Canny Edge Detection Algorithm Modification

Wojciech Mokrzycki and Marek Samko

Faculty of Mathematics and Informatics
University of Warmia and Mazury, Olsztyn, Poland
Żonierska 14 10-561 Olsztyn, Poland
{mokrzycki,samko}@matman.uwm.edu.pl

Abstract. In this paper the novel modification of the well known Canny edge detection algorithm is presented. The first section describes the goal to be achieved by using the new algorithm. The second section describes theoretical basis of Canny algorithm and its practical implementation. Next, basics of the Ramer–Douglas–Peucker algorithm used for reducing the number of points in the curve are presented. The extension of the Canny algorithm and its implementation are presented in the fourth section. The next section shows the results of the new algorithm implementation for various images and presents statistical data to report effectiveness of the proposed algorithm modification.

1 Introduction

One of the most famous and commonly used edge detectors is Canny edge detector. Apart from simple filtering of the input image, the algorithm has a few optimization stages that make edges one-pixel wide and remove spaces between edge fragments to make them continous. The purpose of the researches was to extend Canny algorithm so that detected edges are prepared to be stored in beamlets structures.

Beamlets are a special dyadically organized collection of line segments, exhibiting a range of lengths, positions and orientations [1]. This collection is stored in multiscale pyramidal structure used for analysing linear features in two dimensional space. Relatively few line segments stored in beamlets could build quite general curves [1].

To prepare detected edges for storing in beamlets, the algorithm extension should make detected edges approximated by polygonial curve. Secondary result of this modification is reducing the number of pixels describing the edge. The algorithm extension, which uses Ramer–Douglas–Peucker algorithm, is started while the last stage of Canny algorithm (binarization with hysteresis) is performed.

2 Canny Edge Detector

2.1 Ideal Step Edge Detector

The goal of the Canny's researches was to find an ideal detector of the step edges. Canny assumed that such an ideal detector should satisfy the following conditions [2]:

L. Bolc et al. (Eds.): ICCVG 2012, LNCS 7594, pp. 533–540, 2012.

1. Low level of the edge detection errors. The probability that the pixel, which does not belong to the edge in the input image, is marked as an edge pixel should be as low as possible. The probability of omitting (not marking) a pixel that is really an edge pixel should also be low. This criterion is mathematically represented by the following formula:

$$SNR = \frac{A\left|\int\limits_{-W}^{0} f(x)dx\right|}{n_0\sqrt{\int\limits_{-W}^{W} f^2(x)dx}} \tag{1}$$

   where: A – step edge amplitude, n_0 – standard deviation of the white gaussian noise, f – impulse response of the filter.

2. Good localization of detected edge. Dislocation between the detected edge and the real edge in the input image should be as small as possible. Thus, pixels marked by the detector as edge pixels should be placed as close to the center of the real edge as possible. Mathematical representation of this criterion is:

$$Localization = \frac{A|f'(0)|}{n_0\sqrt{\int\limits_{-W}^{W} f'^2(x)dx}} \tag{2}$$

   where: $f'(x)$ – first derivative of the filter impulse response.

3. Single response for single edge in the image. For each edge in the input image there should be exactly one response of the detector. This constraint is already included in the first criterion (low level of detection errors) – when a single edge gives two responses, one of them is incorrect. The following formula describes this criterion:

$$x_{zc} = \pi\left(\frac{\int\limits_{-\infty}^{\infty} f'^2(x)dx}{\int\limits_{-\infty}^{\infty} f''^2(x)dx}\right)^{\frac{1}{2}} \tag{3}$$

   where: $f''(x)$ – second derivative of the filter impulse response.

Having considered the criteria depicted above, Canny found the filter which maximizes the first and the second criterion and satisfies single response for single edge limitation. Due to the fact that the resultant filter was too complex to have analytic solution, Canny proposed its effective approximation. This is the first derivative of gaussian operator:

$$\nabla G(x,y) = \left(\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right)' \tag{4}$$

2.2 Algorithm Implementation

The first step of the Canny algorithm is the input image convolution with the found operator (4). In practical implementations instead of convolving image with first derivative of gaussian, convolution of image with gaussian followed by derivative calculation is often performed. Both operations are equal which results from convolution properties:

$$\nabla I'(x,y) = \nabla(I(x,y) * G(x,y)) = I(x,y) * \nabla G(x,y)$$

In this paper the second way of convolving (convolving with gaussian and calculating derivative) in Canny algorithm is used. Thus, the algorithm starts with image smoothing using gaussian:

$$I'(x,y) = I(x,y) * G(x,y)$$

where: $I(x,y)$ – resultant smoothed image, $I(x,y)$ – input image, $G(x,y)$ – gaussian operator.

Convolving image with two-dimensional gaussian is computationally complex. That is why it is commonly approximated by image convolution with one-dimensional gaussian in two perpendicular directions.

Next, differentiation in x and y directions is performed for a smoothed image:

$$\nabla_x I(x,y) = \tfrac{\partial I'(x,y)}{\partial x},\ \nabla_y I(x,y) = \tfrac{\partial I'(x,y)}{\partial y}$$

On the basis of the calculated partial derivatives of the smoothed image $I'(x,y)$ the gradient module and direction are determined:

$$M(x,y) = \sqrt{(\nabla_x I'(x,y))^2 + (\nabla_y I'(x,y))^2} \tag{5}$$

$$\Theta(x,y) = \arctan\frac{\nabla_y I'(x,y)}{\nabla_x I'(x,y)} \tag{6}$$

where: $M(x,y)$ – gradient module, $\Theta(x,y)$– angle between $M(x,y)$ vector and x axis of coordinate system.

The next stage of the algorithm is so called non-maximal suppression. It is performed to ensure one-pixel wide edge on the output of the algorithm. In direction perpendicular to the edge only one pixel with maximal gradient module value is preserved as the candidate edge pixel. Other pixels are suppressed – their value is set to background value. This operation is performed by testing 3x3 neighbourhood of each pixel and comparing the gradient module value of the central pixel with the values of the neighbour pixels in the gradient direction (perpendicular to the edge). If the central pixel has the maximum value, it is marked as the candidate edge pixel, otherwise its value is set to the background value.

The last stage of the algorithm is binarization. To avoid discontinuous edge on the output, Canny proposed binarization with hysteresis. This method consists in setting two thresholds. The candidate edge segment is added to the resultant edge map if at least one of its pixels has gradient module value greater or equal to high threshold T_H and other pixels have gradient module value not less than low threshold T_L:

$$L'(x,y) = \begin{cases} 0, \text{ if } & L(x,y) < T_L \\ s, \text{ if } & T_L \leq L(x,y) < T_H \\ 1, \text{ if } & L(x,y) \geq T_H \end{cases} \tag{7}$$

where: $L(x,y)$ – pixel value in source image, $L'(x,y)$ – pixel value in resultant image, T_L – low threshold, T_H – high threshold, $s = 0$ – if pixel does not neighbour with edge pixel, $s = 1$ – if pixel neighbours with edge pixel.

3 Ramer–Douglas–Peucker Algorithm

The purpose of the algorithm is to reduce the number of points describing the curve. Let the curve C_1 be described by the set of points $A = \{n_1, n_2, \ldots, n_p\}$. We want to find curve C_2 that is described by the set of points $B \subset A$ with the assumed accepted error ε.

The algorithm is illustrated in figure 1 and implemented as follows [3]:

1. The first and the last point of the curve C_1 are connected with the segment: $|n_1 n_p|$ (fig. 1b). Points n_1 and n_p are added to resultant set B.
2. From among other points of the curve $\{n_2 \ldots n_{p-1}\}$ point n_k is found, whose distance x from the segment $|n_1 n_p|$ is the largest (fig. 1c).
3. If $x \leq \varepsilon$ then the algorithm is finished. In this case set $B = \{n_1, n_p\}$ and new curve C_2 is a segment $|n_1 n_p|$. Otherwise point n_k is added to B and algorithm is recursively started for curves: C_{1k} described by $\{n_1 \ldots n_k\}$ and C_{kp} described by $\{n_k \ldots n_p\}$ (fig. 1d).

4 Canny Algorithm Modification

As it was shown in the introduction, the goal of the algorithm modification is to reduce the complexity of the edges description and describe edges in the form proper to store them in a tree-based structures like beamlets. The first already implemented step consists in reducing number of pixels describing edges. This is achieved by using Ramer–Douglas–Peucker algorithm for edges detected by the standard Canny algorithm. On the output of the modified algorithm we get edges represented by curves that could be easily prepared for being stored in the beamlet. In this chapter the implementation of the proposed extension is described in more detail.

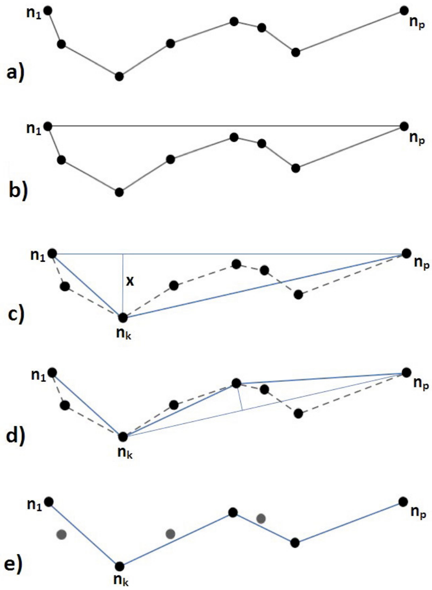

Fig. 1. Input curve, specified stages of the Ramer–Douglas–Peucker algorithm, output curve with reduced number of points

As it was shown earlier, the last stage of the Canny algorithm is binarization with hysteresis. The implementation of this stage starts with scanning pixels of the image and checking their value. When the first pixel with value greater or equal T_H is found, it is marked as an edge pixel. Next, eight of its neighbours are checked and their value is compared to the T_L. The pixel that has value greater or equal to T_L is marked as an edge pixel and recursively its neighbours are scanned. Thanks to this procedure, the chain of neighbouring pixels forming an edge is obtained. After reaching the last edge pixel, the algorithm looks for other edges using the same procedure. Having obtained a set of edges, each formed by a chain of pixels, the algorithm labels every single edge.

Now every labeled edge consists of a set of neighbouring pixels. The number of pixels is then reduced by passing the set of pixels to the input of Ramer–Douglas–Peucker algorithm for specified error level ε. On the output we get an edge described by the reduced number of points and represented by polygonal curve. This procedure is repeated for all labeled edges.

5 Experiments

Several experiments on various pictures were conducted during the research. In the real pictures the number of pixels describing edges was significantly reduced. Even with the minimal error $\varepsilon = 1$ the number of pixels was not bigger than 50% of the pixels obtained in the standard Canny algorithm. The obtained edges are represented by polygonal curves and can be easily prepared for storage in beamlets structures. The results of the algorithm performance for sample pictures are shown in figures 2, 3 and 4. Also statistics are presented in table 1. It shows error level ε, the number of pixels in the standard Canny algorithm output P_C (before reduction), the number of pixels in the modified algorithm output P_M (after reduction) and the calculated reduction rate for all input images.

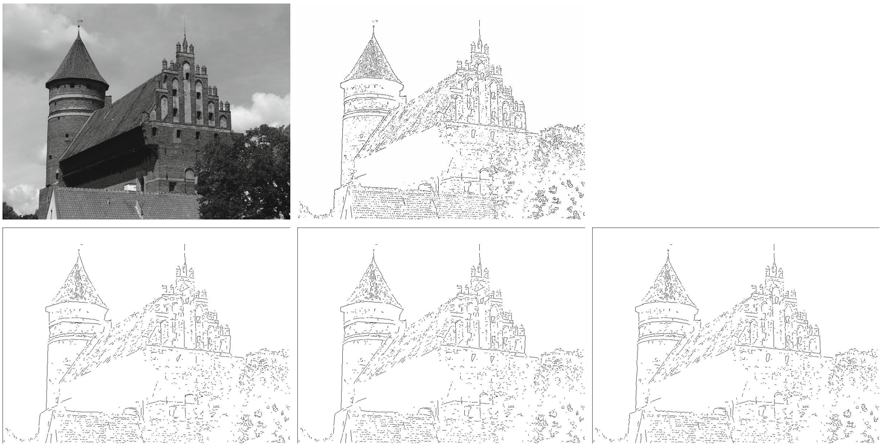

Fig. 2. Castle–input image, output of standard Canny algorithm and output of modified algorithm for $\varepsilon = 10$, $\varepsilon = 5$ and $\varepsilon = 1$ respectively

Fig. 3. Gate–input image, output of standard Canny algorithm and output of modified algorithm for $\varepsilon = 10$, $\varepsilon = 5$ and $\varepsilon = 1$ respectively

Fig. 4. Town hall–input image, output of standard Canny algorithm and output of modified algorithm for $\varepsilon = 10$, $\varepsilon = 5$ and $\varepsilon = 1$ respectively

Table 1. Statistic data of the modified alorithm performance

Image	ε	P_C	P_M	Reduction rate
Castle	10	14075	5034	35.77%
	5	14075	5040	35.81%
	1	14075	5694	40.45%
Gate	10	24237	5950	24.55%
	5	24237	5970	24.63%
	1	24237	6775	27.95%
Town hall	10	9112	2337	25.65%
	5	9112	2345	25.74%
	1	9112	2617	28.72%

6 Conclusions

The paper described Canny algorithm and its implementation. It presented modification of the Canny algorithm and its implementation followed by descriptions and the results of experiments based on various pictures. The modification included using Ramer–Douglas–Peucker algorithm to reduce the number of pixels and present the edges as polygonal curves which allow storing edges in beamlets. Although this article constitutes a discussion on the modification of Canny algorithm, due to limited resources the subject has not been fully examined and thus needs more investigation.

References

1. Donoho, D., Huo, X.: Applications of Beamlets to Detection and Extraction of Lines, Curves and Objects in Very Noisy Images. Nonlinear Signal and Image Processing (NSIP), Baltimore (2001)
2. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8 (1986)
3. Internet resource, `http://en.wikipedia.org/wiki/Ramer-Douglas-Peucker_algorithm` (accessed April 24, 2012)
4. Heath, M., Sarkar, S., Sanocki, T., Bowyer, K.: Comparison of Edge Detectors. Computer Vision and Image Understanding 69(1), 38–54 (1998)