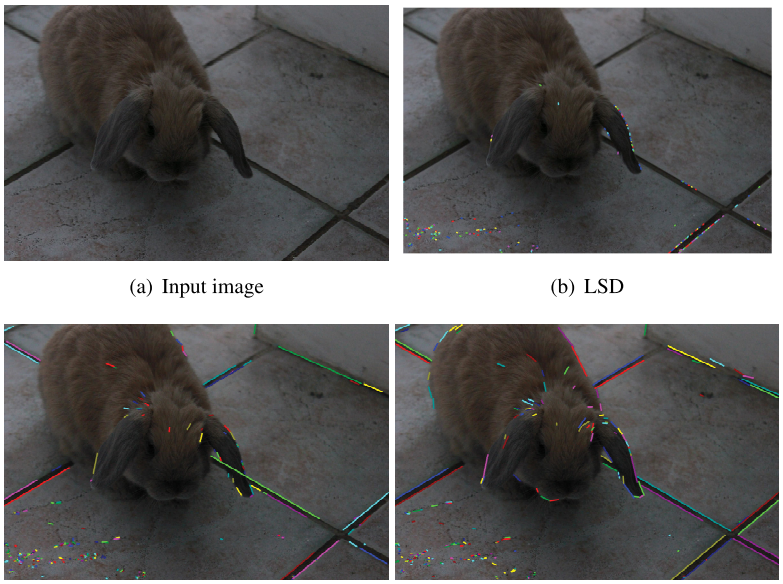


(a) Input image (b) LSD (c) EDlines

(d) LSM (e) MLSD (f) PLSD

FIGURE 10. The visual results of different LS detection methods on the 48th image of the dataset.

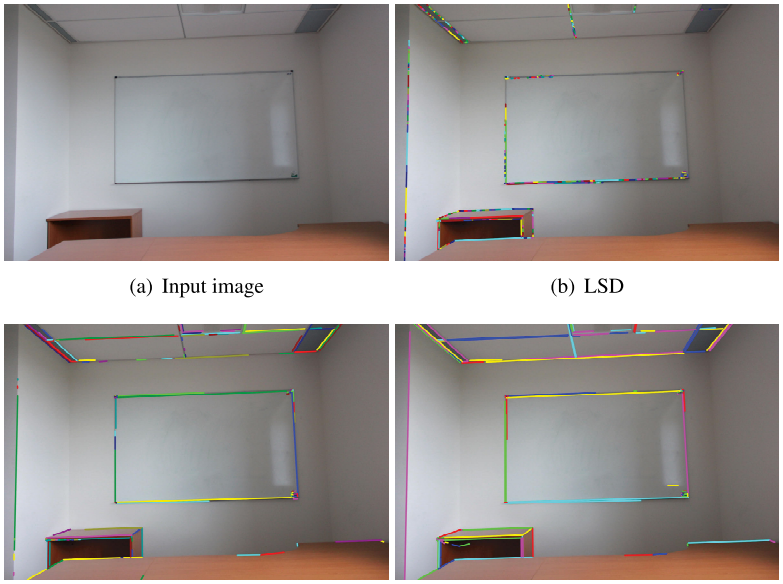
different LS detection methods. As illustrated, both LSM and MLSD generate inaccurate LSs, mainly due to their merging strategies which may fail when coping with close parallel LSs. The LSD, EDlines, and PLSD methods achieve comparable performance in terms of visual results. However, as shown in Fig. 9(c), EDlines detects the least LSs on the yellow wall, which is likely because EDlines uses a high Gaussian kernel to ensure continuity of LSs, which will



(a) Input image (b) LSD

(c) MLSD (d) PLSD

FIGURE 11. LSs detected by LSD, MLSD and PLSD on the image of a rabbit.



(a) Input image (b) LSD

(c) MLSD (d) PLSD

FIGURE 12. LSs detected by LSD, MLSD and PLSD on the image of a classroom.

reduce the contrast and degrade the details. The LSD employs down-scaling to detect longer LSs at the cost of missing some short LSs (see Fig. 9(b) and Fig. 10(b)). Unlike other methods which struggle to balance details and continuity, the proposed method is less prone to over-cutting and maintains good structural information.

To confirm the superior performance of PLSD objectively, statistical results with respect to Fig. 9, Fig. 10 and the whole dataset are provided in Table 4 and show total length (TL) in pixels, number of LSs (N), and the average length (AL) in pixels. The length of the detected LSs of PLSD is longer compared to those of LSD, EDlines and LSM,, which show PLSD has good continuity. Though MLSD is superior to PLSD in length, there are some incorrectly merged LSs in MLSD, as shown in Fig. 9(e) and Fig. 10(e). These incorrectly merged LSs are produced in MLSD due to their unreliable merging criterion and brute force grouping algorithm, which however makes their detected LSs longer.

To further demonstrate the performance of PLSD in high resolution images, two pictures (5184×3456 pixels) used in

MLSD are selected, and the results using PLSD are compared with LSD and MLSD. Note that due to the low contrast of boundaries, classical LS detectors without a multiscale framework generally work poorly on high resolution images, thus only the results of LSD and MLSD are provided in this case. As seen in Fig. 11(c), an LS detected by MLSD is incorrectly merged as it extends across the ear of the rabbit. Moreover, the proposed method provides a more complete description of objects than that of MLSD, see the floor in Fig. 11(d) and the ceiling in Fig. 12(d). It is therefore evident that the proposed method demonstrates a better performance than MLSD in high resolution images.

VII. CONCLUSION

A high-quality LS detection method is proposed in this paper which tends to generate more and longer LSs but can still accurately reflect structural information of the image. An image pyramid is first constructed based on the size of input image. At each scale, cues from the previous scale are then used to refine detected LSs, and the standard LSD algorithm is used to find extra LSs that fail to be detected at coarser scales. At the finest scale, all LSs are grouped based on spatial and angular criteria with adaptive parameters. Finally, in each group, the meaningfulness of separated segments and their merged one are compared using the *a contrario* method. The passed candidate is then tested again by a density criterion. Experimental results show that the proposed method has good continuity and achieves the best F-measure value compared to leading methods. As there is no dataset for quantitative comparison of detection of LSs in high resolution images, subsequent work will focus on creating this benchmark.

REFERENCES

[1] J. Xiao and H. Wei, "Scale-invariant contour segment context in object detection," *Image Vis. Comput.*, vol. 32, no. 12, pp. 1055–1066, Dec. 2014.
[2] H. Dong, D. K. Prasad, and I.-M. Chen, "Accurate detection of ellipses with false detection control at video rates using a gradient analysis," *Pattern Recognit.*, vol. 81, pp. 112–130, Sep. 2018.
[3] I. Suarez, E. Munoz, J. M. Buenaposada, and L. Baumela, "FSG: A statistical approach to line detection via fast segments grouping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 97–102.
[4] B. Fan, F. Wu, and Z. Hu, "Robust line matching through line–point invariants," *Pattern Recognit.*, vol. 45, no. 2, pp. 794–805, Feb. 2012.
[5] Q. Jia, X. Fan, X. Gao, M. Yu, H. Li, and Z. Luo, "Line matching based on line-points invariant and local homography," *Pattern Recognit.*, vol. 81, pp. 471–483, Sep. 2018.
[6] M. Hofer, M. Maurer, and H. Bischof, "Efficient 3D scene abstraction using line segments," *Comput. Vis. Image Understand.*, vol. 157, pp. 167–178, Apr. 2017.
[7] P. S. Rahmdel, R. Comley, D. Shi, and S. McElduff, "A review of Hough transform and line segment detection approaches," in *Proc. 10th Int. Conf. Comput. Vis. Theory Appl.*, 2015, pp. 411–418.
[8] K. Huang, Y. Wang, Z. Zhou, T. Ding, S. Gao, and Y. Ma, "Learning to parse wireframes in images of man-made environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 626–635.
[9] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, no. 2, pp. 111–122, Jan. 1981.
[10] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[11] P. Mukhopadhyay and B. B. Chaudhuri, "A survey of Hough transform," *Pattern Recognit.*, vol. 48, no. 3, pp. 993–1010, Mar. 2015.
[12] N. Guil, J. Villalba, and E. L. Zapata, "A fast Hough transform for segment detection," *IEEE Trans. Image Process.*, vol. 4, no. 11, pp. 1541–1548, Nov. 1995.
[13] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic Hough transform," *Pattern Recognit.*, vol. 24, no. 4, pp. 303–316, Jan. 1991.
[14] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic Hough transform," *Comput. Vis. Image Understand.*, vol. 78, no. 1, pp. 119–137, Apr. 2000.
[15] J. Cha, R. H. Cofer, and S. P. Kozaitis, "Extended Hough transform for linear feature detection," *Pattern Recognit.*, vol. 39, no. 6, pp. 1034–1043, Jun. 2006.
[16] K. Yang, S. S. Ge, and H. He, "Robust line detection using two-orthogonal direction image scanning," *Comput. Vis. Image Understand.*, vol. 115, no. 8, pp. 1207–1222, Aug. 2011.
[17] R. F. C. Guerreiro and P. M. Q. Aguiar, "Connectivity-enforcing Hough transform for the robust extraction of line segments," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4819–4829, Dec. 2012.
[18] S. Du, B. J. van Wyk, C. Tu, and X. Zhang, "An improved Hough transform neighborhood map for straight line segments," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 573–585, Mar. 2010.
[19] S. Du, C. Tu, B. J. van Wyk, and Z. Chen, "Collinear segment detection using HT neighborhoods," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3612–3620, Dec. 2011.
[20] Z. Xu, B.-S. Shin, and R. Klette, "A statistical method for line segment detection," *Comput. Vis. Image Understand.*, vol. 138, pp. 61–73, Sep. 2015.
[21] Z. Xu, B.-S. Shin, and R. Klette, "Closed form line-segment extraction using the Hough transform," *Pattern Recognit.*, vol. 48, no. 12, pp. 4012–4023, Dec. 2015.
[22] J. B. Burns, A. R. Hanson, and E. M. Riseman, "Extracting straight lines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 4, pp. 425–455, Jul. 1986.
[23] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.
[24] A. Desolneux, L. Moisan, and J.-M. Morel, "Meaningful alignments," *Int. J. Comput. Vis.*, vol. 40, no. 1, pp. 7–23, Oct. 2000.
[25] A. Desolneux, "When the a contrario approach becomes generative," *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 46–65, Apr. 2015.
[26] C. Akinlar and C. Topal, "EDLines: A real-time line segment detector with a false detection control," *Pattern Recognit. Lett.*, vol. 32, no. 13, pp. 1633–1642, Oct. 2011.
[27] Y. Salaün, R. Marlet, and P. Monasse, "Multiscale line segment detector for robust and accurate SfM," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 2000–2005.
[28] N. Hamid and N. Khan, "LSM: Perceptually accurate line segment merging," *J. Electron. Imag.*, vol. 25, no. 6, Dec. 2016, Art. no. 061620.
[29] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A line segment detector," *Image Process. Line*, vol. 2, pp. 35–55, Mar. 2012.
[30] J.-H. Jang and K.-S. Hong, "Fast line segment grouping method for finding globally more favorable line segments," *Pattern Recognit.*, vol. 35, no. 10, pp. 2235–2247, Oct. 2002.
[31] X. Zheng, C. Xiang, H. Lu, B. M. Chen, and T. H. Lee, "Human vision inspired multi-scale line segments merging and filtering," in *Proc. IEEE 14th Int. Conf. Control Automat. (ICCA)*, Jun. 2018, pp. 259–264.
[32] F. Cao, J. Delon, A. Desolneux, P. Musé, and F. Sur, "A unified framework for detecting groups and application to shape recognition," *J. Math. Imag. Vis.*, vol. 27, no. 2, pp. 91–119, 2007.
[33] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "On straight line segment detection," *J. Math. Imag. Vis.*, vol. 32, no. 3, pp. 313–347, Jun. 2008.
[34] N.-G. Cho, A. Yuille, and S.-W. Lee, "A novel linelet-based representation for line segment detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1195–1208, May 2018.
[35] P. Denis, J. H. Elder, and F. J. Estrada, "Efficient edge-based methods for estimating manhattan frames in urban imagery," in *Proc. 10th Eur. Conf. Comput. Vis. (ECCV)*, 2008, pp. 197–210.

QIDA YU received the bachelor's degree from the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), where he is currently pursuing the Ph.D. degree. His research interests include imaging processing, machine vision, and deep learning.

GUILI XU received the Ph.D. degree from Jiangsu University, in 2002. He is currently a Professor with the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China. He has authored over 100 publications. His current research interests include photoelectric measurement, computer vision, and intelligent systems.

YUEHUA CHENG received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 2012. She is currently an Associate Professor with the College of Automation Engineering, NUAA. Her current research interests include fault diagnosis and lifespan prediction applied on satellite attitude control systems.

ZHENG H. ZHU (Senior Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in mechanics from Shanghai Jiao Tong University, Shanghai, China, the M.A.Sc. degree in robot control from the University of Waterloo, and the Ph.D. degree in mechanical engineering from the University of Toronto, ON, Canada.

He is currently a Professor, the Tier 1 York Research Chair in space technology, and the Department Chair with the Department of Mechanical Engineering, York University, Toronto, Canada. He has authored over 230 articles. His research interests include the dynamics and control of tethered space systems and on-orbit service robot. He is a Fellow of the Engineering Institute of Canada, CSME, and ASME, an Associate Fellow of AIAA, and a Licensed Professional Engineer in Ontario.