Thick Line Segment Detection with Fast Directional Tracking

Philippe Even[1(✉)], Phuc Ngo[1], and Bertrand Kerautret[2]

[1] Université de Lorraine, LORIA (UMR 7503), Nancy, France
{philippe.even,hoai-diem-phuc.ngo}@loria.fr
[2] Université Lyon 2, LIRIS (UMR 5205), Lyon, France
bertrand.kerautret@univ-lyon2.fr

Abstract. This paper introduces a fully discrete framework for a new straight line detector in gray-level images, where line segments are enriched with a thickness parameter intended to provide a quality criterion on the extracted feature. This study is based on a previous work on interactive line detection in gray-level images. At first, a better estimation of the segment thickness and orientation is achieved through two main improvements: adaptive directional scans and control of assigned thickness. Then, these advances are exploited for a complete unsupervised detection of all the line segments in an image. The new thick line detector is left available in an online demonstration.

Keywords: Line/segment detection · Discrete objects · Digital geometry

1 Introduction

Straight lines are commonly used as visual features for many image analysis processes. In particular in man-made environments, they are a suitable alternative to points for camera orientation [7,18], 3D reconstruction [16] or also simultaneous localization and mapping [10,17].

Therefore, straight line detection is always an active research topic centered on the quest of still faster, more accurate or more robust-to-noise methods [1,2,9,14,15]. Most of the times, they rely on the extraction of an edge map based on gradient magnitude. Gradient orientation is often used to discriminate candidates and thus provide better efficiency. However, they seldom provide an exploitable measure of the output line quality, based on intrinsic properties such as sharpness, connectivity or scattering. This information could be useful to get some confidence level and help to classify these features for further exploitation. It could also be a base for uncertainty propagation within 3D interpretation tools, in order to dispose of complementary measures to reprojection errors for local accuracy evaluation.

In digital geometry, new mathematical definitions of classical geometric objects, such as lines or circles, have been developed to better fit to the discrete

nature of most of today's data to process. In particular, the notion of blurred segment [4,6] was introduced to cope with the image noise or other sources of imperfections from the real world using a thickness parameter. Efficient algorithms have already been designed to recognize these digital objects in binary images [5]. Blurred segments seem well suited to reflect the required line quality information.

The present work aims at designing a flexible tool to detect blurred segments with optimal thickness and orientation in gray-level images for as well supervised as unsupervised contexts. User-friendly solutions are sought, with ideally no parameter to set, or at least quite few values with intuitive meaning. An interactive tool was already designed for live line extractions in gray-level images [12]. But the segment thickness was initially fixed by the user and not estimated, leading to erroneous orientations of the detected lines. Here, the limitations of this first detector are solved by the introduction of two new concepts: (i) adaptive directional scans designed to better track the detected line; (ii) control of assigned thickness to bound its scattering. As a side effect, these two major evolutions also led to a noticeable improvement of the time performance of the detector. They are also put forward within a global line extraction algorithm which can be evaluated through an online demonstration at: http://ipol-geometry.loria.fr/~kerautre/ipol_demo/FBSD_IPOLDemo.

In the next section, the main theoretical notions used in this work are introduced. The new detector workflow, the adaptive directional scan, the control of assigned thickness and their integration into both supervised and unsupervised contexts are then presented in Sect. 3. Experiments led to assess the achieved performance of this new detector are described in Sect. 4. Finally, Sect. 5 gives a short conclusion followed by some open perspectives for future works.

2   Theoretical Background

2.1   Blurred Segment

This work relies on the notion of digital straight line as classically defined in the digital geometry literature [13]. Only the 2D case is considered here.

Definition 1. *A **digital straight line** $\mathcal{L}(a,b,c,\nu)$, with $(a,b,c,\nu) \in \mathbb{Z}^4$, is the set of points $P(x,y)$ of $\mathbb{Z}^2$ that satisfy: $0 \leq ax + by - c < \nu$.*

In the following, we note $\boldsymbol{V}(\mathcal{L}) = (a,b)$ the director vector of digital line $\mathcal{L}$, $w(\mathcal{L}) = \nu$ its arithmetical width, $h(\mathcal{L}) = c$ its shift to origin, and $p(\mathcal{L}) = max(|a|,|b|)$ its period (i.e. the length of its periodic pattern). When $\nu = p(\mathcal{L})$, then $\mathcal{L}$ is the narrowest 8-connected line and is called a *naive line*.

The *thickness* $\mu = \frac{\nu}{max(|a|,|b|)}$ of $\mathcal{L}(a,b,c,\nu)$ is the minimum of the vertical and horizontal distances between lines $ax + by = c$ and $ax + by = c + \nu$.

Definition 2. *A **blurred segment** $\mathcal{B}$ of assigned thickness ε is a set of points in $\mathbb{Z}^2$ that all belong to a covering digital straight line $\mathcal{L}$ of thickness $\mu = \varepsilon$. The **optimal line** of the blurred segment is the covering line with minimal thickness. The **thickness** of the blurred segment is the thickness of its optimal line.*

A linear-time algorithm to recognize a blurred segment of assigned thickness ε [6] is used in this work. It is based on an incremental growth of the convex hull of the blurred segment when adding each point P_i successively. As depicted on Fig. 1, the extension of the blurred segment $\mathcal{B}_{i-1}$ of assigned thickness ε and thickness μ_{i-1} at step $i-1$ with a new input point P_i is thus controlled by the recognition test $\mu_i < \varepsilon$.
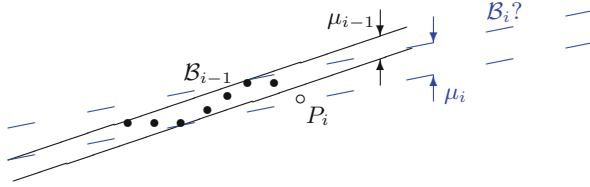

Fig. 1. A growing blurred segment $\mathcal{B}_i$: when adding the new point P_i, the blurred segment minimal thickness augments from μ_{i-1} to μ_i; if the new thickness μ_i exceeds the assigned thickness ε, then the new input point is rejected and $\mathcal{B}_i = \mathcal{B}_{i-1}$.

Associated to this primitive, the following definition of a directional scan is an important point in the proposed method.

2.2   Directional Scan

Definition 3. *A directional scan DS is an ordered partition restricted to the image domain $\mathcal{I}$ of a thick digital straight line $\mathcal{D}$, called the **scan strip**, into scans S_i, each of them being a segment of a naive line $\mathcal{N}_i$, called a **scan line**, orthogonal to $\mathcal{D}$.*

$$DS = \left\{ S_i = \mathcal{D} \cap \mathcal{N}_i \cap \mathcal{I} \left| \begin{array}{l} \boldsymbol{V}(\mathcal{N}_i) \cdot \boldsymbol{V}(\mathcal{D}) = 0 \\ \wedge\ h(\mathcal{N}_i) = h(\mathcal{N}_{i-1}) + p(\mathcal{D}) \end{array} \right. \right\} \tag{1}$$

In this definition, the clause $\boldsymbol{V}(\mathcal{N}_i) \cdot \boldsymbol{V}(\mathcal{D}) = 0$ expresses the orthogonality constraint between the scan lines $\mathcal{N}_i$ and the scan strip $\mathcal{D}$. Then the shift of the period $p(\mathcal{D})$ between successive scans guarantees that all points of the scan strip are traversed one and only one time.

The scans S_i are developed on each side of a start scan S_0, and ordered by their distance to the start line $\mathcal{N}_0$ with a positive (resp. negative) sign if they are on the left (resp. right) side of $\mathcal{N}_0$ (Fig. 2). The directional scan is iteratively parsed from the start scan to both ends. At each iteration i, the scans S_i and S_{-i} are successively processed.

A directional scan can be defined by its start scan S_0. If $A(x_A, y_A)$ and $B(x_B, y_B)$ are the end points of S_0, and if we note $\delta_x = x_B - x_A$, $\delta_y = y_B - y_A$, $c_1 = \delta_x \cdot x_A + \delta_y \cdot y_A$, $c_2 = \delta_x \cdot x_B + \delta_y \cdot y_B$ and $p_{AB} = max(|\delta_x|, |\delta_y|)$, it is then defined by the following scan strip $\mathcal{D}^{A,B}$ and scan lines $\mathcal{N}_i^{A,B}$:

$$\begin{cases} \mathcal{D}^{A,B} = \mathcal{L}(\delta_x,\ \delta_y,\ min(c1, c2),\ 1 + |c_1 - c_2|) \\ \mathcal{N}_i^{A,B} = \mathcal{L}(\delta_y,\ -\delta_x,\ \delta_y \cdot x_A - \delta_x \cdot y_A + i \cdot p_{AB},\ p_{AB}) \end{cases} \tag{2}$$
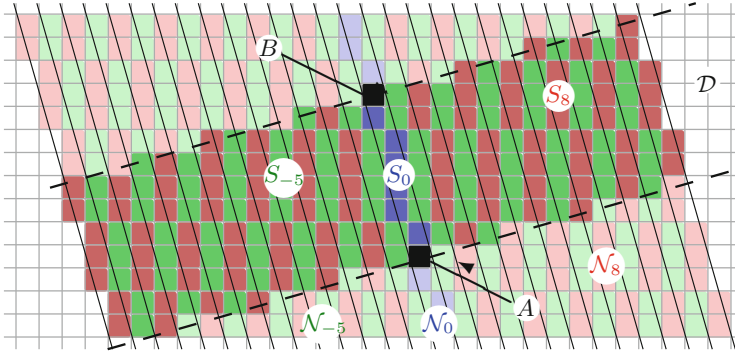
Fig. 2. A directional scan. The start scan S_0 is drawn in blue, odd scans in green, even scans in red, the bounds of scan lines $\mathcal{N}_i$ with plain lines and the bounds of scan strip $\mathcal{D}$ with dotted lines. (Color figure online)

A directional scan can also be defined by a central point $C(x_C, y_C)$, a direction $\boldsymbol{D}(X_D, Y_D)$ and a minimal thickness w. If we note $p_{\boldsymbol{D}} = max(|X_D|, |Y_D|)$, $\nu_{\boldsymbol{D}} = \lceil w \cdot p_{\boldsymbol{D}} \rceil$, $c_3 = x_C \cdot Y_D - y_C \cdot X_D - \frac{\nu_{\boldsymbol{D}}}{2}$, and $c_4 = x_C \cdot X_D + y_C \cdot Y_D - \frac{p_{\boldsymbol{D}}}{2}$, it is then defined by the following scan strip $\mathcal{D}^{C,\boldsymbol{D},w}$ and scan lines $\mathcal{N}_i^{C,\boldsymbol{D},w}$:

$$\begin{cases} \mathcal{D}^{C,\boldsymbol{D},w} = \mathcal{L}(Y_D, \ -X_D, \ c_3, \ \nu_{\boldsymbol{D}}) \\ \mathcal{N}_i^{C,\boldsymbol{D},w} = \mathcal{L}(X_D, \ Y_D, \ c_4 + i \cdot p_{\boldsymbol{D}}, \ p_{\boldsymbol{D}}) \end{cases} \tag{3}$$

3   The Detection Method

In this line detection method, only the gradient information is processed as it provides a good information on the image dynamics, and hence the presence of edges. Trials to use the intensity signal were also made through costly correlation techniques, but they were mostly successful for detecting shapes with a stable appearance such as metallic tubular objects [3]. Contrarily to most detectors, no edge map is built here, but gradient magnitude and orientation are examined in privileged directions to track edge traces. In particular, we use a Sobel operator with a 5×5 pixels mask to get high quality gradient information [11].

3.1   Previous Work

In a former paper [12], an efficient tool to detect blurred segments of fixed thickness in gray-level images was already introduced. It was based on a first rough detection in a local image area defined by the user. At that stage, the goal was to disclose the presence of a straight edge. Therefore as simple a test as the gradient maximal value was performed. In case of success, refinement steps were then run through an exploration of the image in the direction of the detected edge. In order to prevent local disturbances such as the presence of a

sharper edge nearby, all the local gradient maxima were successively tested until a correct candidate with an acceptable gradient orientation was found.

Despite of a good behavior reported, several drawbacks remained. First, the blurred segment thickness was not measured but initially set by the user according to application requirements. The produced information on edge quality was rather poor, and especially when the edge is thin, the risk to incorporate outlier points was quite high, thus producing a biased estimation of the edge orientation. Then, two refinement steps were systematically performed. On the one hand, this is useless when the first detection is successful. On the other hand, there is no guarantee that this approach is able to process larger images. The search direction relies on the support vector of the blurred segment detected at former step. Because the numerization rounding fixes a limit on this estimated orientation accuracy, more steps are inevitably required to process larger images. In the following, we present the improvements in the new detector to overcome these limitations.

3.2   Workflow of the New Detection Process

The workflow of the detection process is summarized in the following figure (Fig. 3).
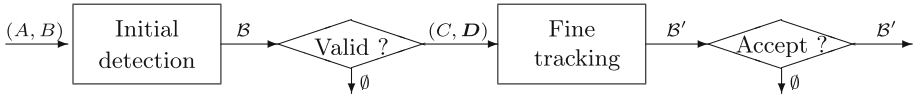

Fig. 3. The main workflow of the detection process.

The initial detection consists in building and extending a blurred segment $\mathcal{B}$ of assigned thickness ε_0, based on points with highest gradient magnitude found in each scan of a static directional scan defined by an input segment AB. The extension is stopped after five point addition failures on each side. Notice that the gradient direction is not used in this step.

Validity tests are then applied to decide of the detection pursuit. They aim at rejecting too small segments (less than 4 points) or too sparse ones (more than 50% of point addition failures) or also those with a close orientation to AB (less than $\pi/6$).

In the fine tracking step, another blurred segment $\mathcal{B}'$ is built and extended with points that correspond to local maxima of the image gradient, ranked by magnitude order, and with gradient direction close to start point gradient direction (less than $\pi/6$). At this refinement step, a *control of assigned thickness* is applied and an *adaptive directional scan* based on found position C and direction $\mathbf{D}$ is used in order to extend the segment in appropriate direction. These two notions are described in following Sects. 3.3 and 3.4.

Output segment $\mathcal{B}'$ is finally accepted based on application criteria. Final length and sparsity thresholds can be set accordingly. They are the only parameters of this local detector, together with the input assigned thickness ε_0.

3.3   Adaptive Directional Scan

The blurred segment is searched within a directional scan with position and orientation approximately drawn by the user, or blindly defined in unsupervised mode. In most cases, the detection stops where the segment escapes sideways from the scan strip (Fig. 4a). A second search is then run using another directional scan aligned on the detected segment (Fig. 4b). In the given example, an outlier added to the initial segment leads to a wrong orientation value. But even in case of a correct detection, this estimated orientation is subject to the numerization rounding, and the longer the real segment is, the higher the probability gets to fail again on an escape from the scan strip.
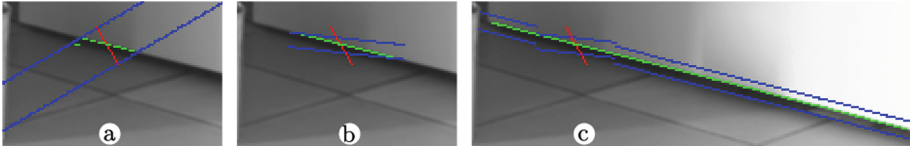

Fig. 4. Aborted detections on side escapes of static directional scans and successful detection using an adaptive directional scan. The last points added to the left of the blurred segment during initial detection (a) lead to a bad estimation of its orientation, and thus to an incomplete fine tracking with a classical directional scan (b). An adaptive directional scan instead of the static one allows to continue the segment expansion as far as necessary (c). Input selection is drawn in red color, scan strip bounds in blue and detected blurred segments in green. (Color figure online)

To overcome this issue, in the former work, an additional refinement step was run in the direction estimated from this longer segment. It was enough to completely detect most of the tested edges, but certainly not all, especially if big size images with much longer edges were processed. As a solution, this operation could be iterated as long as the blurred segment escapes from the directional scan using as any fine detection steps as necessary. But at each iteration, already tested points are processed again, thus producing a useless computational cost.

Here the proposed solution is to dynamically align the scan direction on the blurred segment all along the expansion stage. At each iteration i of the expansion, the scan strip is aligned on the direction of the blurred segment $\mathcal{B}_{i-1}$ computed at previous iteration $i-1$. More formally, an *adaptive directional scan* ADS is defined by:

$$ADS = \left\{ S_i = \mathcal{D}_i \cap \mathcal{N}_i \cap \mathcal{I} \, \middle| \, \begin{array}{l} \boldsymbol{V}(\mathcal{N}_i) \cdot \boldsymbol{V}(\mathcal{D}_0) = 0 \\ \wedge \; h(\mathcal{N}_i) = h(\mathcal{N}_{i-1}) + p(\mathcal{D}_0) \\ \wedge \; \mathcal{D}_i = \mathcal{D}^{C_{i-1}, \boldsymbol{D}_{i-1}, \mu_{i-1}}, i > \lambda \end{array} \right\} \tag{4}$$

where C_i, $\boldsymbol{D}_i$ and μ_i are respectively a position, a director vector and a thickness observed at iteration i, used to update the scan strip and lines in accordance to Eq. 3. The last clause expresses the update of the scan bounds at iteration i: C_{i-1}, $\boldsymbol{D}_{i-1}$ and μ_{i-1} are respectively the intersection of the input selection and the central line of $\mathcal{B}_{i-1}$, the director vector of the optimal line of $\mathcal{B}_{i-1}$, and the thickness of $\mathcal{B}_{i-1}$. λ is a delay which is set to 20 iterations to avoid direction instabilities when too few points are inserted. Compared to static directional scans where the scan strip remains fixed to the initial line $\mathcal{D}_0$, here the scan strip moves while scan lines remain fixed. This behavior ensures a complete detection of the blurred segment even when the orientation of $\mathcal{D}_0$ is wrongly estimated (Fig. 4c).

3.4   Control of Assigned Thickness

The assigned thickness ε to the blurred segment recognition algorithm is initially set to a large value ε_0 in order to allow the detection of thick blurred segments. Then, when no more augmentation of the blurred segment thickness is observed after τ iterations ($\mu_{i+\tau} = \mu_i$), it is set to the observed thickness augmented by a half pixel tolerance factor, in order to take into account all the possible discrete lines which digitization fits to the selected points:

$$\varepsilon = \mu_{i+\tau} + \frac{1}{2} \tag{5}$$

This strategy aims at preventing the incorporation of spurious outliers in further parts of the segment. Setting the observation distance to a constant value $\tau = 20$ seems appropriate in most experimented situations.

3.5   Supervised Blurred Segments Detection

In supervised context, the user draws an input stroke across the specific edge that he wants to extract from the image. The detection method previously described is continuously run during mouse dragging and the output blurred segment is displayed on-the-fly. Details about the supervised mode are discussed in [12].

An option, called *multi-detection* (Algorithm 1), allows the detection of all the segments crossed by the input stroke AB. In order to avoid multiple detections of the same edge, an occupancy mask, initially empty, collects the dilated points of all the blurred segments, so that these points can not be used any more.

First the positions M_j of the prominent local maxima of the gradient magnitude found under the stroke are sorted from the highest to the lowest. For each of them the main detection process is run with three modifications:

1. the initial detection takes M_j and the orthogonal direction $\boldsymbol{AB}_\perp$ to the stroke as input to build a static scan of fixed thickness $2 \cdot \varepsilon_0$, and M_j is used as start point of the blurred segment;
2. the occupancy mask is filled in with the points of the dilated blurred segments $\mathcal{B}'_j$ at the end of each successful detection (a 5×5 octagonal neighborhood region of 21 pixels is used);

Algorithm 1. MultiDetect: finds all segments crossing the selection stroke.

input : Stroke points A, B, occupancy mask $\mathcal{M}$, initial thickness ε_0
output: ListOfBS $\rightarrow$ list of detected blurred segments

ListOfBS $\leftarrow \emptyset$;
LocMax $\leftarrow$ `ComputeAndSortGradientLocalMax` (A, B);

for $i \leftarrow 0$ **to** `Size` *(LocMax)* **do**
    | BlurredSegment $\leftarrow$ detect (LocMax [i], $\boldsymbol{AB}_\perp$, 2 ε_0, $\mathcal{M}$);
    | `UpdateOccupancyMask` $(\mathcal{M}$, BlurredSegment);
    | ListOfBS $\leftarrow$ ListOfBS + BlurredSegment;
end

---

3. points marked as occupied are rejected when selecting candidates for the blurred segment extension in the fine tracking step.

3.6   Automatic Blurred Segment Detection

An unsupervised mode is also proposed to automatically detect all the straight lines in the image. A stroke that crosses the whole image, is swept in both directions, vertical then horizontal, from the center to the borders. At each position, the multi-detection algorithm is run to collect all the segments found under the stroke. Then small blurred segments are rejected in order to avoid the formation of misaligned segments when the sweeping stroke crosses an image edge near one of its ends. In such situation, any nearby disturbing gradient is likely to deviate the blurred segment direction, and its expansion is quickly stopped. The stroke sweeping step is an additional parameter for automatic detections, that could be set in relation to the final length threshold parameter.

The automatic detection is available for testing from the online demonstration and from a *GitHub* source code repository: https://github.com/evenp/FBSD.

4   Experimental Validation

In the experimental stage, the proposed approach is validated through comparisons with other recent line detectors: LSD [9], ED-Lines [1] and CannyLines [14], written in C or C++ language and without any parameter settings. Only LSD provides a thickness value based on the width of regions with same gradient direction. This information does not match the line sharpness or scattering quality addressed in this work, so that it can not be actually compared to the thickness value output by the new detector. Moreover, we did not find any data base with ground truth including line thickness. Therefore, we proceed in two steps: (i) evaluation on synthetic images of the new concepts enhancement on line orientation and thickness estimation; (ii) evaluation of more global performance of the proposed approach compared to other detectors. For all these experiments in unsupervised mode, the stroke sweeping step is set to 15 pixels.

At first, the performance of both versions of the detector (with and without the concepts) is tested on a set of 1000 synthesized images containing 10 randomly placed input segments with random thickness between 2 and 5 pixels. The initial assigned thickness ε_0 is set to 7 pixels to detect all the lines in the defined thickness range in unsupervised mode. The absolute value of the difference of each found segment to its matched input segment is measured. Results in Table 1 show that the new concepts afford improved thickness and angle measurements, better precision with a smaller amount of false detections, and that they help to find most of input segments. More results can be found in a public report: https://doi.org/10.5281/zenodo.3277091.

Table 1. Measured performance of both versions of the detector on a set of synthesized images. Old refers to the previous version [12], whereas new is the proposed detector (with adaptive directional scans and control of assigned width). S is the set of all the input segments, D the set of all the detected blurred segments.

Detector	Old	New
Detected blurred segments per image	17.06 ± 3.22	16.83 ± 3.11
Undetected input segments per image	0.152 ± 0.43	$\mathbf{0.003 \pm 0.05}$
Precision (%): $P = \#(D \cap S)/\#D$	80.46 ± 7.22	$\mathbf{83.87 \pm 6.04}$
Recall (ratio of true detection, %): $R = \#(D \cap S)/\#S$	90.23 ± 3.30	$\mathbf{91.15 \pm 2.52}$
F-measure (harmonic mean, %): $F = 2 \times P \times R/(P + R)$	84.87 ± 4.42	$\mathbf{87.23 \pm 3.59}$
Thickness difference (pixels) to matched input segment	0.70 ± 0.24	$\mathbf{0.59 \pm 0.19}$
Angle difference (degrees) to matched input segment	0.61 ± 0.66	$\mathbf{0.57 \pm 0.62}$

Next experiments aim at comparing the new approach with recent line detectors. Tests are run on the York Urban database [7] composed of 102 images with their ground truth lines. As it was set in the scope of Manhattan-world environments, only lines in the three main directions are provided. For these experiments, initial assigned thickness ε_0 is set to 3 pixels, considering that the other detectors are designed to find thin lines, and final length threshold to 10 points to suit the stroke sweeping step value. Output lines smaller than 10 pixels are discarded for all the detectors. Compared measures are execution time T, covering ratio C, detected lines amount N, cumulated length of detected lines L and mean length ratio L/N. On each image of the database and for each line detector, the execution time of 100 repetitions of a complete detection, gradient extraction included, was measured using Intel Core i5 processor; T is the mean value found per image. Then, assuming that a pixel of a ground truth line is identified if there is a detected line in its 8-neighborhood, measure C is the mean ratio of the length of ground truth line pixels identified on the total amount of ground truth line pixels. Results are given in Table 2.

The example of Fig. 5 indicates that the new detector produces many small segments which could be considered as visually non-meaningful. The other detectors eliminates them by a validation test based on Helmholtz principle [8]. Such
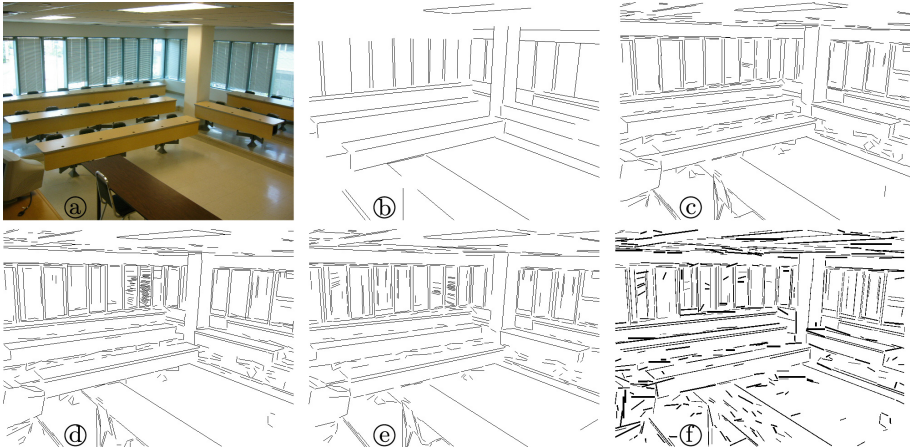
Fig. 5. Comparison of line detectors on one of the 102 ground truth images of the York Urban database: (a) input image, (b) ground truth lines, (c) LSD output, (d) ED-Lines output, (e) CannyLines output, (f) thick lines of the new detector.

Table 2. Measured performance of recent line detectors (LSD [9], ED-Lines [1] and CannyLines [14]) and of our detector on the York Urban Database [7].

Measure	T (ms)	C (%)	N	L (pixels)	L/N
LSD	63.5 ± 13.6	60.9 ± 11.2	536 ± 193	17745 ± 5337	34.6 ± 7.9
ED-Lines	55.5 ± 9.9	64.0 ± 11.2	570 ± 210	19351 ± 5669	35.8 ± 8.9
CannyLines	69.6 ± 10.5	60.5 ± 10.6	467 ± 138	17679 ± 4398	39.5 ± 10.1
Our detector	66.9 ± 15.6	$\mathbf{67.9 \pm 9.6}$	478 ± 110	19472 ± 3914	$\mathbf{41.7 \pm 7.5}$

test is not yet integrated into the new detector. But even so, the mean length of output lines is greater. Except for execution time where ED-Lines performs best, global performance of the new detector is pretty similar and competitive to the other ones. Furthermore, it provides additional information on the detected line quality through the estimated thickness.

5   Conclusion and Perspectives

This paper introduced a new straight line detector based on a local analysis of the image gradient and on the use of blurred segments to embed an estimation of the line thickness. It relies on directional scans of the input image around maximal values of the gradient magnitude, and on the integration of two new concepts: adaptive directional scans and control of assigned thickness. Comparisons to other recent line detectors show competitive global performance in terms of execution time and mean length of output lines, while experiments on

synthetic images indicate a better estimation of length and thickness measurements brought by the new concepts.

A residual weakness of the approach is the sensitivity to the initial conditions. In supervised context, the user can select a favourable area where the awaited edge is dominant. But in unsupervised context, gradient perturbations in the early stage of the line expansion, mostly due to the presence of close edges, can affect the result. In future works, we intend to provide solutions by scoring the detection result on the basis of a characterization of the local context.

References

1. Akinlar, C., Topal, C.: EDPF: a real-time parameter-free edge segment detector with a false detection control. Int. J. Pattern Recogn. Artif. Intell. **26**(01), 1255002 (2012)
2. Almazàn, E.J., Tal, R., Qian, Y., Elder, J.H.: MCMLSD: a dynamic programming approach to line segment detection. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 2031–2039 (2017)
3. Aubry, N., Kerautret, B., Even, P., Debled-Rennesson, I.: Photometric intensity profiles analysis for thick segment recognition and geometric measures. Math. Morphol. Theory Appl. **2**, 35–54 (2017)
4. Buzer, L.: A simple algorithm for digital line recognition in the general case. Pattern Recogn. **40**(6), 1675–1684 (2007)
5. Debled-Rennesson, I., Feschet, F., Rouyer-Degli, J.: Optimal blurred segments decomposition of noisy shapes in linear times. Comput. Graph. **30**, 30–36 (2006)
6. Debled-Rennesson, I., Feschet, F., Rouyer-Degli, J.: Optimal blurred segments decomposition in linear time. In: Andres, E., Damiand, G., Lienhardt, P. (eds.) DGCI 2005. LNCS, vol. 3429, pp. 371–382. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31965-8_34
7. Denis, P., Elder, J.H., Estrada, F.J.: Efficient edge-based methods for estimating Manhattan frames in urban imagery. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008. LNCS, vol. 5303, pp. 197–210. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_15
8. Desolneux, A., Moisan, L., Morel, J.M.: From Gestalt Theory to Image Analysis: A Probabilistic Approach. Interdisciplinary Applied Mathematics, vol. 34. Springer, New York (2008). https://doi.org/10.1007/978-0-387-74378-3
9. von Gioi, R.G., Jakubowicz, J., Morel, J.M., Randall, G.: LSD: a fast line segment detector with a false detection control. IEEE Trans. Pattern Anal. Mach. Intell. **32**(4), 722–732 (2010)
10. Hirose, K., Saito, H.: Fast line description for line-based SLAM. In: British Machine Vision Conference (2012)
11. Kekre, H., Gharge, S.: Image segmentation using extended edge operator for mammographic images. Int. J. Comput. Sci. Eng. **2**(4), 1086–1091 (2010)
12. Kerautret, B., Even, P.: Blurred segments in gray level images for interactive line extraction. In: Wiederhold, P., Barneva, R.P. (eds.) IWCIA 2009. LNCS, vol. 5852, pp. 176–186. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10210-3_14
13. Klette, R., Rosenfeld, A.: Digital Geometry – Geometric Methods for Digital Picture Analysis. Morgan Kaufmann, Burlington (2004)

14. Lu, X., Yao, J., Li, K., Li, L.: CannyLines: a parameter-free line segment detector. In: International Conference on Image Processing, pp. 507–511. IEEE (2015)
15. Matas, J., Galambos, C., Kittler, J.: Robust detection of lines using the progressive probabilistic Hough transform. Comput. Vis. Image Underst. **78**(1), 119–137 (2000)
16. Park, S., Lee, H., Lee, S., Yang, H.S.: Line-based single view 3D reconstruction in Manhattan world for augmented reality. In: Proceedings of the 14th ACM SIGGRAPH International Conference on Virtual Reality Continuum and Its Applications in Industry, VRCAI 2015, pp. 89–92. ACM (2015)
17. Ruifang, D., Fremont, V., Lacroix, S., Fantoni, I., Changan, L.: Line-based monocular graph SLAM. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, pp. 494–500 (2017)
18. Xu, C., Zhang, L., Cheng, L., Koch, R.: Pose estimation from line correspondences: a complete analysis and a series of solutions. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1209–1222 (2017)