Edge Detection Based on Digital Shape Elongation Measure

Faisal Alamri[1(✉)] and Joviša Žunić[2(✉)]

[1] Computer Science, University of Exeter, Exeter, UK
fa269@exeter.ac.uk
[2] Mathematical Institute, Serbian Academy of Sciences, Belgrade, Serbia
jovisa_zunic@mi.sanu.ac.rs

Abstract. In this paper, we justify the hypothesis that the methods based on the tools designed to cope with digital images can outperform the standard techniques, usually coming from differential calculus and differential geometry. Herein, we have employed the shape elongation measure, a well known shape based image analysis tool, to offer a solution to the edge detection problem. The shape elongation measure, as used in this paper, is a numerical characteristic of discrete shape, computable for all discrete point sets, including digital images. Such a measure does not involve any of the infinitesimal processes for its computation. The method proposed can be applied to any digital image directly, without the need of any pre-processing.

1 Introduction

In this paper, we propose a new method for detecting edges on digital images. Because of its importance, the edge detecting problem is well studied [1,4,6,7,11, 13]. Due to the diversity of images and applications, there is no single method that would outperform all the others in all situations. This is why there are several methods in a common use - not a single one (i.e. 'the best one'). Most of the methods are based on theoretical frameworks already established, mostly in differential calculus and geometry areas. These methods accept that there must be an inherent error when dealing with digital images. Simply, infinitesimal processes cannot be performed in discrete spaces. Thus, there are reasons to assume that in some situations, methods developed to deal with discrete data can outperform the existing ones [3,8]. We will follow this concept and design a new method without using any theoretical framework that involves infinitesimal processes (e.g. gradients, curvature, etc.). The performance of the new method will be illustrated by a comparison with the Sobel and Prewitt edge detectors [2]. Those detectors use a 3×3 neighborhood for a judgment whether a certain pixel is a good candidate to be treated as an edge point or not. It is important to notice that the new method does work (without any modification) on an

J. Žunić—Work supported by the Serbian Ministry of Education, Science and Technology.

M. Mendoza and S. Velastín (Eds.): CIARP 2017, LNCS 10657, pp. 19–27, 2018.
https://doi.org/10.1007/978-3-319-75193-1_3

arbitrary pixel neighborhood. The same 3×3 neighborhood has been chosen herein, for a fair comparison, with the edge detectors mentioned.

We provide a collection of experimental illustrations[1]. Examples are selected to be illustrative, rather than to insist on a dominance of the new method.

2 Preliminaries

In a common theoretical framework, an image is given by the image function, $I(x, y)$, defined on a rectangular array of pixels, i.e. $(x, y) \in \{1, 2, \ldots, m\} \times \{1, 2, \ldots, n\}$, while a pixel (x_0, y_0), roughly speaking, is considered to be an *edge pixel* if the image function $I(x, y)$, essentially changes the value in a certain neighborhood of (x_0, y_0). From such an interpretation, it is very natural to employ the standard differential calculus, which is very powerful and theoretically well founded mathematical tool, designed to analyze the behavior of functions in arbitrary dimensions. In this approach, the image function $I(x, y)$ is initially treated as a function defined on a continuous domain, let say $[a, b] \times [c, d]$ with $a, b, c, d \in R$, and the discontinuities of $I(x, y)$, are considered as candidates for the edge points, on the image defined by $I(x, y)$. A very popular method, Sobel edge detector and its variants (e.g. Prewitt and Canny) are based on such an approach. To determine the discontinuities of the image function $I(x, y)$, the derivatives are used. However, the function derivatives assume infinitesimal processes, what brings us to a well recognised problem: *Infinitesimal processes cannot be performed in a discrete space.* Only the approximative solutions can be given, in discrete spaces (by the way, as it has been done in Sobel method). To be more precise, the first derivative $\frac{\partial I(x,y)}{\partial x}$, defined as,

$$\frac{\partial I(x, y)}{\partial x} = \lim_{\Delta x \to 0} \frac{I(x + \Delta x, y) - I(x, y)}{\Delta x} \tag{1}$$

requires the infinitesimal process '$\Delta x \to 0'$, which cannot be performed in the digital images domain, because Δx is bounded below with the pixel size, i.e. Δx cannot be smaller than the pixel size. Thus, all the edge detecting methods based on differential calculus have an inherent loss of precision, since the derivatives can be approximated only[2] - not computed exactly. The question is: *How big the loss (made in the computation precision) is?* The experience has shown that, in many situations, the loss is not big and that the related methods are efficient. Of course, there are situations, where the approach lead to different performances of the related methods - in some situations some of the methods perform well, while others show a bed performance.

In this paper, we suggest another approach that can be, at least, compatible with the established methods. Our hypothesis is that when dealing with a discrete space, some of the methods developed to work in discrete spaces may have better performances than the existing methods do. If we go back to the

[1] Please always enlarge the images appearing, in order to provide a better viewing.

[2] Common approximation in (1) is: $\frac{\partial I(x,y)}{\partial x} \approx I(x + 1, y) - I(x, y)$, assuming $\Delta x = 1$.

initial intuition, which a good candidate for an edge point should be a point (x_0, y_0) where the image function $I(x, y)$ essentially changes its value (i.e. in a continuous space the point (x_0, y_0) is a discontinuity of $I(x, y)$), then a good candidate to indicate such a discontinuity can be *the shape elongation measure*, applied to a neighborhood of the pixel considered. Notice that there are many other shape descriptors/measures intensively studied in the literature and in a frequent use in image analysis tasks [5,9,10,12].

The rest of the paper includes: The basic definitions and statements about the elongation measure (Subsect. 2.1); The new edge detecting method (Sect. 3); Some experimental results (Sect. 4); Concluding remarks (Sect. 5).

2.1 Shape Elongation Measure

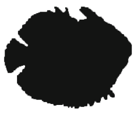
We deal with the digital images defined on a domain $\mathcal{D} = \{1, \ldots, m\} \times \{1, \ldots, n\}$, by the image function $I(x, y)$. A point $(x_0, y_0) \in \mathcal{D}$, is referred to as a pixel, while $I(x_0, y_0)$ represents the the image intensity, at the pixel (x_0, y_0).

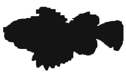
The centroid (x_s, y_S) of S is $(x_S, y_S) = \left(\frac{\sum_{(x,y)\in S} x}{\sum_{(x,y)\in S} 1}, \frac{\sum_{(x,y)\in S} y}{\sum_{(x,y)\in S} 1} \right)$ while $\overline{m}_{p,q}(S) = \sum_{(x,y)\in S} (x - x_S)^p \cdot (y - y_S)^q$, are the centralized moments.

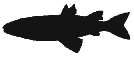
The shape orientation of a discrete point set S, is defined by the line that minimizes the total sum of the square distances of the points of S to a certain line. Such a line passes through the centroid of S. The line, which maximizes the sum of the squared distances of all the points of S and passes through the centroid of S, is orthogonal to the axis of the least second moment of inertia. Both these extreme quantities, minimum and maximum, are straightforward to compute and their ratio defines the shape elongation measure. The elongation of S is denoted as $\mathcal{E}(S)$, and computed as follows:

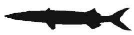
$$\mathcal{E}(S) = \frac{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) + \sqrt{(\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2 + 4\overline{m}_{1,1}(S)^2}}{\overline{m}_{2,0}(S) + \overline{m}_{0,2}(S) - \sqrt{(\overline{m}_{2,0}(S) - \overline{m}_{0,2}(S))^2 + 4\overline{m}_{1,1}(S)^2}} \qquad (2)$$

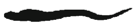
The measure $\mathcal{E}(S)$ is invariant to translation, rotation and scaling transformations, and $\mathcal{E}(S) \in [1, \infty)$. Several shape examples are below. The $\mathcal{E}(S)$ values (given below the shapes), are in accordance with our expectation.


$\mathcal{E}(S) = 1.57$


$\mathcal{E}(S) = 5.08$


$\mathcal{E}(S) = 13.32$


$\mathcal{E}(S) = 52.40$


$\mathcal{E}(S) = 129.57$

3 New Method for Detecting Edges

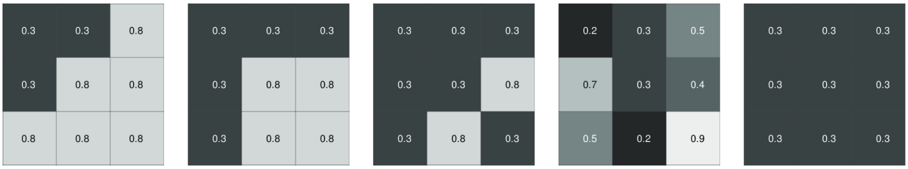
As it has been seen from (2), $\mathcal{E}(S)$ can be computed for any discrete point set S. Our hypothesis is that the $\mathcal{E}(S)$ value, calculated in a certain neighborhood of a pixel, from a grey-level image, can be used as a good indicator whether this pixel belongs to an edge or not. We will use the 3 × 3 pixel neighborhood, the same as the Sobel method does. Since $\mathcal{E}(S)$ definition allows multiple points, its extension to the grey-level images is straightforward. So, for a given image function, $I(x, y)$, we have the following formulas

$$(x_P, y_P) \;=\; \left(\frac{\sum_{(x,y)\in P} x \cdot I(x,y)}{\sum_{(x,y)\in P} I(x,y)}, \; \frac{\sum_{(x,y)\in P} y \cdot I(x,y)}{\sum_{(x,y)\in P} I(x,y)} \right) \tag{3}$$

$$\overline{m}_{p,q}(P) \;=\; \sum_{(x,y)\in P} (x - x_P)^p \cdot (y - y_P)^q \cdot I(x,y), \;\; \text{and}\, p, q \in \{0, 1, 2 \ldots\}. \tag{4}$$

$$\mathcal{E}(P) \;=\; \frac{\overline{m}_{2,0}(P) + \overline{m}_{0,2}(P) + \sqrt{(\overline{m}_{2,0}(P) - \overline{m}_{0,2}(P))^2 + 4\overline{m}_{1,1}(P)^2}}{\overline{m}_{2,0}(P) + \overline{m}_{0,2}(P) - \sqrt{(\overline{m}_{2,0}(P) - \overline{m}_{0,2}(P))^2 + 4\overline{m}_{1,1}(P)^2}} \tag{5}$$

for the centroid, centralized moments, and elongation of a grey-level image P, respectively. Notice that we have used the same notations for both black-white (i.e. $\mathcal{E}(S)$) and grey-level (i.e. $\mathcal{E}(P)$) images. This is because there is no conceptual difference between these two situations. Examples in Fig. 1 illustrate how $\mathcal{E}(P)$ acts on grey-level images. The obtained values suggest that the central pixel, in Fig. 1(a) is a good candidate to be an edge pixel, since its 3 × 3 neighborhood has the largest $\mathcal{E}(P) = 1.46$ value, comparing to all situations displayed in Fig. 1(a)–(e). The computed value $\mathcal{E}(P) = 1.00$, for the central pixel in Fig. 1(e), suggests that this pixel is not a candidate to be an edge pixel. This supports our hypothesis that $\mathcal{E}(P)$ can be efficiently employed in an edge detection task.



(a): 1.46 (b): 1.25 (c): 1.06 (d): 1.26 (e): 1.00

Fig. 1. $\mathcal{E}(P)$ values are displayed below the 3 × 3 figures. Normalized pixel intensities $I(x, y) \leq 1$ are inscribed in the corresponded pixel.

In order to present the core idea of the new method, we use the Matlab implementation of Sobel method. The only change made is that the image produced by using the gradient operator, is replaced with the image produced by using the elongation $\mathcal{E}(P)$ operator, both applied to 3 × 3 pixel neighborhood. Thus, basically both implementations have the following common steps: **Step-1:** Input (original) image; **Step-2:** Image produced by applying the related operator (i.e. the elongation operator, $\mathcal{E}(P)$, for the new method, and the gradient operator for Sobel method); **Step-3:** Thresholding applied to the image produced in step 2.; **Step-4:** Computation of the output image.

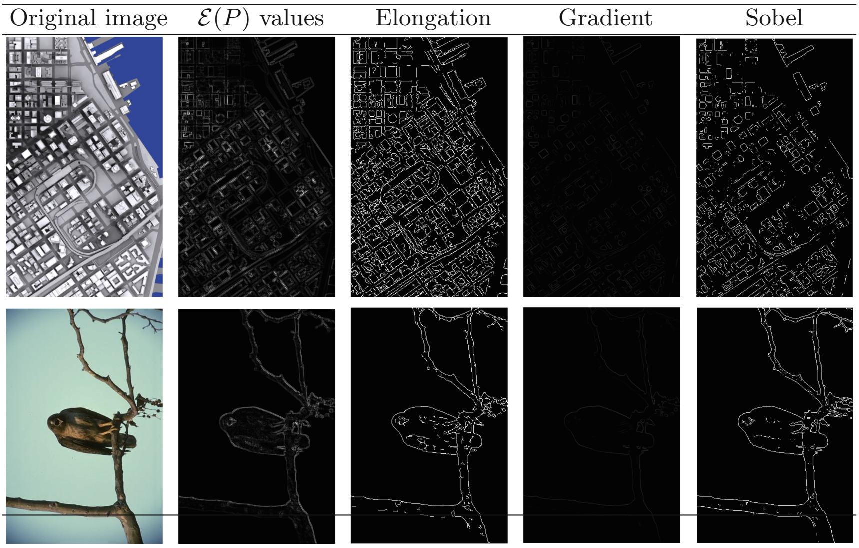


Fig. 2. Original images, images produced by the $\mathcal{E}(P)$ values and the output images, by the new method, are in the 1st, 2nd, and 3rd column, respectively. Images produced by the gradient operator and the images produced by Sobel method, are in the 4th and 5th column, respectively.

Once again, we have used Matlab code, and used $\mathcal{E}(P)$), in step 2. The threshold level (in step 3.) was not changed and adapted to the new method for a better performance achieved. These steps are illustrated in Fig. 2. The images obtained after applied elongation operator and the output images produced by the new method are in the 2nd and 3rd column, respectively. The images produced by the Sobel operator and the output images produced by Sobel method are in the 4th and 5th column, respectively. The images produced by $\mathcal{E}(P)$ and by the Sobel operator have a different structure. This causes different output images produced by these two methods.

4 Experimental Illustrations

In this section, we provide a number of experiments to illustrate the performance of the new method and compare it with the Sobel and Prewitt method.

First Experiment: Black-white images. We start with the black-white images, in Fig. 3. 1st and 4th images are original ones. The edges detected by the new method are in the 2nd and 5th image, while Sobel method produced images are 3rd and 6th ones. The new method performs better in the case of the camel image. In the case of the stick image the new method also performs better in detecting the long straight sections, while Sobel method detects a small spot in the middle of the shape.

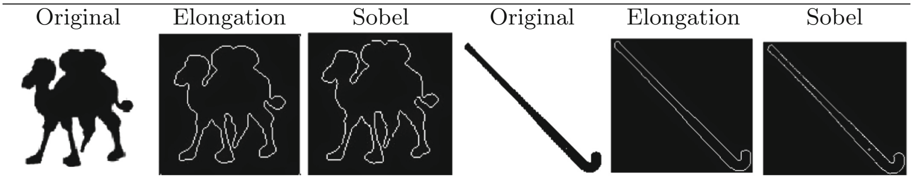


Fig. 3. 1st and 4th image: original images; 2nd and 5th image: edges detected by the new method; 3rd and 6th image: edges detected by the Sobel method.

Second Experiment: Images with the provided ground-truth images. Figure 4 includes the original images (column on the left) and the corresponding ground truth images (the column on the right). Output images produced by the new method and by Sobel method are in the 2nd and 3rd columns, respectively. At this moment, we are not going to judge which method performs better, even though the ground-truth images are provided. However, it is clear that the new method produces essentially different outputs (see the 2nd row in Fig. 4), which is compatible with Sobel method, and has a potential for applications.

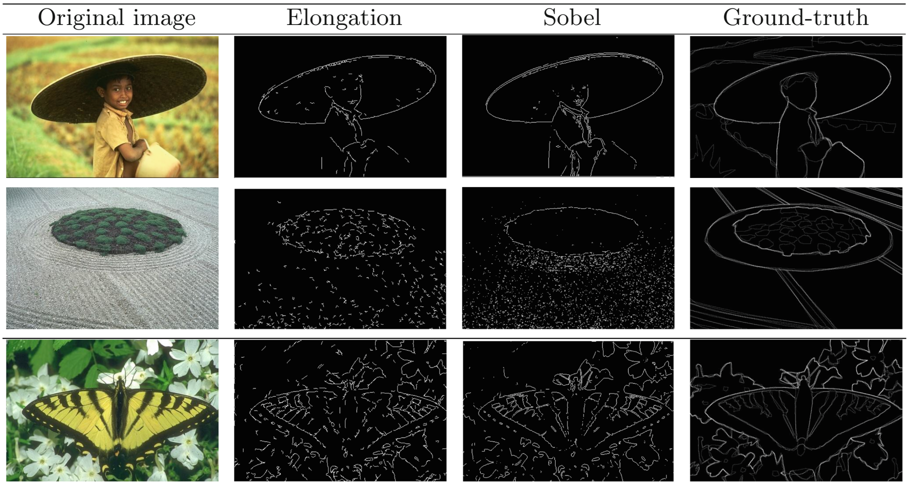


Fig. 4. Original images and output images, produced by the new method and by Sobel method, and ground-truth images are in the 1st, 2nd, 3rd, and 4th column, respectively.

Third Experiment: Images without clearly enhanced edges. In this experiment we have used art painting/drawings images[3]. As such, they do not have enhanced edges. The elongation based edge detector, Prewitt, and Sobel edge detectors, have been applied. The original images and the results produced

[3] The authors thank Radoslav Skeledžić - Skela, a Serbian painter, who has provided these images.

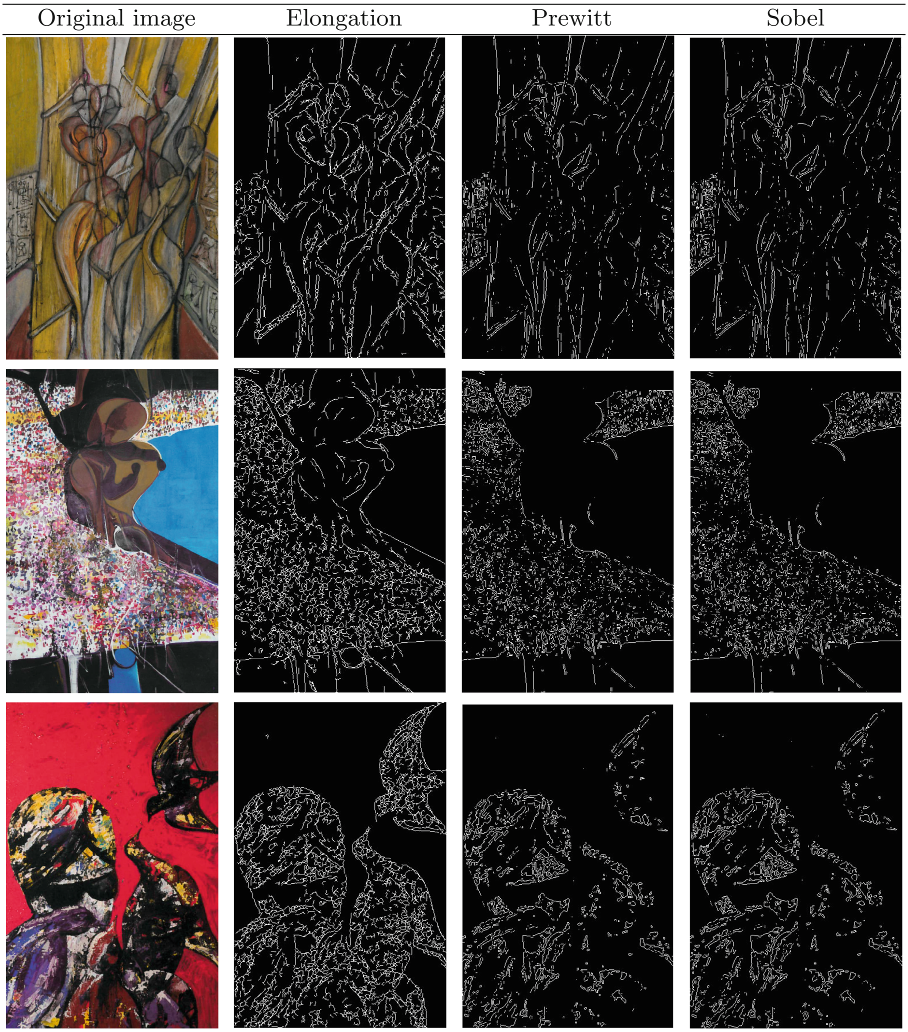


Fig. 5. Art painting images are displayed in the left column. Edges detected by the new method, Prewitt and Sobel methods are in the 2nd, 3rd, and 4th column, respectively.

by the three different edge detectors are displayed in Fig. 5. The results of Prewitt and Sobel edge detectors are slightly similar. The new method is based on a different concept, and this is why its results essentially differ from the results obtained by Prewitt and Sobel edge detectors. Without a judgment which of these three edge detectors performs the best, it could be said that the new edge detector is compatible with the existing ones.

5 Concluding Remarks

We have proposed a new method for detecting edges on digital images. The method uses shape elongation measure and does not use any mathematical tools based on infinitesimal processes. Such processes involve an inherent error, whose influence varies and strongly depends on the input image. The new measure conceptually differs from popular edge detectors. Thus, it is expected to produce very different results. Our aim was to present the core idea of the new method. This is why we have followed Matlab implementation of Sobel method. We have replaced the intermediate image produced by the gradient/Sobel operator, with the image produced by the elongation operator, and have used 3×3 pixel neighborhood (as Sobel method does). Alos, the necessary threshold parameter has been computed by Matlab, directly. It could come out that on this way we have disadvantaged the new method. At this moment, our goal was to show that the new method has a good potential to be used, together with the other edge detectors. We have provided a spectrum of small, but different in their nature, experiments. It is difficult to judge which method dominates the others. However, it is clear that the methods behavior differs from situation to situation. This implies that a use of one of the methods, in a particular situation, does not exclude a use of the other methods in some different situations.

We have not used the available common methods to upgrade our method, again, in order to point out the core idea. Some additional upgrade possibilities are already visible. First of all, a variable pixel neighborhoods and adaptive thresholding. This would be investigated in our further work.

References

1. Förstner, W.: A framework for low level feature extraction. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 801, pp. 383–394. Springer, Heidelberg (1994). https://doi.org/10.1007/BFb0028370
2. Gonzalez, R., Woods, R.: Digital Image Processing. Addison Wesley, Reading (1992)
3. Hahn, J., Tai, X.C., Borok, S., Bruckstein, A.M.: Orientation-matching minimization for image denoising. Int. J. Comput. Vis. **92**, 308–324 (2011)
4. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of Fourth Alvey Vision Conference, pp. 147–151 (1988)
5. Lachaud, J.-O., Thibert, B.: Properties of Gauss digitized shapes and digital surface integration. J. Math. Imaging Vis. **54**, 162–180 (2016)
6. Köthe, U.: Edge and junction detection with an improved structure tensor. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 25–32. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-45243-0_4
7. Lyvers, E.P., Mitchell, O.R.: Precision edge contrast and orientation estimation. IEEE Trans. Pattern Anal. Mach. Intell. **10**, 927–937 (1988)
8. Lukić, T., Žunić, J.: A non-gradient-based energy minimization approach to the image denoising problem. Inverse Prob. **30**, 095007 (2014)
9. Rosin, P.L., Žunić, J.: Measuring squareness and orientation of shapes. J. Math. Imaging Vis. **39**, 13–27 (2011)

10. Roussillon, T., Sivignon, I., Tougne, L.: Measure of circularity for parts of digital boundaries and its fast computation. Pattern Recogn. **43**, 37–46 (2010)
11. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. Int. J. Comput. Vis. **37**, 151–172 (2000)
12. Stojmenović, M., Nayak, A., Žunić, J.: Measuring linearity of planar point sets. Pattern Recogn. **41**, 2503–20511 (2008)
13. Torre, V., Poggio, T.: On edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 147–163 (1984)