

Edge Detection with Sub-pixel Accuracy Based on Approximation of Edge with Erf Function

Miroslav HAGARA, Peter KULLA

Dept. of Radio and Electronics, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava, Slovakia

miroslav.hagara@stuba.sk, peter.kulla@stuba.sk

Abstract. Edge detection is an often used procedure in digital image processing. For some practical applications it is desirable to detect edges with sub-pixel accuracy. In this paper we present edge detection method for 1-D images based on approximation of real image function with Erf function. This method is verified by simulations and experiments for various numbers of samples of simulated and real images. Results of simulations and experiments are also used to compare proposed edge detection scheme with two often used moment-based edge detectors with sub-pixel precision.

Keywords

Edge detection, sub-pixel accuracy, image processing.

1. Introduction

Physical contour is one of the most important properties of an object. In order to extract the contour of an object, we must detect the edges forming that object. So a lot of methods were developed to detect edges. One can use simple edge detectors like Sobel, Kirsch, Prewit [1], more sophisticated Canny edge detector [2] or robust morphological edge detectors [3], [4]. All these methods perform edge detection with a pixel accuracy.

Sometimes it is useful to detect edges with sub-pixel precision. Most edge detectors at sub-pixel level fall in three groups: fitting, moment-based and interpolation-based methods. The methods of the first group use continuous functions, such as hyperbolic tangent [5] or B-spline [6], to fit samples of image function. Then sub-pixel edge location is found as inflection point of continuous function. Another fitting methods use a local energy function [7] or wavelets [22] to determine the edge parameters.

Interpolation-based methods achieve the sub-pixel accuracy by interpolating the image data to obtain a finer grid of pixels. Then usual edge detectors, such as Canny [8] or LoG operator [9], are applied to resized image.

Moment-based methods apply statistical moments to determine unknown edge model parameters. One can use

gray level moments [10, 11], spatial moments [12, 13], Fourier-Mellin moments [14] or Zernike moments [15, 16].

Some industrial applications, e.g. measurement of the objects with high precision, need to detect edges with sub-pixel accuracy in 1-D images. For such a task we introduce in this paper sub-pixel edge detection method based on approximation of real image function with Erf function. We compare the proposed algorithm with two often used sub-pixel edge detectors: gray level moment (GLM) edge operator [10] and spatial moment (SM) edge detector [12].

This paper is organized as follows. Section 2 includes edge models. In section 3 we describe moment-based edge detection methods which we use for comparison with our proposed algorithm. This algorithm is introduced in section 4 results. Section 5 and section 6 include simulations and experiments. In section 7 conclusions are made.

2. Edge Models

For analysis of proposed edge detectors and their verification by simulations three basic models are mostly used. Step edge (Fig. 1) is a simplest model and is represented by step function [12]:

$$f(x) = \begin{cases} h, & x < l \\ h+k, & x \geq l \end{cases} \quad (1)$$

The model is characterized by three parameters: background intensity h , edge contrast k , and edge location l .

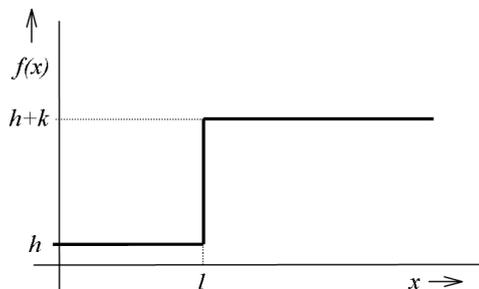


Fig. 1. Step edge.

In real images the brightness changes gradually and ramp edge (Fig. 2) is more suitable [7]:

$$f(x) = \begin{cases} h, & x < l_1 \\ h + \frac{k}{l_2 - l_1}(x - l_1), & l_1 \leq x < l_2 \\ h + k, & l_2 \leq x \end{cases} \quad (2)$$

Ramp edge has four parameters: background intensity h , edge contrast k , edge beginning l_1 and edge end l_2 . Location of the edge l is equal to the arithmetic average of l_1 and l_2 .

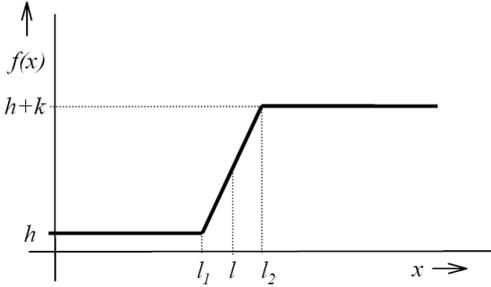


Fig. 2. Ramp edge.

The third model is closest to real edge because it respects defocusing, or blurring due to the effects of the point spread function of the optic system. This model (Fig. 3) is called blurred edge and is represented by function [17]:

$$f(x) = \frac{k}{2} \left(\operatorname{erf} \left(\frac{x-l}{\sqrt{2}\sigma} \right) + 1 \right) + h \quad (3)$$

where $\operatorname{erf}(x)$ is defined as [18]:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (4)$$

This model has four parameters: background intensity h , edge contrast k , edge location l and edge blurring σ .

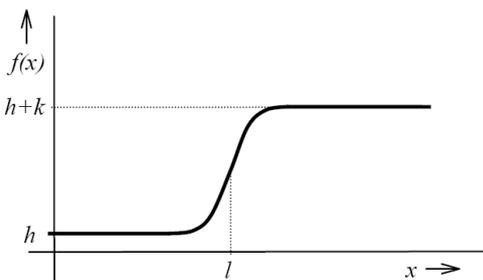


Fig. 3. Blurred edge.

3. Moment-Based Edge Operators

Tabatabai and Mitchel proposed gray level moment (GLM) edge operator for 1-D image [10] based on the first three moments m_1, m_2, m_3 of the input data sequence:

$$m_i = \frac{1}{n} \sum_{j=1}^n x_j^i \dots\dots i=1,2,3 \quad (5)$$

where x_1, x_2, \dots, x_n are image samples. Let suppose that they are the samples of ideal step edge (Fig. 1) and p_h is a number of samples with gray level h (they are the pixels on the left of the edge). If we define the densities p_1 and p_2 as:

$$p_1 = \frac{p_h}{n}, \quad (6)$$

$$p_2 = \frac{n - p_h}{n} = 1 - p_1, \quad (7)$$

then solution of three equations:

$$m_1 = (1 - p_2)h + p_2(h + k), \quad (8)$$

$$m_2 = (1 - p_2)h^2 + p_2(h + k)^2, \quad (9)$$

$$m_3 = (1 - p_2)h^3 + p_2(h + k)^3, \quad (10)$$

with three unknown variables h, k, p_2 results in:

$$p_2 = \frac{1}{2} \left(1 - s \sqrt{\frac{1}{4 + s^2}} \right), \quad (11)$$

$$h = m_1 - \beta \sqrt{\frac{p_2}{p_1}}, \quad (12)$$

$$k = 2\beta \sqrt{\frac{p_2}{p_1}} \quad (13)$$

where

$$s = \frac{m_3 + 2m_1^3 - 3m_1m_2}{\beta^3}, \quad (14)$$

$$\beta^2 = m_2 - m_1^2. \quad (15)$$

In the case of real image, $p_h = n \cdot p_1$ is not integer and represents sub-pixel edge location.

Another sub-pixel edge detector [12] is based on spatial moments (SM) of continuous function $f(x)$ of order p , which are defined:

$$M_p = \int x^p f(x) dx. \quad (16)$$

Let function $f(x)$ represents step edge (Fig. 4) and x is from -1 to +1 (to simplify calculations). Then equation (16) for $p=0, 1$ and 2 can be written as:

$$M_0 = h \int_{-1}^l dx + k \int_l^1 dx = 2h + k(1-l), \quad (17)$$

$$M_1 = h \int_{-1}^l x dx + k \int_l^1 x dx = \frac{1}{2} k(1-l^2), \quad (18)$$

$$M_2 = h \int_{-1}^l x^2 dx + k \int_l^1 x^2 dx = \frac{2}{3} h + \frac{1}{3} k(1-l^3). \quad (19)$$

The solution of these equations results in formulas for edge location l :

$$l = \frac{3M_2 - M_0}{2M_1}, \quad (20)$$

edge contrast k

$$k = \frac{2M_1}{(1-l^2)}, \quad (21)$$

and background intensity h

$$h = \frac{1}{2}[M_0 - k(1-l)]. \quad (22)$$

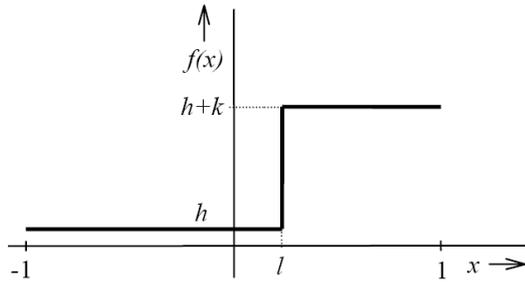


Fig. 4. Edge model for spatial moment edge detector.

4. Edge Detector Based on Approximation with Erf Function (AEF)

We propose the sub-pixel edge detector based on approximation of real image function $f_r(i)$ with function $f_a(i)$, which is equal to blurred edge model (3) and has four parameters - h , k , l and σ . The core of this AEF edge detector is parametric fitting by minimizing a difference between the real image function $f_r(i)$ and function $f_a(i)$. This difference is defined:

$$E(h, k, l, \sigma) = \sum_{i=1}^N (f_r(i) - f_a(i))^2 \quad (23)$$

where N is a number of samples. Minimizing the difference $E(h, k, l, \sigma)$ gives subpixel edge location l . Edge detection algorithm based on approximation consists of three steps:

- Edge detection with pixel accuracy.
- Initial values estimation of parameters h , k , l and σ .
- Parametric fitting by minimizing difference function $E(h, k, l, \sigma)$.

The first step can be done by any edge detection method with pixel accuracy. We use the simplest way, we find maximum of discrete derivative of function $f_r(i)$:

$$df_r(i) = f_r(i+1) - f_r(i) \quad (24)$$

and we denote i_{\max} for which $df_r(i)$ reaches its maximum as the edge position.

Initial value of edge location we set to $l_0 = i_{\max}$. To estimate initial value of σ we apply derivation of continuous function $f_a(x)$ [18]:

$$f'_a(x) = \frac{k}{\sqrt{2\pi\sigma}} \exp\left(-\left(\frac{x-l}{\sqrt{2\sigma}}\right)^2\right). \quad (25)$$

For $x_{\max}=l$, $x_d=(l-2\sigma)$ and $x_u=(l+2\sigma)$ we can derive:

$$\frac{f'_a(x_d)}{f'_a(x_{\max})} = \frac{f'_a(x_u)}{f'_a(x_{\max})} = \frac{\exp(-2)}{\exp(0)} \cong 0.135. \quad (26)$$

In the case of $f_r(i)$ if we find $i_d < i_{\max}$ and $i_u > i_{\max}$ which satisfy:

$$\frac{df_r(i_d)}{df_r(i_{\max})} \leq 0.135 \quad (27)$$

and

$$\frac{df_r(i_u)}{df_r(i_{\max})} \leq 0.135 \quad (28)$$

and are as close as possible to i_{\max} , we can estimate $\sigma_0 = 0.25(i_u - i_d)$ and also $h_0 = f_r(i_d)$ and $k_0 = f_r(i_u) - f_r(i_d)$.

For parametric fitting by minimizing difference function $E(h, k, l, \sigma)$ we apply Matlab function *fminsearch* which uses the simplex search method [19].

5. Simulations

We did all simulations in program Matlab (version 7.3.0.267). Let there is 1-D image sensor which consists of elements with width w and gap g between two sensor elements (Fig. 5b). Let the brightness around the edge is constant in time and varies only in the direction x according to (3) (Fig. 5a). Then simulated noiseless output signal $f_{rs}(i)$ of the i -th sensor element (Fig. 5c) can be calculated:

$$f_{rs}(i) = cT_a \int_{i-w/2}^{i+w/2} \left(\frac{k}{2} \left(\operatorname{erf}\left(\frac{x-l}{\sqrt{2\sigma}}\right) + 1 \right) + h \right) dx \quad (29)$$

where c is sensor integral sensitivity and T_a is accumulation time. For simulations we can set $cT_a = 1$. Because the gap between two sensor elements is very small we can set $w=1$. Then noiseless output signal $f_{rs}(i)$ of the i -th sensor element is:

$$f_{rs}(i) = \int_{i-0.5}^{i+0.5} \left(\frac{k}{2} \left(\operatorname{erf}\left(\frac{x-l}{\sqrt{2\sigma}}\right) + 1 \right) + h \right) dx. \quad (30)$$

For all simulations in this section the background intensity used in (30) is $h=0.1$ and edge contrast is $k=1$.

The first simulations of noiseless signal inspect how the edge location error depends on actual position of the edge for different values of blurring parameter $\sigma = 0.5, 1, 2$

and 5. Number of samples of output signal $f_{rs}(i)$ is $N = 41$. The results of these simulations are presented in Tab. 1, and graphically are interpreted in Fig. 6.

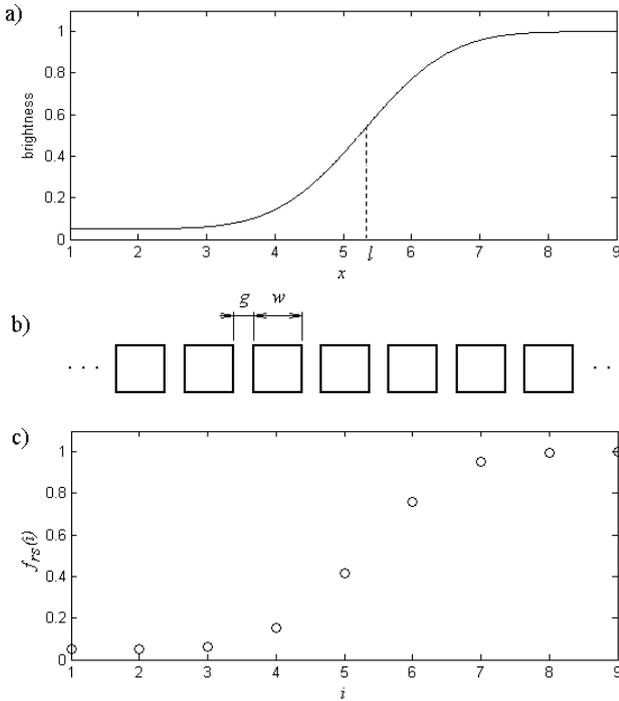


Fig. 5. a) Brightness around the edge, b) elements of image sensor, c) simulated noiseless output signal.

l	$\sigma = 0.5$			$\sigma = 1$		
	AEF	GLM	SM	AEF	GLM	SM
-0.5	0	0,0002	0,001	0	0,0007	0,0028
-0.4	0,0007	-0,0096	0,0008	0	0,0005	0,0022
-0.3	0,0012	-0,0156	0,0006	0	0,0004	0,0017
-0.2	0,0013	-0,0157	0,0004	0	0,0002	0,0011
-0.1	0,0009	-0,0097	0,0002	0,0001	0,0001	0,0006
0	0	0	0	0	0	0
0.1	-0,0009	0,0097	-0,0002	-0,0001	-0,0001	-0,0006
0.2	-0,0013	0,0156	-0,0004	0	-0,0002	-0,0011
0.3	-0,0011	0,0156	-0,0006	0	-0,0004	-0,0017
0.4	-0,0007	0,0096	-0,0008	0	-0,0005	-0,0022
0.5	0	-0,0002	-0,001	0	-0,0007	-0,0028
l	$\sigma = 2$			$\sigma = 5$		
	AEF	GLM	SM	AEF	GLM	SM
-0.5	0	0,0028	0,01	0	0,0243	0,0637
-0.4	0	0,0023	0,008	0	0,0194	0,0509
-0.3	0	0,0017	0,006	0	0,0146	0,0382
-0.2	0	0,0011	0,004	0	0,0097	0,0255
-0.1	0	0,0006	0,002	0	0,0049	0,0127
0	0	0	0	0	0	0
0.1	0	-0,0006	-0,002	0	-0,0049	-0,0127
0.2	0	-0,0011	-0,004	0	-0,0097	-0,0255
0.3	0	-0,0017	-0,006	0	-0,0146	-0,0382
0.4	0	-0,0023	-0,008	0	-0,0194	-0,0509
0.5	0	-0,0028	-0,01	0	-0,0243	-0,0637

Tab. 1. Edge location error of simulated noiseless signal.

The edge with blurring $\sigma = 0.5$ is very close to step function and such a case does not occur in real images. Real values of blurring in well-focused images are approximately equal to $\sigma = 1$. One can see from simulation results that for this value the precision of all three methods presented in this paper is theoretically very high. For

slightly unfocused images ($\sigma = 2$) the theoretical accuracy of GLM and SM is pretty good (better then 0.01), for strongly unfocused images ($\sigma = 5$) come close to 0.07. AEF method has (except for $\sigma = 0.5$) zero location error. It's understandable, because the function under which an output signal is generated for simulation is equal to the function used for approximation. Simulations with noisy signal and experiments with real images help to determine real properties of AEF.

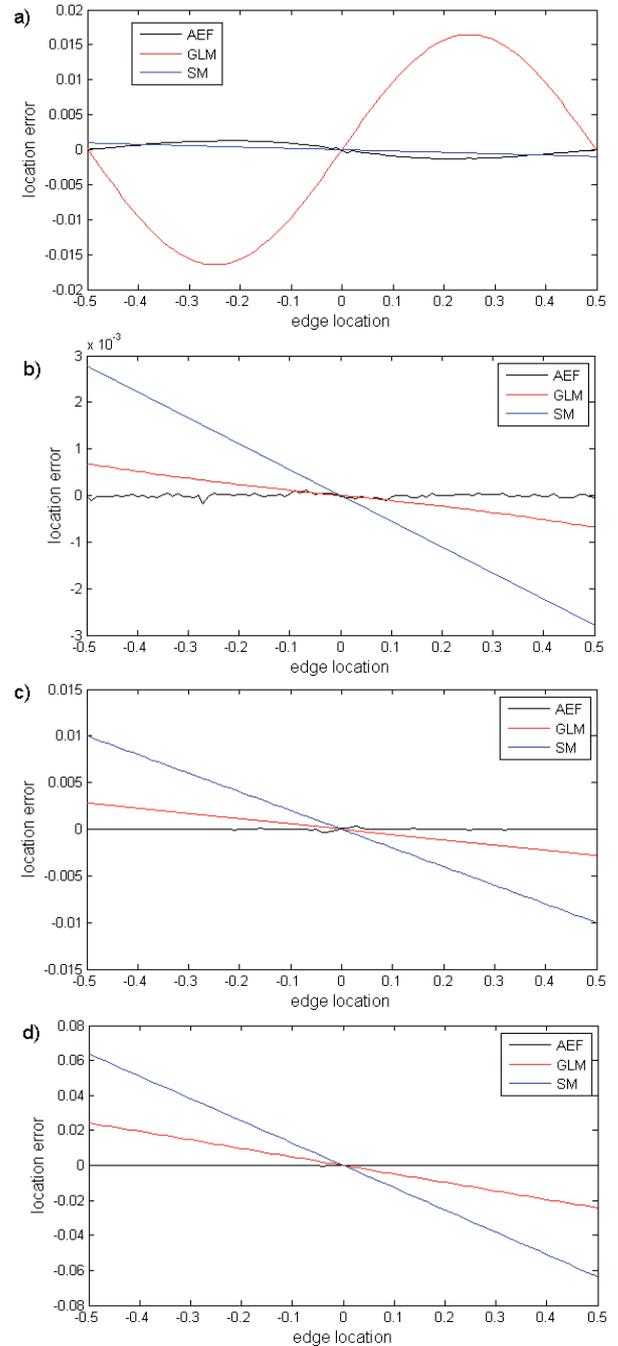


Fig. 6. Edge location error of simulated noiseless signal for blurring parameter: a) $\sigma = 0.5$, b) $\sigma = 1$, c) $\sigma = 2$ and d) $\sigma = 5$.

To add noise to signal defined in (30) we apply Matlab function *randn*, which returns a pseudorandom, scalar

value drawn from a normal distribution with mean 0 and standard deviation 1. Values of function *randn* we multiply with 0.008, so we get signal-to-noise ratio 37dB. We calculated noisy signal hundred times and for each noisy signal we used successively 41, 21 and 11 samples around the edge to find sub-pixel edge location. Then we calculated edge location error for all hundred realizations. For these hundred errors we determined the standard deviation and upper q_u and lower q_d 5% quantiles. Upper 5% quantile q_u means, that there is 5% probability that the error will be bigger than q_u . Lower 5% quantile q_d means, that there is 5% probability that the error will be smaller than q_d .

We did calculations mentioned above for different values of blurring parameter $\sigma = 0.1, 1, 2$ and 5. The results are presented in Tab. 2 and Tab. 3, some of them are graphically interpreted in Fig. 7 and Fig. 8.

N	$\sigma = 0.5$			$\sigma = 1$		
	AEF	GLM	SM	AEF	GLM	SM
41	0,0121	0,0127	0,0442	0,0161	0,0167	0,0454
21	0,0123	0,0128	0,0327	0,0163	0,0169	0,0335
11	0,0129	0,0133	0,0236	0,0187	0,0187	0,0241
N	$\sigma = 2$			$\sigma = 5$		
	AEF	GLM	SM	AEF	GLM	SM
41	0,0236	0,0244	0,0462	0,0438	0,0436	0,0482
21	0,0253	0,0262	0,0335	0,0714	0,0553	0,0431
11	0,0371	0,0318	0,0278	0,2377	0,0722	0,0447

Tab. 2. Standard deviation of edge location error of simulated noisy signal.

N	$\sigma = 0.5$					
	AEF		GLM		SM	
	q_u	q_d	q_u	q_d	q_u	q_d
41	0,022	-0,0207	0,024	-0,0239	0,091	-0,0921
21	0,0225	-0,0217	0,025	-0,0242	0,0652	-0,0639
11	0,0235	-0,0234	0,0267	-0,0249	0,0472	-0,0471
N	$\sigma = 1$					
	AEF		GLM		SM	
	q_u	q_d	q_u	q_d	q_u	q_d
41	0,0324	-0,0326	0,0335	-0,0323	0,0928	-0,0867
21	0,0341	-0,0327	0,0341	-0,0329	0,0644	-0,0695
11	0,0363	-0,0354	0,037	-0,0365	0,0445	-0,0478
N	$\sigma = 2$					
	AEF		GLM		SM	
	q_u	q_d	q_u	q_d	q_u	q_d
41	0,0452	-0,0461	0,0473	-0,0471	0,0993	-0,0852
21	0,0523	-0,0545	0,0509	-0,0518	0,0649	-0,0675
11	0,0722	-0,0727	0,0616	-0,061	0,0544	-0,0538
N	$\sigma = 5$					
	AEF		GLM		SM	
	q_u	q_d	q_u	q_d	q_u	q_d
41	0,0883	-0,0858	0,0876	-0,0816	0,1	-0,097
21	0,1412	-0,149	0,1026	-0,109	0,081	-0,0876
11	0,5017	-0,5093	0,1369	-0,1484	0,0865	-0,0921

Tab. 3. Upper q_u and lower q_d 5% quantiles of edge location error of simulated noisy signal.

From the presented results it is clear that for small blurring parameters ($\sigma = 0.5$ and 1) the number of samples used for calculations is not significant. This is because visual function in this case varies only in the vicinity of edge. For sufficient number of samples the precision of AEF and GLM methods depends only on blurring parameter (see Fig. 7 and Fig. 8). By contrast, SM method

has better precision for smaller number of samples. However, for sufficient number of samples AEF and GLM are more precise than SM.

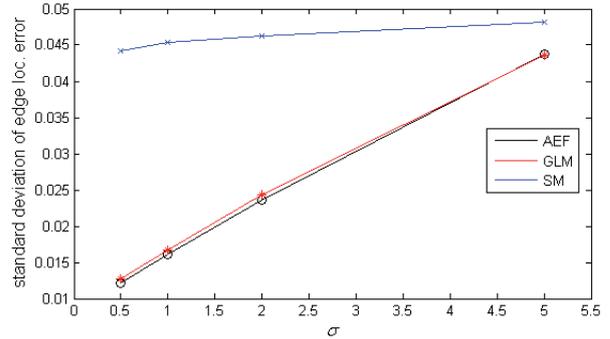


Fig. 7. Standard deviation of edge location error of simulated noisy signal.

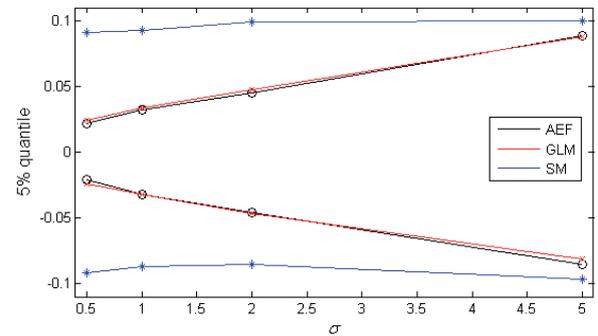


Fig. 8. Upper q_u and lower q_d 5% quantiles of edge location error of simulated noisy signal.

6. Experiments



Fig. 9. Source for test images (car engine valve).

For experimental verification we shot few images of car engine valve (Fig. 9). As a background we used PC monitor so we got good edge contrast. Monitor was not immediately behind the valve to get background without texture (due to un-focusing). We used camera Nikon D300 with 12 mega-pixel resolution. We shot some images with automatic focusing and some with manual focusing so we got images with different values of blurring parameter. Uncompressed images were converted from original NEF

format to 24 bits color TIFF format. For computations we used detail of 150x80 pixels around the edge (Fig. 10). We worked only with green channel of image.

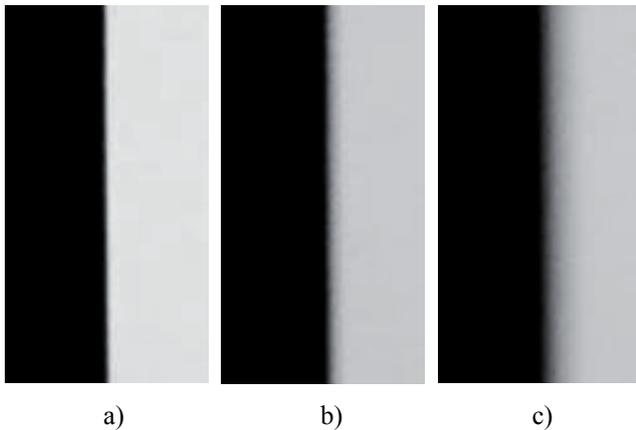


Fig. 10. Test images: a) “valve1.tiff”, b) “valve2.tiff” and c) “valve3.tiff”.

We applied AEF, GLM and SM edge detection methods to one hundred adjacent rows to compute the edge location with the sub-pixel accuracy. Since the valve must be perfectly straight, computed sub-pixel edge positions should create a straight line, which can be represented as polynomial $p(x) = a_1x + a_0$. We used the Matlab function *polyfit* to find the coefficients a_1 and a_0 of a polynomial $p(x)$ that fits the data. Difference between the computed edge position and the value of the polynomial can be considered to be the edge location error. In such a way we computed the edge location errors for all processed rows. Standard deviation and 5% quantiles were also computed.

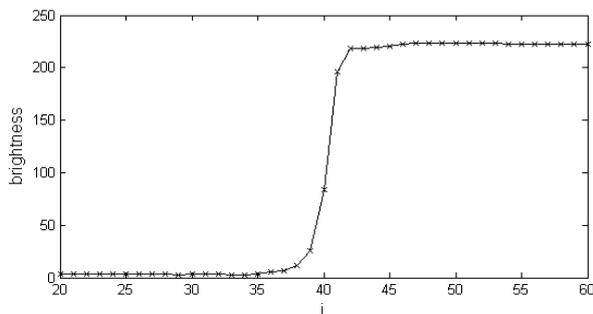


Fig. 11. Row n.36 of test image “valve1.tiff”.

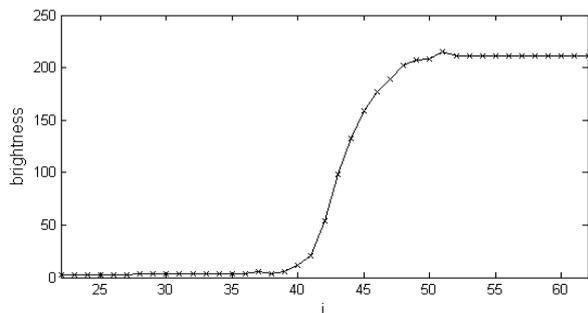


Fig. 12. Row n.36 of test image “valve2.tiff”.

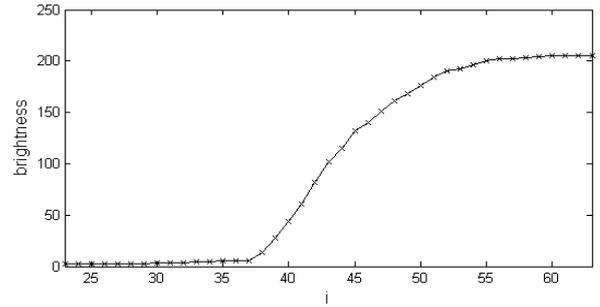


Fig. 13. Row n.36 of test image “valve3.tiff”.

To illustrate examples, there are selected rows of test images “valve1.tiff”, “valve2.tiff” and “valve3.tiff” in Fig. 11, Fig. 12 and Fig. 13. Computed edge locations and approximating straight lines are in Fig. 14, Fig. 15 and Fig. 16.

For each row of the test image the edge position was calculated using a different number of pixels around the edge $N = 41, 21$ and 11 . Calculated polynomial coefficients a_1 and a_0 , standard deviation, 5% upper q_u and lower q_d quantiles of edge location error of test images are presented in Tab. 4.

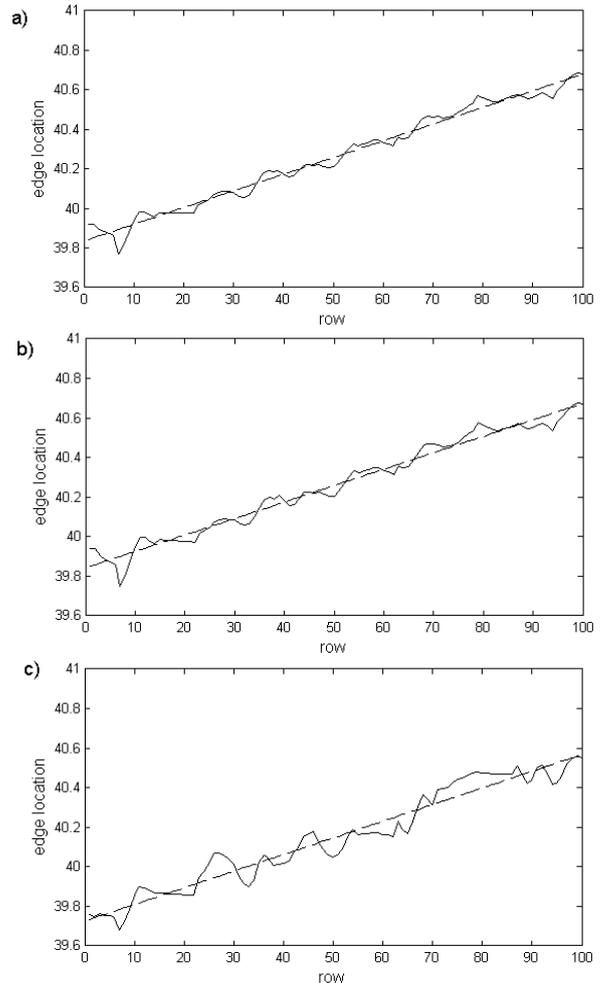


Fig. 14. Edge locations and approximating straight line of test image “valve1.tiff”: a) AEF, b) GLM, c) SM.

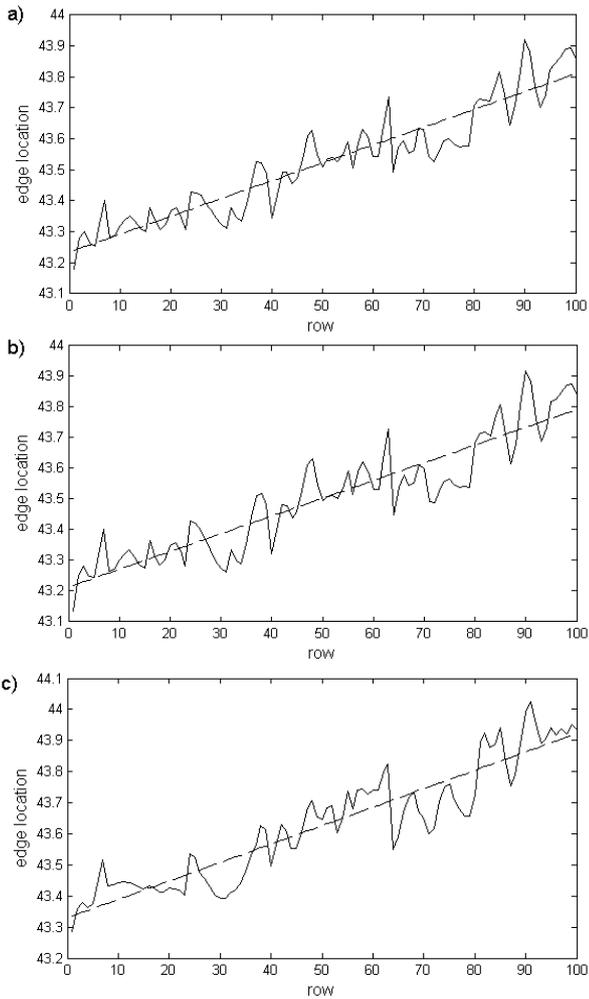


Fig. 15. Edge locations and approximating straight line of test image “valve2.tiff”: a) AEF, b) GLM, c) SM.

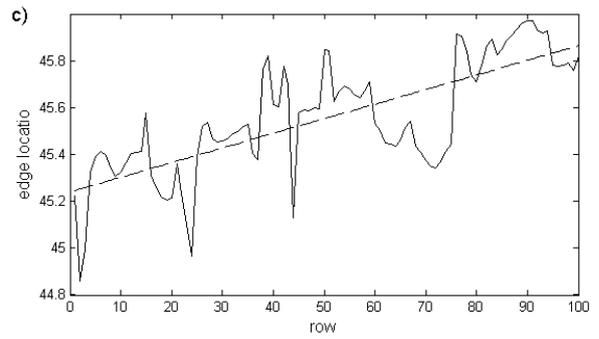
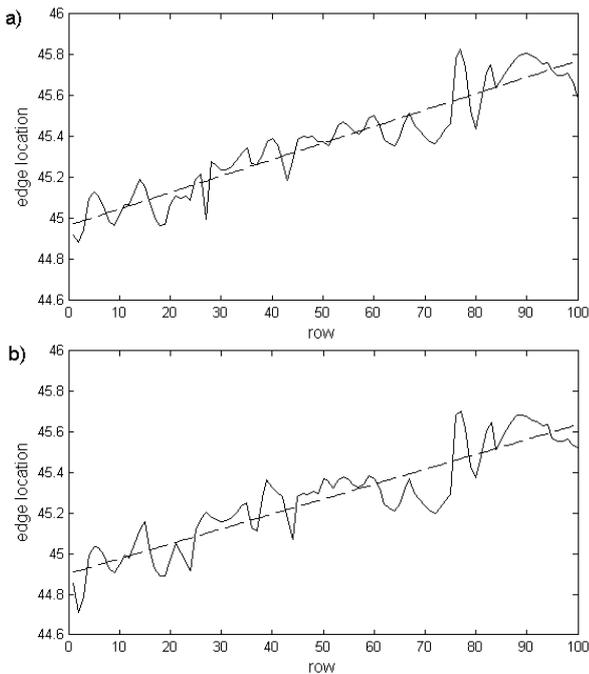


Fig. 16. Edge locations and approximating straight line of test image “valve3.tiff”: a) AEF, b) GLM, c) SM.

N		valve1.tiff ($\sigma = 0,85$)				
		a_1	a_0	std	q_u	q_d
41	AEF	0,0084	39,84	0,033	0,048	-0,048
	GLM	0,0083	39,84	0,039	0,055	-0,052
	SM	0,0084	39,73	0,058	0,088	-0,099
21	AEF	0,0084	39,85	0,033	0,048	-0,043
	GLM	0,0083	39,85	0,039	0,057	-0,048
	SM	0,0085	39,76	0,052	0,086	-0,095
11	AEF	0,0084	39,86	0,034	0,044	-0,042
	GLM	0,0082	39,87	0,042	0,056	-0,057
	SM	0,0080	39,81	0,051	0,074	-0,085
N		valve2.tiff ($\sigma = 2,16$)				
		a_1	a_0	std	q_u	q_d
41	AEF	0,0058	43,23	0,063	0,104	-0,104
	GLM	0,0058	43,21	0,073	0,127	-0,124
	SM	0,0059	43,33	0,071	0,107	-0,126
21	AEF	0,0057	43,18	0,068	0,112	-0,106
	GLM	0,0058	43,13	0,078	0,115	-0,140
	SM	0,0061	43,21	0,088	0,138	-0,160
11	AEF	0,0059	42,87	0,093	0,145	-0,165
	GLM	0,0061	42,78	0,133	0,218	-0,237
	SM	0,0064	42,73	0,179	0,282	-0,285
N		valve3.tiff ($\sigma = 4,50$)				
		a_1	a_0	std	q_u	q_d
41	AEF	0,0080	44,97	0,088	0,118	-0,157
	GLM	0,0074	44,90	0,104	0,133	-0,182
	SM	0,0062	45,24	0,165	0,242	-0,326
21	AEF	0,0071	44,27	0,148	0,214	-0,279
	GLM	0,0053	44,23	0,235	0,423	-0,427
	SM	0,0037	44,33	0,359	0,775	-0,605
11	AEF	0,0050	43,54	0,452	0,431	-0,323
	GLM	0,0014	43,77	0,497	1,114	-0,746
	SM	0,0003	43,83	0,632	1,460	-0,972

Tab. 4. Polynomial coefficients a_1 and a_0 , standard deviation, 5% upper q_u and lower q_d quantiles of edge location error of test images.

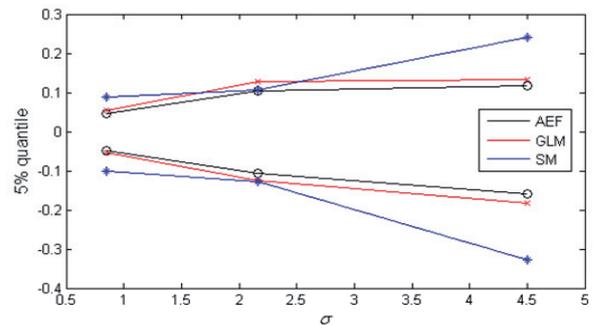


Fig. 17. Dependence of accuracy (upper and lower quantiles) on the blurring parameter σ (for AEF with $N = 41$).

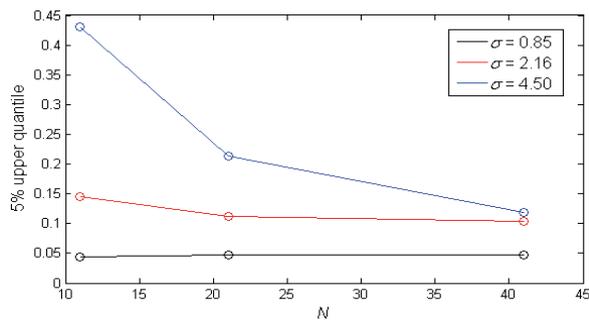


Fig. 18. Dependence of accuracy (upper quantile) on the number of pixels used for computation (for AEF).

From the obtained results one can see that the accuracy of all presented methods (AEF, GLM, SM) get worse with increasing blurring parameter σ (Fig. 16). Also, it can be concluded that for small values of σ the number of pixels used for computation is not important (Fig. 17). This is not true for larger values of σ and according to the experiments we can say that required number of pixels is approximately equal to $8 \div 10 \sigma$.

7. Conclusion

In this paper, sub-pixel edge detection method based on approximation of real image function with Erf function (AEF) is presented. This method is verified through simulations and experiments and compared with two other methods: GLM and SM. Although these methods are not recent, we chose them for comparison because they are designed primarily for 1-D images. Newer methods that are mentioned in the introduction are geared towards the 2-D images and for 1-D images do not achieve such accuracy. There are also few methods specifically designed for 1-D images [20], [21] but they are hardware oriented in order to achieve greater speed and their accuracy is not very high.

The results obtained from experiments are fairly well consistent with the results of simulation for small values of the blurring parameter ($\sigma \approx 1$), which correspond to the well-focused images. Achieved accuracy of edge location for real images is about ± 0.05 of pixel. For slightly ($\sigma \approx 2$) and strongly ($\sigma \approx 5$) unfocused real images the experimental results are more different from the results of simulation. This may be because the brightness around the edge does not fit exactly the used Erf function. It looks like the image on the right of the edge has different parameter σ as the image on the left, probably due to different distances of background and foreground from the camera. But it is typical for the real situations. However, the accuracy of AEF method is about 15–20 % better than GLM and much better than SM. We also found that detection accuracy can be affected by the insufficient number of pixels used for computation and we recommend working with the number of pixels that is $8 \div 10$ bigger than blurring parameter σ . How to estimate this parameter one can find in section 4.

For some industrial applications such as contactless measurement of objects to improve the resolution of sys-

tem can be desirable. As an example we can give magnesite bricks, which are used in blast furnace as lining. These bricks are manufactured by compression and transverse dimension has to be checked with great accuracy. Improvement of installed hardware is usually limited by the cost and realization. In the case, that computational cost is not a critical attribute, the presented method for sub-pixel edge detection can constitute an appropriate solution to this problem.

References

- [1] PRATT, W. K. *Digital Image Processing*. 4th ed. Hoboken: John Wiley & Sons, 2007.
- [2] CANNY, J. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986, vol. 8, no. 6, p. 679-698.
- [3] LI, Z., YANG, Y., JIANG, W. Multi-scale morphologic tracking approach for edge detection. In *Proceedings of the 4th International Conference on Image and Graphics*. Chengdu (China), 2007, p. 358-362.
- [4] BAI, X., ZHOU, F. Edge detection based on mathematical morphology and iterative thresholding. In *Proceedings of International Conference on Computational Intelligence and Security*. Guangzhou (China), 2006, p. 1849-1852.
- [5] NALWA, V. S., BINFORD, T. O. On detecting edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986, vol. 8, no. 6, p. 699-714.
- [6] BOUCHARA, F., BERTRAND, M., RAMDANI, S., HAYDAR, M. Sub-pixel edge fitting using B-spline. In *Proceedings of the 3rd International Conference on Computer Vision*. Berlin (Germany), 2007, p. 353-364.
- [7] KISWORO, M., VENKATESH, S., WEST, G. Modeling edges at subpixel accuracy using the local energy approach. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 1994, vol. 16, no. 4, p. 405-410.
- [8] HERMOSILLA, T., BERMEJO, E., BALAGUER, A., RUIZ, L.A. Non-linear fourth-order image interpolation for subpixel edge detection and localization. *Image and Vision Computing*, 2008, vol. 26, no. 9, p. 1240-1248.
- [9] PAP, L., ZOU, J. J. Sub-pixel edge detection for photogrammetry using laplace difference of Gaussian and 4th order ENO interpolation. In *Proceedings of IEEE 17th International Conference on Image Processing*. Hong Kong, 2010, p. 2841-2844.
- [10] TABATABAI, A. J., MITCHELL, O. R. Edge location to subpixel values in digital imagery. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 1984, vol. 6, no. 2, p. 188-201.
- [11] DA, F. P., ZHANG, H. Sub-pixel edge detection based on an improved moment. *Image and Vision Computing*, 2010, vol. 28, no. 12, p. 1645-1658.
- [12] LYVERS, E. P., MITCHELL, O. R., AKEY, M. L., REEVS, A. P. Subpixel measurements using a moment-based edge operator. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 1989, vol. 11, no. 12, p. 1293-1309.
- [13] LI, B., CUI, Y. Study on edge subpixel location of ellipse in computer vision measurement. In *Proceedings of 3rd International Congress on Image and Signal Processing*. Yantai (China), 2010, p. 1684-1688.
- [14] BIN, T. J., LEI, A., CUI, J. W., KANG, W. J., LIU, D. D. Subpixel

edge location based on orthogonal Fourier-Mellin moments. *Image and Vision Computing*, 2008, vol. 26, no. 9, pp. 563-569.

- [15] LI, J. Q., WANG, J. W., CHEN, S. B., WU, L. Improved algorithm of subpixel edge detection using Zernike orthogonal moments. *Optical Technique*, 2003, vol. 29, no. 4, p. 500-504.
- [16] ZHANG, B., BAI, L., ZENG, X. A novel subpixel edge detection based on the Zernike moment. *Information Technology Journal*, 2010, vol. 9, no. 1, p. 41-47.
- [17] ZHANG, W., BERGHOLM, F. Multi-scale blur estimation and edge type classification for scene analysis. *International Journal of Computer Vision*, 1997, vol. 24, no. 3, p. 219 – 250.
- [18] WEISSTEIN, E. W. *Erf*. [Online] Cited 2011-02-21. Available at: <http://mathworld.wolfram.com/Erf.html>
- [19] LAGARIAS, J. C., REEDS, J. A., WRIGHT, M. H, WRIGHT, P. E. Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM Journal of Optimization*, 1998, vol. 9, no. 1, p. 112-147.
- [20] BABA, M., OHTANI, K. A novel subpixel edge detection system for dimension measurement and object localization using an analogue-based approach. *Journal of Optics A: Pure and Applied Optics*, 2001, vol. 3, p. 276-283.
- [21] HUSSMANN, S., HO, T. H. A high-speed subpixel edge detector implementation inside a FPGA. *Real-Time Imaging*, 2003, vol. 9, no. 5, p. 361-368.
- [22] HAGARA, M. Sub-pixel edge detection with wavelets. In *Conf. Proc. of Radioelektronika 2006*. Bratislava, 2006, p. 353-356.

About Authors ...

Miroslav HAGARA was born in 1961 in Handlová, Czech and Slovak Republic. He received the MSc. degree in Radio Electronics from the Slovak University of Technology in Bratislava, in 1986. Since 1992, he is an assistant and lecturer at the Department of Radio and Electronics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava. His research interests include digital image processing, edge detection in digital images.

Peter KULLA (Fellow of IET) was born in 1949 in Šumiac, Czech and Slovak Republic. He received the MSc. Degree in Electrical Engineering from the FEE of Czech TU Prague in 1973 and the Ph.D. degree in Radio and Electronics from FEE of Slovak TU in Bratislava, 1980. From 1984 he is the Associate professor at the Department of Radio and Electronics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava. His lifelong scientific research and education interests are focused on Television engineering, Applications of image sensors CCD, Digital Circuits, Digital image processing and encoding, Analogue and Digital Television.