A novel image edge detection algorithm based on neutrosophic set ☆

Yanhui Guo [a,*], Abdulkadir Şengür [b]

[a] School of Science, Saint Thomas University, Miami Gardens, FL 33054, USA
[b] Firat University, Technology Faculty, Department of Electrical and Electronics Engineering, Elazig, Turkey

ARTICLE INFO

ABSTRACT

Neutrosophic set (NS) theory is a formal framework to study the origin, nature, and scope of the neutral state. In this paper, neutrosophic set is applied in image domain and a new directional α-mean operation is defined. Based on this operation, neutrosophic set is applied into image edge detection procedure. First, the image is transformed into NS domain, which is described by three membership sets: T, I and F. Then, a directional α-mean operation is employed to reduce the indeterminacy of the image. Finally, a neutrosophic edge detection algorithm (NSED) is proposed based on the neutrosophic set and its operation to detect edge. Experiments have been conducted using numerous artificial and real images. The results demonstrate the NSED can detect the edges effectively and accurately. Particularly, it can remove the noise effect and detect the edges on both the noise-free images and the images with different levels of noises.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Image edge is the foundation of image texture and shape, which contains a great deal of important information and plays a critical role in images. Edge is extensively used to separate the object from the background in image processing.

Edge detection is one of the important techniques in image processing and analysis, and it is also the foundation to object recognition, computer vision, motion analysis, scene analysis, etc. [1]. Edge detection influences all results of the image analysis directly, and its accuracy is a crucial factor in determining the overall performance. The detection results benefit many applications such as image enhancement, recognition, morphing, compression, retrieval, watermarking, hiding, restoration and registration [1]. High-level processing tasks such as image segmentation and object recognition directly depend on the quality of the edge detection procedure. Therefore, how to find a better algorithm to detect image edge is one of the key techniques in image processing and analysis, and it has received much attention during the past two decades because of its significant importance in many research areas.

Image edge detection technique is also one of the most difficult tasks in image processing, and numerous publications have been proposed to study image edge detection. So far, numerous edge detection methods have been developed, and these techniques can be categorized into four different groups: gradient-based, multi-scale, neural network, and fuzzy theory methods.

1.1. Gradient based approaches

Edges can be defined as a discontinuity or change in intensity of an image [2]. For pixels, edges can be viewed as the region in which the image intensities take a sharp variation. The common approach is to compute the gradient of the image and then find the local maximum (or zero-crossings) to detect edges [3]. The gradient-based edge detection methods, like those of Canny, Sobel, Robert, Prewitt, etc., obtain the gradient to find edges [1]. Since gradient-based methods are very sensitive to noise, their performance is limited. Especially when the images are highly corrupted, the errors in the edge results may be very large. In addition, most of these methods require a predetermined threshold for determining whether or not a zero-crossing point is an edge point. The threshold value is usually obtained through trial and error that results in poor efficiency [2].

1.2. Multi-scale methods

The edges are also interpreted as one class of singularities in images. The multi-scale theory is proved to be a powerful tool to detect signal singularities and has been applied into image edge detection. Bao et al. [4] combined the Canny edge detection with a multiplication technique. A scale multiplication function was defined as the product of the responses of the detection filter at two scales. The edge map was obtained as the local maxima by thresholding the scale multiplication results. Tang et al. [5] presented an edge detection algorithm on X-ray image based on the multi-scale and multi-resolution of wavelet transform. The local maxima in vertical, horizontal and diagonal directions were detected by quadratic B-spline. Mallat algorithm was used in wavelet decomposition to determine defect edges. Shih and Tseng [6] combined a gradient-based edge detection and a wavelet-based multiscale edge tracking scheme to extract edges. The wavelet transformation decomposed an image into different scale and different frequency subbands, and multiscale shift-invariant gradient images were made from the high-frequency subbands. A contextual-filter edge detector detected edges from the finest-scale gradient images, and the edge tracker refined the detected edges on the multiscale gradient images.

For multiscale methods, a large number of edge details and high accuracy edge orientation are obtained in small scales; however they are often interfered with by noise. With large scale, it can get stable edges and better noiseproof feature while the lower edge orientation. Hence, the multi-resolution edge detection method has a trade-off between localization and edge details. A fine resolution gives too much redundant detail, whereas a coarse resolution lacks accuracy of edge detection.

1.3. Neural network techniques

In neural network methods, edge detection is treated as a classification process. Chang [2] presented a contextual Hopfield neural network (CHNN) to detect the edges of CT and MRI images. The CHNN maps the two-dimensional Hopfield network at the original image plane. Using the mapping results, the network incorporates pixels' contextual information into edge detection procedure. Chacon-M et al. [7] proposed an edge detection approach based on analysis of the information provided by the time matrix generated from a pulse coupled neural network (PCNN). Two different schemes are employed for edge detection. The first scheme was developed to generate edges from coarse images and the second one to deal with more detailed edges. Emin Yüksel [8] presented a neuro-fuzzy (NF) operator for edge detection in images by combining a desired number of neuro-fuzzy (NF) subdetectors with a postprocessor. In the proposed method, the edges were directly determined by a NF network.

1.4. Fuzzy theory based approaches

The common premise conditions of some edge detection methods are that the edges of an image must be clear. However, the image in reality is fuzzy and the edges are not clear. Fuzzy theory gives a mechanism to represent ambiguity within an image. Each pixel of an image has a degree of belonging to a region or a boundary. For its powerful ability to deal with the fuzziness of edges, fuzzy theory has been applied into edge detection. Verma et al. [9] proposed an edge detection technique for the noisy image using fuzzy derivative and bacterial foraging algorithm. The bacteria detect edge pixels and noisy pixels in its foraging paths. The direction of movement of each bacterium was found using the fuzzy inference rules.

There are some other methods to detect image edges, such as entropy-based and fractal-based methods. Entropy is employed to describe the characteristic of a pixel and its neighbor, and it has been employed to detect image edges. Singh and Singh [10] utilized Shannon entropy to detect edges in gray level images. A suitable threshold value was employed to segment the image and achieve the binary image. The edge detector was to detect and locate the edges in the binary image.

Bhardwaj and Mann [11] proposed an edge detection method based on Adaptive Neuro-Fuzzy Inference System (ANFIS). A first order Sugeno-type fuzzy inference system was built with 4-inputs and 1 output. The ANFIS was then trained with the edge patterns. The obtained results were better than the Sobel and Roberts edge detectors. Zhang et al. introduced the contourlet transform into the gradient vector flow (GVF) snake model to solve the initial active contour problem of the snake algorithm [12]. The improved scheme improved the edge detection results of the GVF snake model. Zhou et al. presented a multiscale gradient multiplication-based thresholding method for edge detection [13]. The bilevel thresholding with different histogram patterns was employed to find edges on the images.

In summary, the existing edge detection methods have different merits and demerits at the same time. Table 1 is employed to summarize their advantages and disadvantages.

Table 1
Summary of edge detection methods.

Methods	Advantages	Disadvantages
Gradient based approaches	Simply and fast	Sensitive to noise
Multiscale methods	Detect edges in different scales with high localization ability	In dilemma to select scales
Neural network techniques	Employ prior knowledge	Performance depends on training process which is time consuming
Fuzzy theory-based approaches	Ability to handle unclear edge	Fuzzy rules are hard to find and describe

Generally, these classic edge detectors work well with high-quality images, but some of them are not good enough for noisy images. Edge detection for noisy images is more important because noise is common in images. Most gradient-based edge detection algorithms are sensitive to image noise, which leads to low effectiveness of edge detection [2]. Our motivation of this paper is to overcome the existing edge detection methods' drawbacks on the noisy image. To achieve this goal, an edge detection method is developed based on neutrosophic set theory that aims at achieving the preservation of image edge details and reduction of false edges caused by noise. At first, the image is transformed into the neutrosophic set (NS) and a directional α-mean operation is employed to reduce the indeterminacy degree of the image. In the edge detection process, the gradient value is computed and updated according to the directional α-mean operation result. The directional α-mean and edge detection are taken iteratively. Finally, the edges are obtained in NS domain. As a result, the effect of noise will be effectively removed by the proposed method and the drawback of disconnected regions can be overcome. The proposed edge detection method will be tested on popular images having different image properties and also compared with popular edge detectors from the literature. Its advantage is the ability to detect edges efficiently in clean or noisy images. Experimental results show the proposed edge detector exhibits much better performance than the competing operators and may efficiently be used for the detection of edges in images with or without noise. The proposed method can obtain more appropriate and continued edges than other current methods in either clean or noisy images.

The paper is organized as follows. In Section 2, the neutrosophic set theory and its operation are described, and the proposed edge detection approach is presented in Section 3. The experiments and comparisons are discussed in Section 4. Finally, the conclusions are given in Section 5.

2. Neutrosophic set theory

2.1. Neutrosophic set

For the classic set, the indeterminacy of each element in the set could not be evaluated and easily described. The fuzzy set has been applied to many real applications to handle uncertainty. The traditional fuzzy set uses a real number $\mu(x)$ ($0 \leqslant \mu(x) \leqslant 1$, $x \in A$) to represent the membership in the set A defined on universe R. Sometimes $\mu(x)$ itself is uncertain and is difficult to be defined by a crisp value [14]. Some applications could not only consider the true membership, but also the false membership and the indeterminacy of the membership. For instance, in the application of image processing, an image might have a few regions and pixels, such as noise, shadow and boundary, which have high indeterminacy value. It is difficult to solve these problems using the classic fuzzy set [14].

Neutrosophy is regarded as a generalization of dialectics, and studies the origin, nature, and scope of neutralities. It considers a proposition, theory, event, concept, or entity <A> in relation to its opposite <Anti-A> and the neutrality <Neut-A>, which is neither <A> nor <Anti-A>. Neutrosophy is the basis of neutrosophic logic, neutrosophic probability, neutrosophic sets, and neutrosophic statistics.

Neutrosophy set theory provides a powerful tool to deal with the indeterminacy, and the indeterminacy is quantitatively described using a membership. In neutrosophic set, a set A is described by three subsets: <A>, <Neut-A> and <Anti-A>, which is interpreted as truth, indeterminacy, and false set. Neutrosophic set provides a new tool to describe the image with uncertain information, which had been applied to image processing techniques, such as image segmentation, thresholding and denoise. Guo and Cheng [15] combined neutrosophic set with K-mean clustering method for image segmentation. The image was segmented based on neutrosophic set operations and clustering results. Guo and Cheng [16] applied the neutrosophic set into image domain and defined concepts and operators for image denoising. It processed not only noisy images with different levels of noise, but also images with different kinds of noise without knowing the type of the noise. Sengur and Guo [17] employed the wavelet transform in the neutrosophic set for color texture image segmentation.

2.2. Neutrosophic image

An image might have a few indeterminate regions and pixels, such as noise, shadow and boundary. The classic set is hard to interpret the indeterminate region clearly. In neutrosophic set theory, a new subset, indeterminate set I is proposed to interpret the indeterminacy in the image. Using the indeterminate set, it can describe the indeterminate regions easily.

An image is transformed into neutrosophic set and a neutrosophic image was defined in [15–17]. A neutrosophic image Im_{NS} is described by three membership sets T, I and F. The pixel $P(i,j)$ in the image domain is transformed into the neutrosophic set domain, denoted as $P_{NS}(i,j)$ $(P_{NS}(i,j) = \{T(i,j), I(i,j), F(i,j)\})$. $T(i,j)$, $I(i,j)$ and $F(i,j)$ are the membership values belonging to the bright pixel set, indeterminate set and non-bright pixel set, respectively, which are defined as follows [15–17]:

$$T(i,j) = \frac{\bar{g}(i,j) - \bar{g}_{min}}{\bar{g}_{max} - \bar{g}_{min}} \tag{1}$$

$$\bar{g}(i,j) = \frac{1}{w \times w} \sum_{m=i-w/2}^{i+w/2} \sum_{n=j-w/2}^{j+w/2} g(m,n) \tag{2}$$

$$I(i,j) = \frac{\delta(i,j) - \delta_{min}}{\delta_{max} - \delta_{min}} \tag{3}$$

$$\delta(i,j) = abs(g(i,j) - \bar{g}(i,j)) \tag{4}$$

$$F(i,j) = 1 - T(i,j) \tag{5}$$

where $\bar{g}(i,j)$ is the local mean value, and $\delta(i,j)$ is the absolute value of the difference between intensity $g(i,j)$ and its local mean value at (i,j). The value of I measures the indeterminacy degree of P_{NS}. For T and F are correlated with I, the changes in T and F influence the distribution of the elements in I.

2.3. Directional α-mean filtering operation

In [18], an α-mean filtering operation was defined on neutrosophic image, and it removed noise efficiently. However, it made the image blur and reduced the contrast, which could make the detected edges inaccurate. To overcome this drawback, a directional α-mean operation (denoted as DAM) is newly proposed to remove the noise effect and preserve the edges at the same time.

The DAM is described as follows: First, the direction of pixels in an image is discussed. In [18], a pixel's direction was determined by the neighbors' information. The idea is adopted to define the pixel's direction. In Fig. 1, the pixel's directions in (a), (b) and (c) are called all-directional, horizontal, and vertical, respectively. The idea to determine the pixel's direction is as follows: if the value of the horizontal gradient is higher than the value of the vertical, the pixel's direction is horizontal; if the value of the vertical gradient is higher than the value of the horizontal, the pixel's direction is regarded as vertical; otherwise, it is all-directional.

Therefore, the directional mean operation has three masks according to the pixel's directions [18].

$$M_1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad M_2 = \frac{1}{3} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad M_3 = \frac{1}{3} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{6}$$

$$R_1 = cov(M_1, Im) \quad R_2 = cov(M_2, Im) \quad R_3 = cov(M_3, Im) \tag{7}$$

where M_1, M_2 and M_3 are the masks with size 3×3 to process all-directional, horizontal, and vertical pixels, respectively. R_1, R_2 and R_3 are the filtering results when the image Im is convoluted with M_1, M_2 and M_3, and $cov(\cdot)$ is the convolution function.
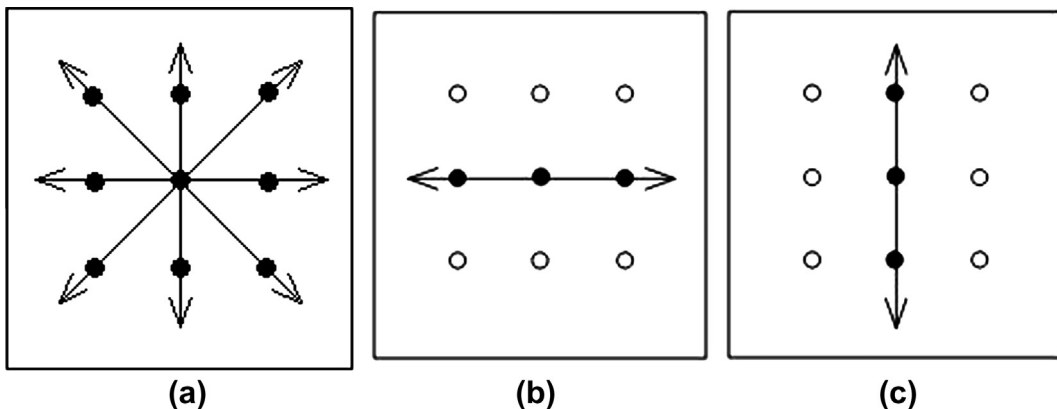

Fig. 1. Pixel's directions.

The function of the directional mean *DMF* is defined as:

$$DMF(Im(i,j)) = \begin{cases} R_1 & |Gh(i,j) - Gv(i,j)| \leq \xi \\ R_2 & Gh(i,j) - Gv(i,j) > \xi \\ R_3 & Gv(i,j) - Gh(i,j) > \xi \end{cases} \tag{8}$$

where $Gh(i,j)$ and $Gv(i,j)$ are the norm of the gradient at (i,j) on the image *Im*. ξ is a small constant.

Finally, the directional α-mean operation is defined according to the pixel's direction in the neutrosophic image:

$$\overline{P}_d(\alpha) = \{\overline{T}_d(\alpha), \overline{I}_d(\alpha), F\} \tag{9}$$

$$\overline{T}_d(\alpha) = \begin{cases} T & I < \alpha \\ DMF(T) & I \geqslant \alpha \end{cases} \tag{10}$$

$$\overline{I}_d(\alpha) = \frac{\overline{\delta}_T - \overline{\delta}_{T\min}}{\overline{\delta}_{T\max} - \overline{\delta}_{T\min}} \tag{11}$$

$$\overline{\delta}_T(i,j) = abs(\overline{T}_d(i,j) - \overline{\overline{T}}_d(i,j)) \tag{12}$$

$$\overline{\overline{T}}_d(i,j) = \frac{1}{w \times w} \sum_{m=i-w/2}^{i+w/2} \sum_{n=j-w/2}^{j+w/2} \overline{T}_d(m,n) \tag{13}$$

where $\overline{P}_d(\alpha)$ is the value after the directional α-mean operation, and $\overline{\delta}_T(i,j)$ is the absolute value of the difference between the mean intensity $\overline{T}_d(i,j)$ and its local mean value $\overline{\overline{T}}_d(i,j)$. In the proposed method, *F* is not processed using DAM and it keeps unchanged. The directional α-mean operation can be regarded as a simplified anisotropic filtering process, in which the strength of the smoothing is controlled by the value of α. This operation is faster than the standard formulation, while it can also enhance the edge information, which is suitable for edge detection.

3. Proposed method

A new edge detection algorithm, neutrosophic set edge detection (NSED) is proposed. In NSED, an iterative directional α-mean process is employed to make the detection more accurate, and the termination criterion for the process is defined accordingly.

The gradients of the pixels in $\overline{T}_d(\alpha)$ are used to evaluate the degree of the pixels belonging to edge pixels, and a threshold value of gradient is selected to determine whether the pixels are edge pixels.

$$E_{NS}(X) = \begin{cases} 1 & \|\nabla(\overline{T}_d(\alpha))\| \geqslant \nabla_{th} \\ 0 & \|\nabla(\overline{T}_d(\alpha))\| < \nabla_{th} \end{cases} \tag{14}$$

where $\|\nabla(\overline{T}_d(\alpha))\|$ is the norm of the gradient on $\overline{T}_d(\alpha)$, and ∇_{th} is the threshold value to determine the edge pixels.

The DAM operation and gradient computation is taken iteratively. If extracted edges remain unchanged, the iteration procedure will be terminated. The DAM can be regarded as a low frequency filter. After the iterative processing, the image becomes homogeneity. In this situation, the number of the detected edge pixels becomes constant, which makes the edge detection result stable.

Based on the above definitions and operations, NSED is designed by executing the directional α-mean operation and edge detection iteratively. At first, the image is transformed into the neutrosophic set domain. Then, the indeterminacy of the neutrosophic set is decreased using the directional α-mean filtering operation and the edge is preserved and enhanced. The detection process is terminated when the detected edges become unchanged. The whole procedure is summarized as follows:

(1) Input: *Im*, $t = 0$.
(2) Transform image to *NS* which results in $Im_{NS}^{(0)} = \{T^{(0)}\}$.
(3) Perform the DAM operation on $Im_{NS}^{(0)}$ which results in $Im_d^{(0)}$.
(4) Compute $E^{(0)}$.
(5) $t = t + 1$.
(6) Perform DAM operation on $Im_d^{(t-1)}$ which results in $Im_d^{(t)}$.
(7) Compute $E^{(t)}$
   If $|E^{(t)} - E^{(t-1)}| \geqslant \varepsilon$, go to 5).
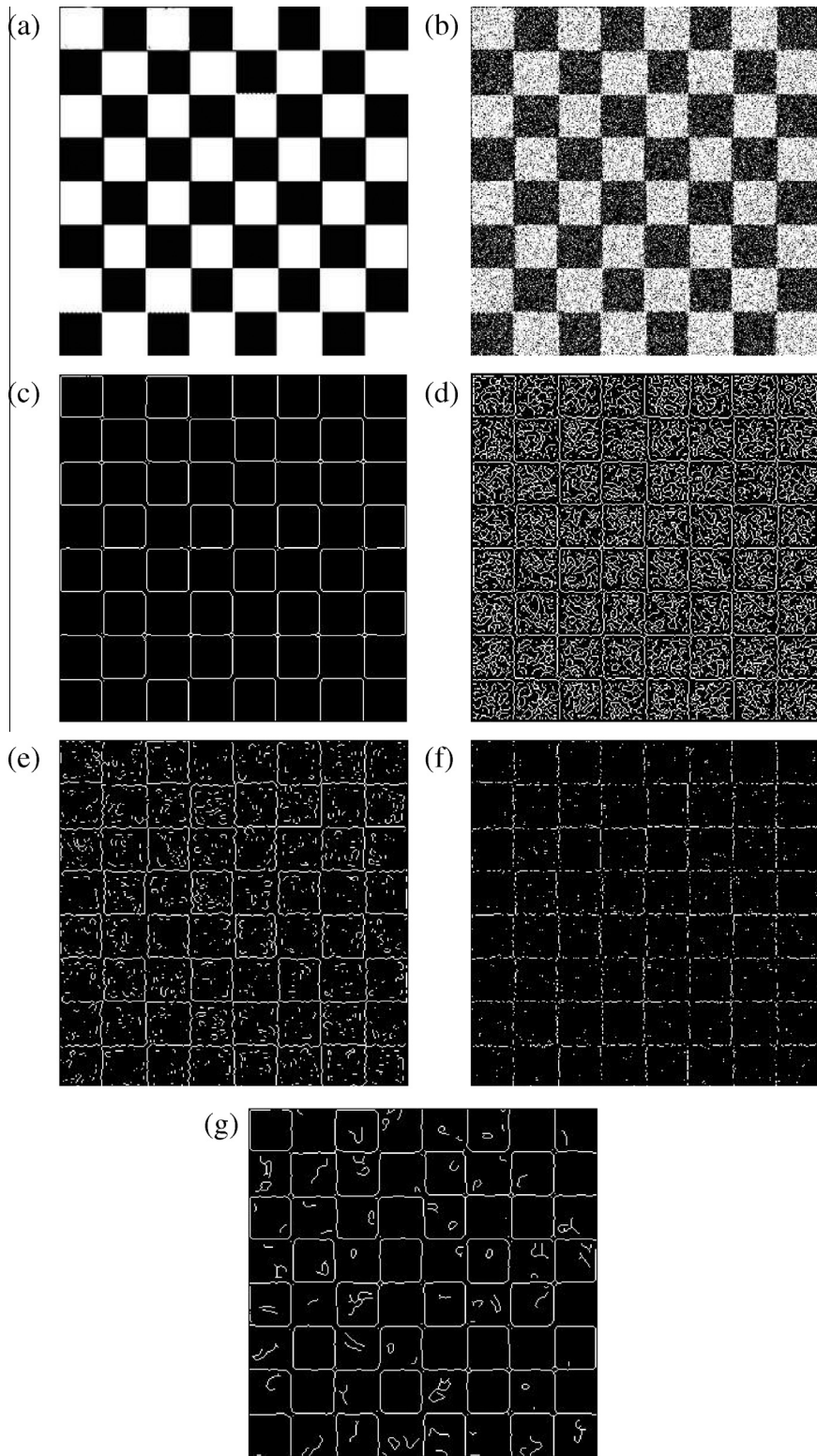(8) Return $E^{(t-1)}$ as the image edge.

Fig. 2. (a) The original chessboard image (b) the image with Gaussian noise (c) the ground truth of edge (d) the edge detection result by SMED (e) the edge detection result by RROED (f) the edge detection result by BFED (g) the edge detection result by NSED.

4. Experiments and discussions

To test the performance of the proposed method, a variety of artificial images and natural images with different noise levels are employed. The NSED method is compared with that of the recently proposed scale multiplication edge detector (denoted as SMED) by Bao et al. [4], which claimed it has significantly improved the performance of the traditional Canny edge detector [3] and achieves better results than the anisotropic diffusion edge detector (ADED) by Black et al. [19]. In addition, the NSED method is also compared with another two edge detector methods based on robust rank-order test (denoted as RROED) by Lim [20] and the bacterial foraging scheme (denoted as BFED) by Verma et al. [9], which claimed their performances are robust to variations in noise on edge detection.

In the NSED method, three parameters, the local window size w in Eq. (2), the α value in Eq. (10) and gradient threshold value ∇_{th} in Eq. (15) are optimized using experiments, which are taken on the artificial images with different noise levels. These parameters are tuned on the training image by doing a local discrete grid search routine with a fixed step-size on the parameter space and in the feasible ranges of parameter values namely: $w \in [3,5,7]$, $\alpha \in [0 1.0]$ and $\nabla_{th} \in [0.001 0.9]$. The grid search routine is a standard optimization algorithm which consists, for several initial guesses of the parameters to be estimated, in employing a moving 3-dimensional (w, α, ∇_{th}) grid. The algorithm tries to center the grid on the best performance score for parameters, moving in an appropriate direction at each iteration. The optimization is successful when the grid becomes centered on the best performance score across all dimensions. The optimal parameters are obtained as $w = 3$, $\alpha = 0.15$ and $\nabla_{th} = 0.04$.

The metric figure of merit *FOM* proposed by Pratt [21] is used as the performance score, which is defined as:

$$FOM = \frac{1}{max(N_I, N_A)} \sum_{k=1}^{N_A} \frac{1}{1 + \beta d^2(k)} \qquad (15)$$

where N_I and N_A are the numbers of the detected edge points and the actual edge points, respectively. $d(k)$ is the distance from the *k*th actual edge point to the nearest detected edge point. β is a scaling constant set to 1/9 in Pratt's paper. The greater the *FOM*, the better the detection results.

4.1. Experiments on artificial images

In this section, several synthetic images are used to compare the performance of the proposed NSED with those of the SMED, RROED and BFED methods. The optimal parameters ($w = 3$, $\alpha = 0.15$ and $\nabla_{th} = 0.04$) in NSED are employed in the following experiments. These parameters for SMED, PROED and BFED are optimized according to the guidelines in their papers. For SMED, the scale parameters are $s_1 = 2^2$ and $s_2 = 2^3$, and the threshold constant is $c = 6$. In RROED method, 3×3 difference-of-boxes for the Wilcoxon and T detectors are used. The threshold for these detectors are set at a significance level $\alpha = 0.05$. In BFED, the parameters are selected the default optimized values in [9]:

Table 2
Edge detection comparisons of SMED and NSED.

Standard deviation	SNR (dB)	SMED		RROED		BFED		NSED	
		E	FOM	E	FOM	E	FOM	E	FOM
20	21.7754	0.0503	0.8774	0.0191	0.9163	0.0389	0.9507	0.0116	0.9861
25	19.8799	0.0534	0.8512	0.0199	0.8961	0.0392	0.9501	0.0136	0.9798
30	18.3461	0.0589	0.8026	0.0203	0.8741	0.0402	0.9512	0.0142	0.9780
35	16.9836	0.0686	0.7267	0.0211	0.8227	0.0395	0.9479	0.0156	0.9605
40	15.8179	0.0779	0.6465	0.0299	0.8015	0.0407	0.9401	0.0207	0.9551
45	14.7129	0.0943	0.5685	0.0306	0.7849	0.0425	0.9212	0.0202	0.9453
50	13.8486	0.1079	0.5141	0.0417	0.7163	0.0427	0.9119	0.0202	0.9404
55	13.0344	0.1249	0.4655	0.0527	0.6822	0.0439	0.9070	0.0190	0.9364
60	12.3123	0.1376	0.4298	0.0629	0.6123	0.0466	0.8902	0.0189	0.9343
65	11.6531	0.1518	0.4076	0.0723	0.5929	0.0473	0.8727	0.0206	0.9261
70	10.9455	0.1606	0.3929	0.0854	0.5256	0.0494	0.8555	0.0269	0.8940
75	10.4151	0.1731	0.3775	0.0966	0.4831	0.0506	0.8341	0.0288	0.8808
80	9.8376	0.1807	0.3677	0.1089	0.4126	0.0542	0.8011	0.0252	0.8748
85	9.3766	0.1881	0.3595	0.1184	0.3924	0.0563	0.7790	0.027	0.8632
90	8.9076	0.1950	0.3541	0.1292	0.3620	0.0624	0.7466	0.0292	0.8549
95	8.4628	0.2022	0.3480	0.1311	0.3511	0.0637	0.7257	0.0303	0.8478
100	8.0139	0.2075	0.3440	0.1412	0.3483	0.0702	0.6823	0.0294	0.8444
105	7.6421	0.2157	0.3373	0.1534	0.3401	0.0724	0.6536	0.0330	0.8365
110	7.2635	0.2204	0.3372	0.1639	0.3394	0.0783	0.6218	0.0398	0.7721
115	6.9123	0.2272	0.3330	0.1758	0.3340	0.0847	0.5953	0.0405	0.7667
Average	12.3070	0.1448	0.4920	0.0837	0.5794	0.0532	0.8269	**0.0242**	**0.8989**

The bold values display the best measurements in the all evaluation results.

- Total number of bacteria N_b = 3500 approx.
- Step size (c) = 1.
- Number of chemotactic steps N_c = 16.
- Number of elimination–dispersion events N_{ed} = 4.
- Bacteria split ratio = 2:1.
- Threshold value (α) = 0.7.
- Threshold value for deciding number of bacteria to eliminate/reproduce (β) = 30.

Fig. 2(a) is an artificial image *chessboard*, and Fig. 2(b) is a noisy *chessboard* image having Gaussian noise, whose mean is 0 and standard deviation is 115. Fig. 2(c) is the ground truth of edge. Fig. 2(d–g) are the edge detection results generated by SMED, RROED, BFED and NSED, respectively. It is seen that SMED and RROED do not suppress noise well on this artificial noisy image. Many pixels are wrongly detected as edge in Fig. 2(d and e), while they are detected correctly in Fig. 2(g) by NSED. BFED has better performance than SMED and RROED, while it is worse than NSED in Fig. 2(g). The edges of small blocks in Fig. 2(g) are more consistent and distinct, which are better for further processing, such as object extraction and detection. The result in Fig. 2(g) removes most of the false edges and achieves very high edge localization performance. The shapes of the small blocks are precisely detected. It shows clearly that the NSED can perform better than other methods for detecting edges on artificial images with noise.

To evaluate the performance of the edge detection algorithms quantitatively, two objective criteria are employed to evaluate the results of the algorithms.

To compare the edge detection results of the real edges on original images, a measurement edge detection error E by Ma and Staunton [22] is employed, which is defined as:

$$E = 1 - \frac{\#(E_o \cap E_n)}{\#(E_o)} \tag{16}$$

where E_o is the ideal edge result, and E_n is the real edge detection result by the method. $\#(\cdot)$ is the number of elements in a set. The smaller the E, the better the detection results.

The *FOM* and E provide quantitative measurements of the performance between the ideal edges and the real results by the algorithms. The quality of the tested image can be described in terms of signal-to-noise ratio (SNR):
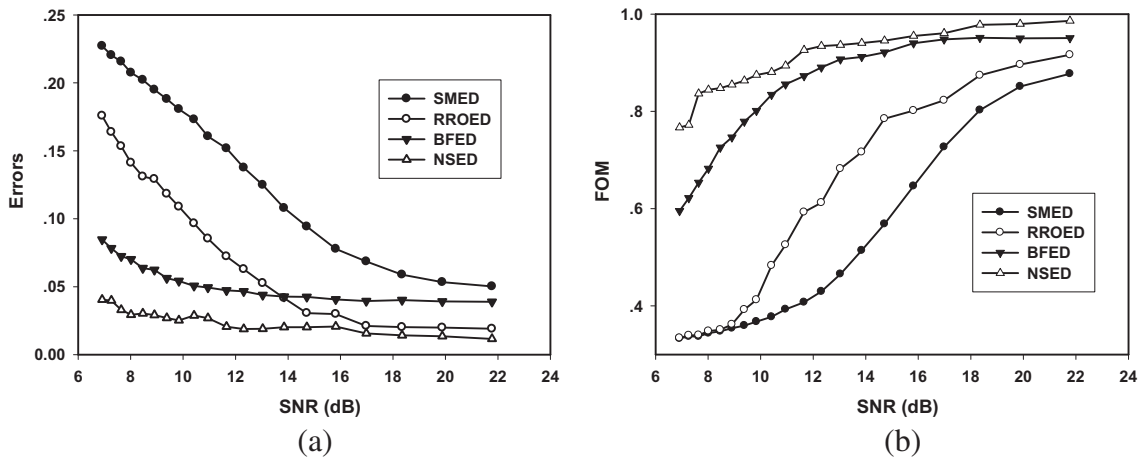
Fig. 3. Edge detection performance comparison on artificial image (a) E comparison (b) *FOM* comparison.

Table 3
Statistical comparison on artificial image using E and *FOM*.

Metrics	p-Value		
	SMED vs. NSED	RROED vs. NSED	BFED vs. NSED
E	2.3815×10^{-9}	6.95079×10^{-6}	7.24513×10^{-14}
FOM	2.5378×10^{-11}	1.11519×10^{-8}	9.14105×10^{-6}

$$SNR = 10\log_{10}\left[\frac{\sum_{r=0}^{H-1}\sum_{c=0}^{W-1}I_o^2(r,c)}{\sum_{r=0}^{H-1}\sum_{c=0}^{W-1}(I_o(r,c)-I_n(r,c))^2}\right] \tag{17}$$

where $I_o(r,c)$ and $I_n(r,c)$ represent the intensities of pixel (r,c) in the original and noisy images, respectively.

In the experiments, the *chessboard* image is employed as a noise-free image and the Gaussian noises with different standard deviations are added onto it. The images with Gaussian noises are employed to test the SMED, RROED, BFED and NSED. Table 2 lists the values of E and *FOM* for their detection results on artificial images at different SNR levels and the comparisons are plotted in Fig. 3.

From Table 2 and Fig. 3, it is clear the NSED method achieves better performance and lower detection error at all SNR levels. The detection errors of NSED method are all smaller than 0.0405, and the errors of SMED, RROED and BFED are all bigger than those of NSED. When the SNR is low, the proposed method performs much better than other methods. The NSED approach can obtain the optimum detection result with error rate 0.0405 which is very low when SNR is 6.9123 dB, while the error of SMED approach reaches 0.2272. Meanwhile, the values of *FOM* of NSED are all bigger than those of SMED, RROED and BFED at all SNR levels. The average value of *FOM* of NSED, SMED, RROED and BFED are 0.8989, 0.4920, 0.5794, and 0.8269, respectively.

The performance of comparison is tested on each metric using the paired t-test. The testing results, p-values, are listed in Table 3. All p-values are less than 0.05. The statistical analysis demonstrates the improvement is statistically significant comparing the NSED with SMED, RROED and BFED.

4.2. Experiments on real images

To validate the results, several different kinds of test images are considered to examine the versatility of the proposed edge detector. In this section, four real images, *Rice*, *House*, *Eight* and *Alumigns*, are used to test the performance of the proposed method. These images were selected based on the criteria they have either with clear boundary on the objects or plenty of texture in the region. The results on these images can show the edge detection performance in different boundary situations.
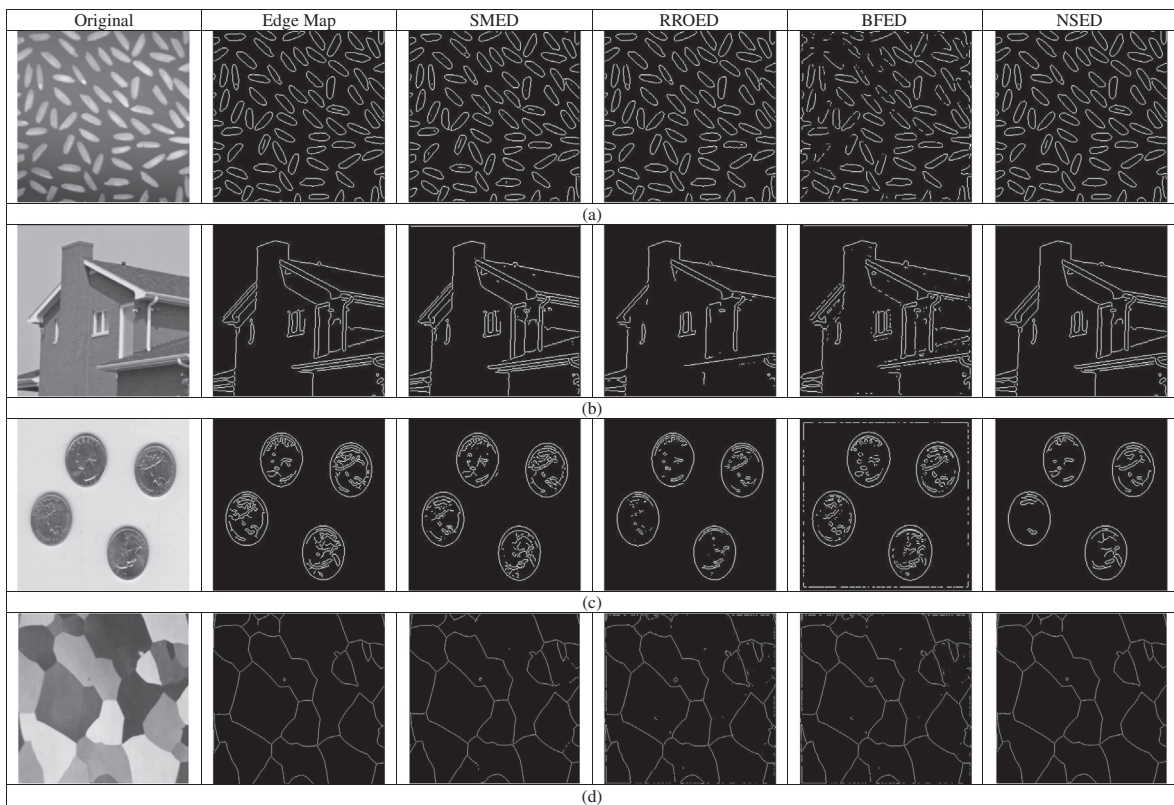
Fig. 4. Edge detection result comparisons on four original images (a) results on *Rice* image (b) results on *House* image (c) results on *Eight* image (d) results on *Alumgrns* image.

Noisy images	SMED	RROED	BFED	NSED

(a)

(b)

(c)

(d)
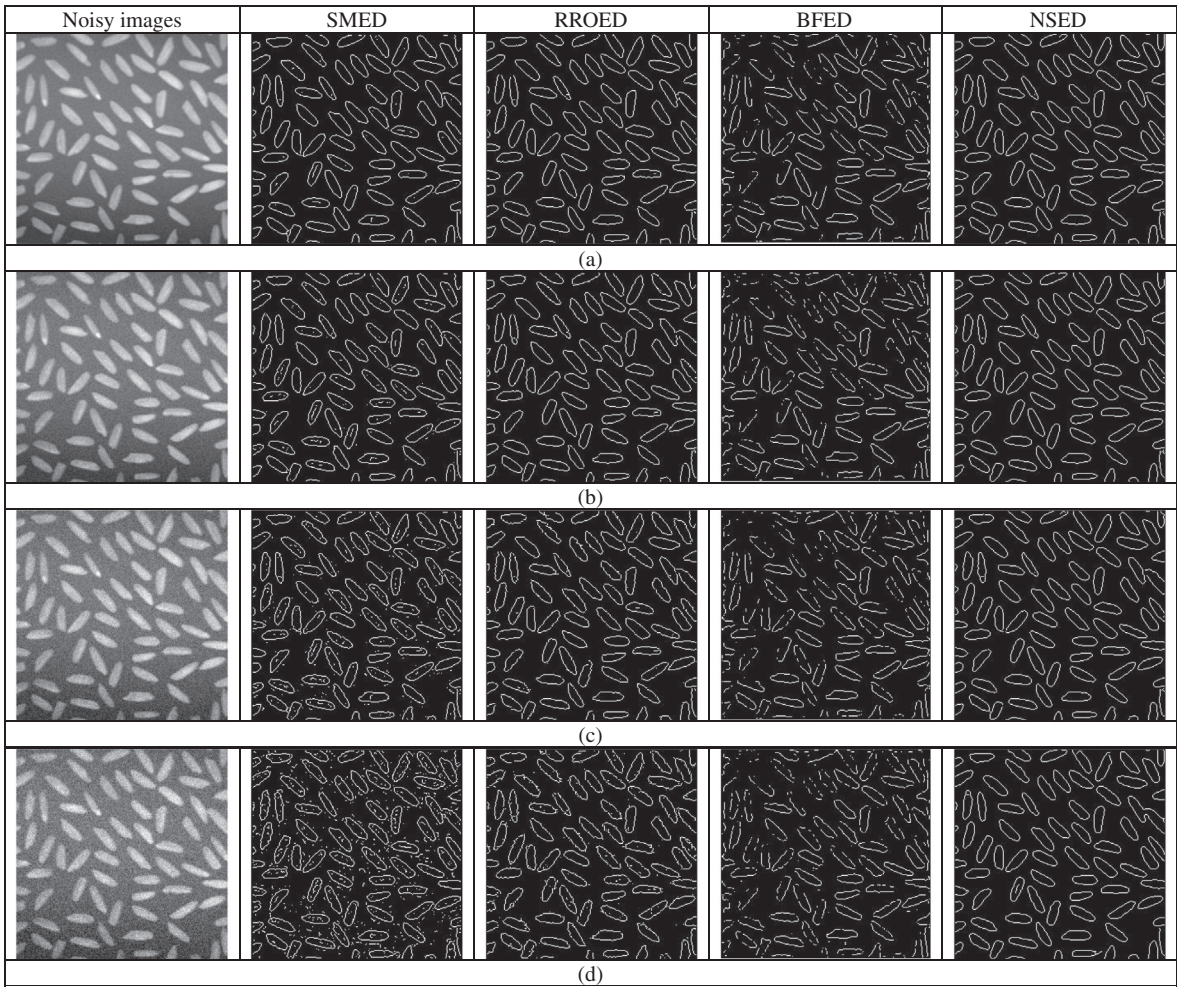
Fig. 5. Edge detection result comparison on *Rice* image with different Gaussian noise levels (a) results on *Rice* image with Gaussian noise (σ = 5) (b) results on *Rice* image with Gaussian noise (σ = 10) (c) results on *Rice* image with Gaussian noise (σ = 15) (d) results on *Rice* image with Gaussian noise (σ = 20).
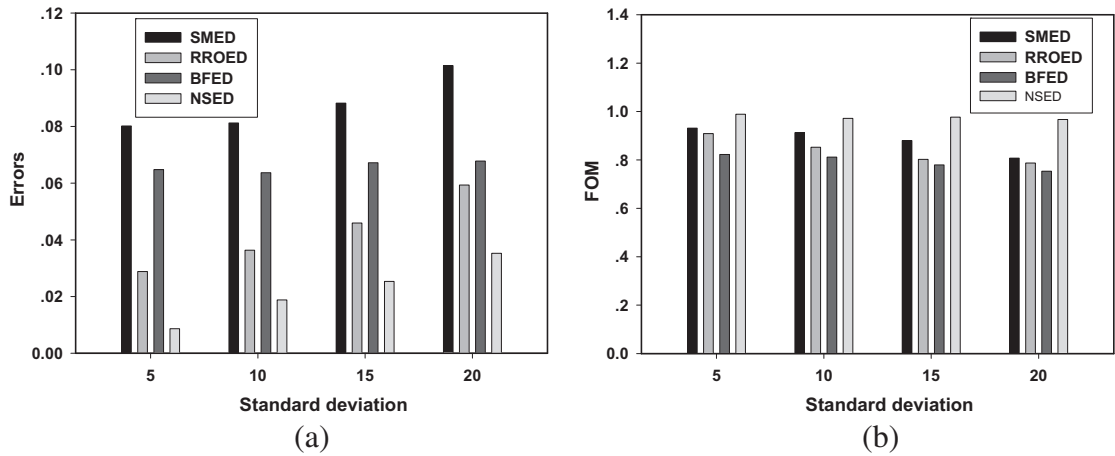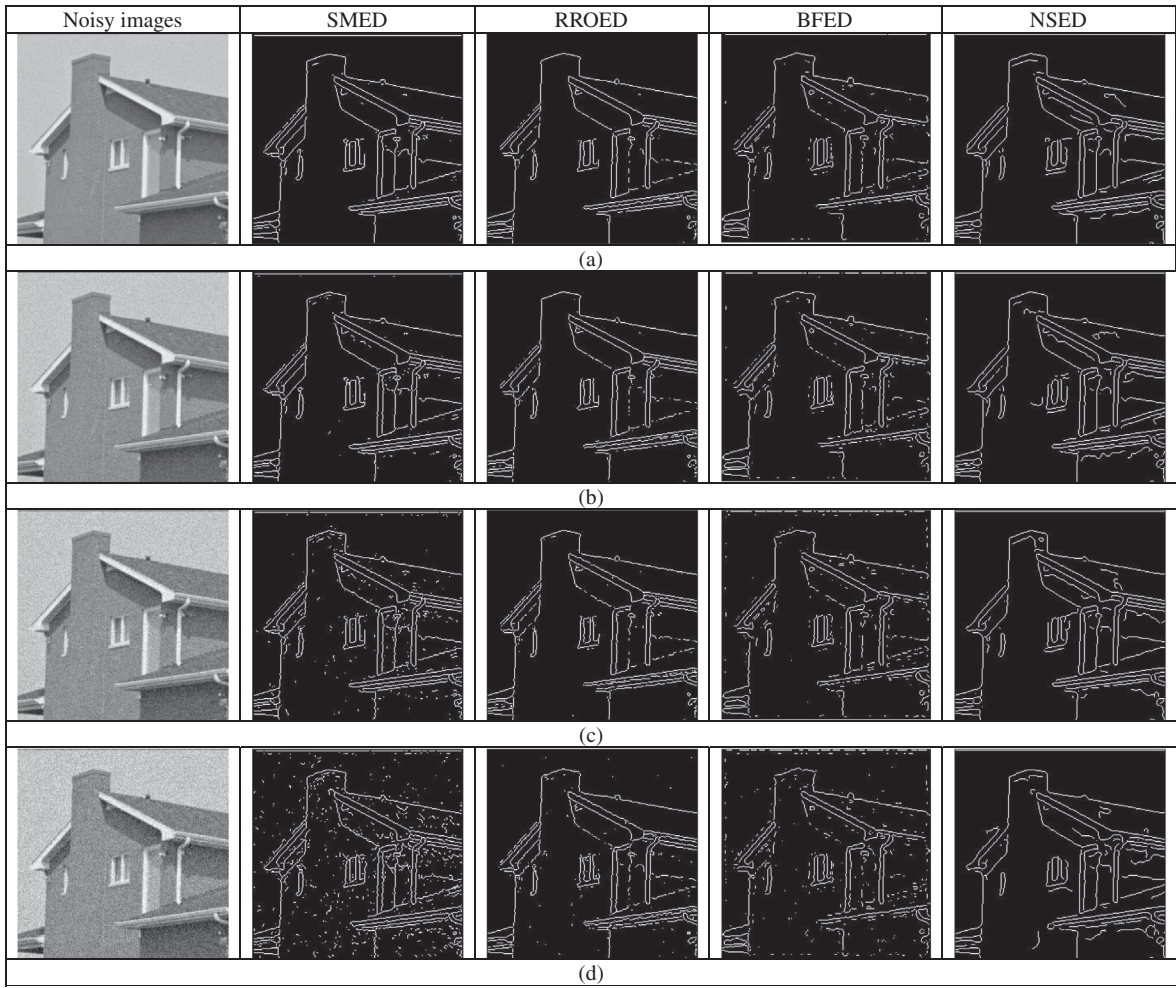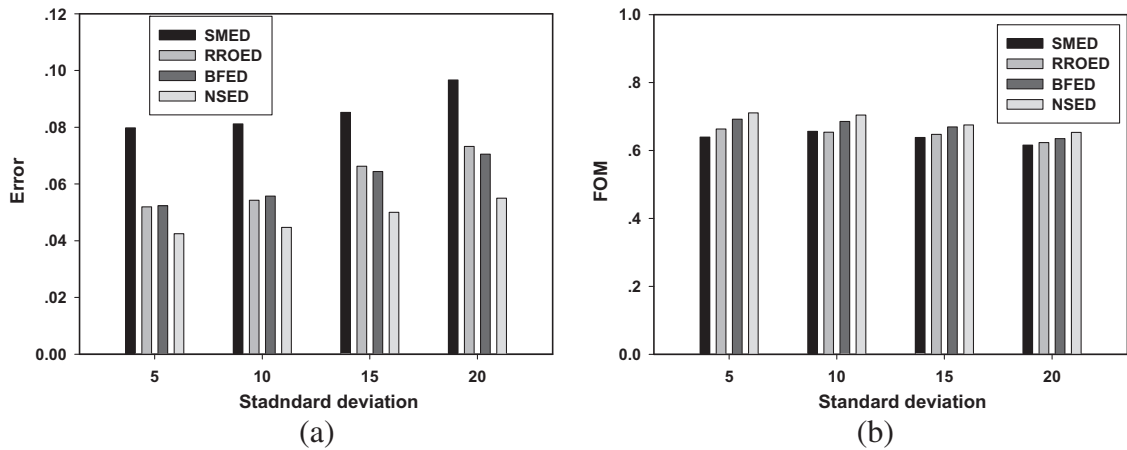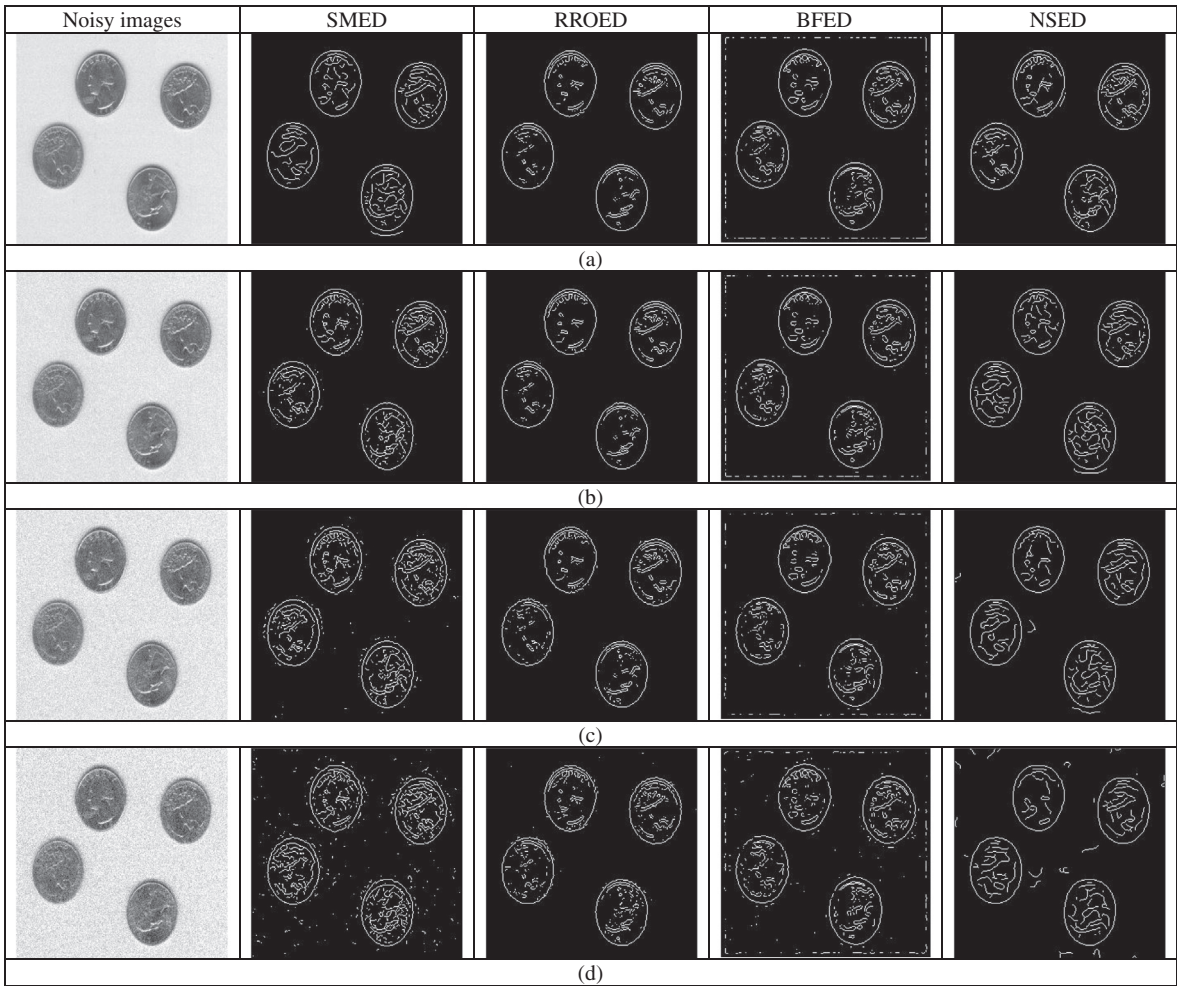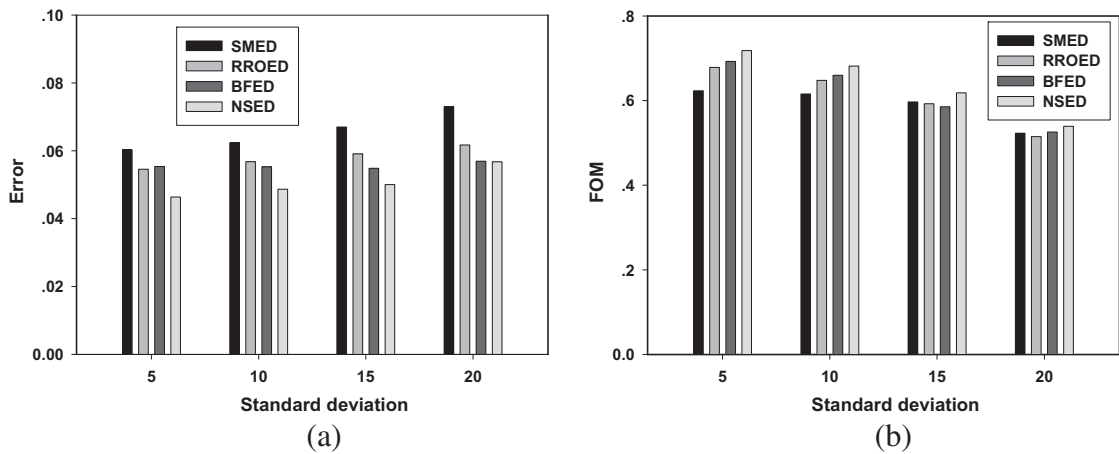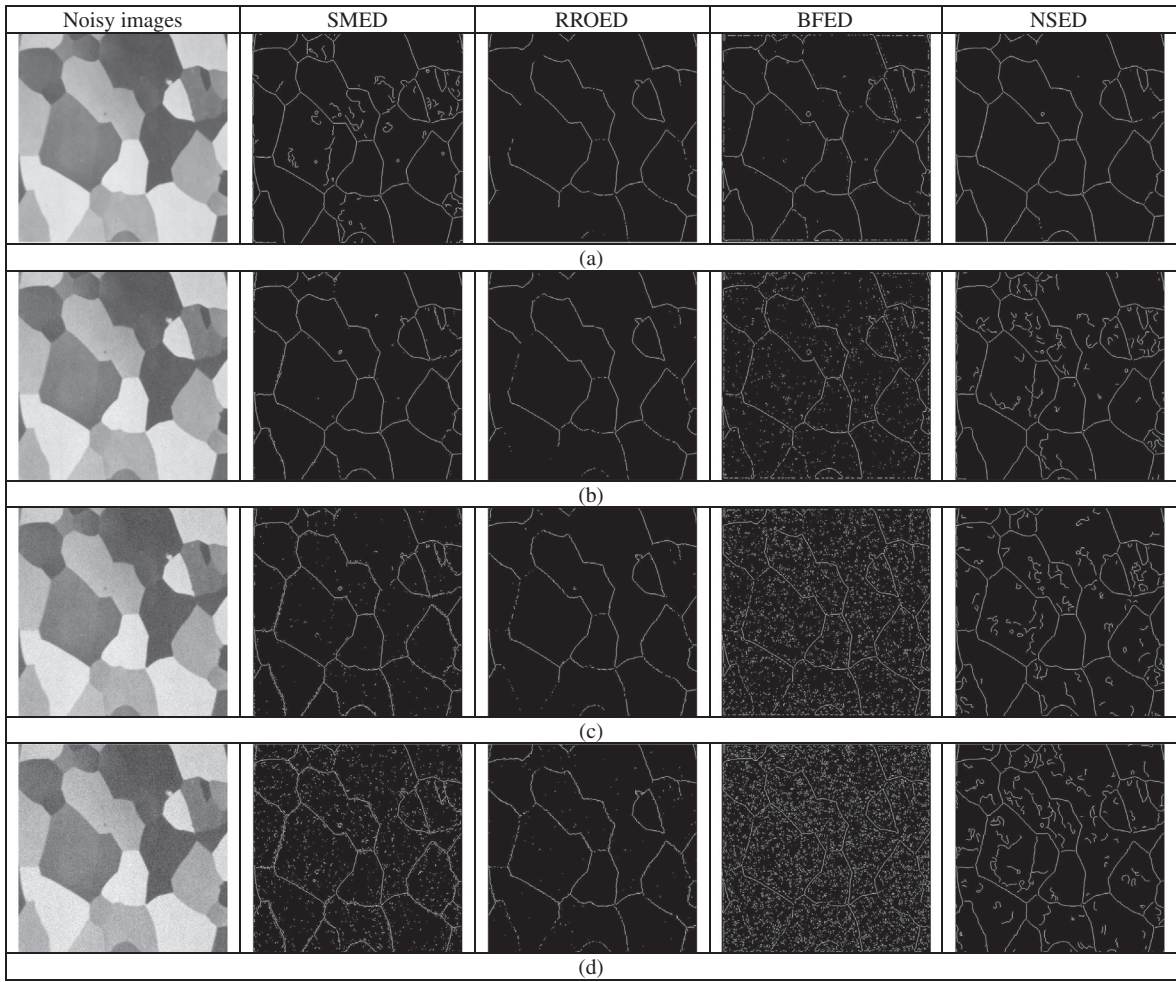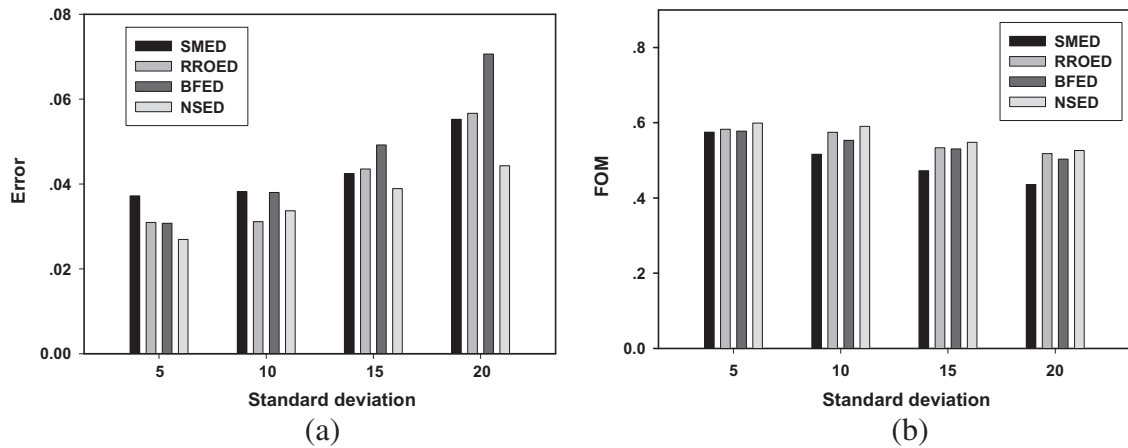

(a)

(b)

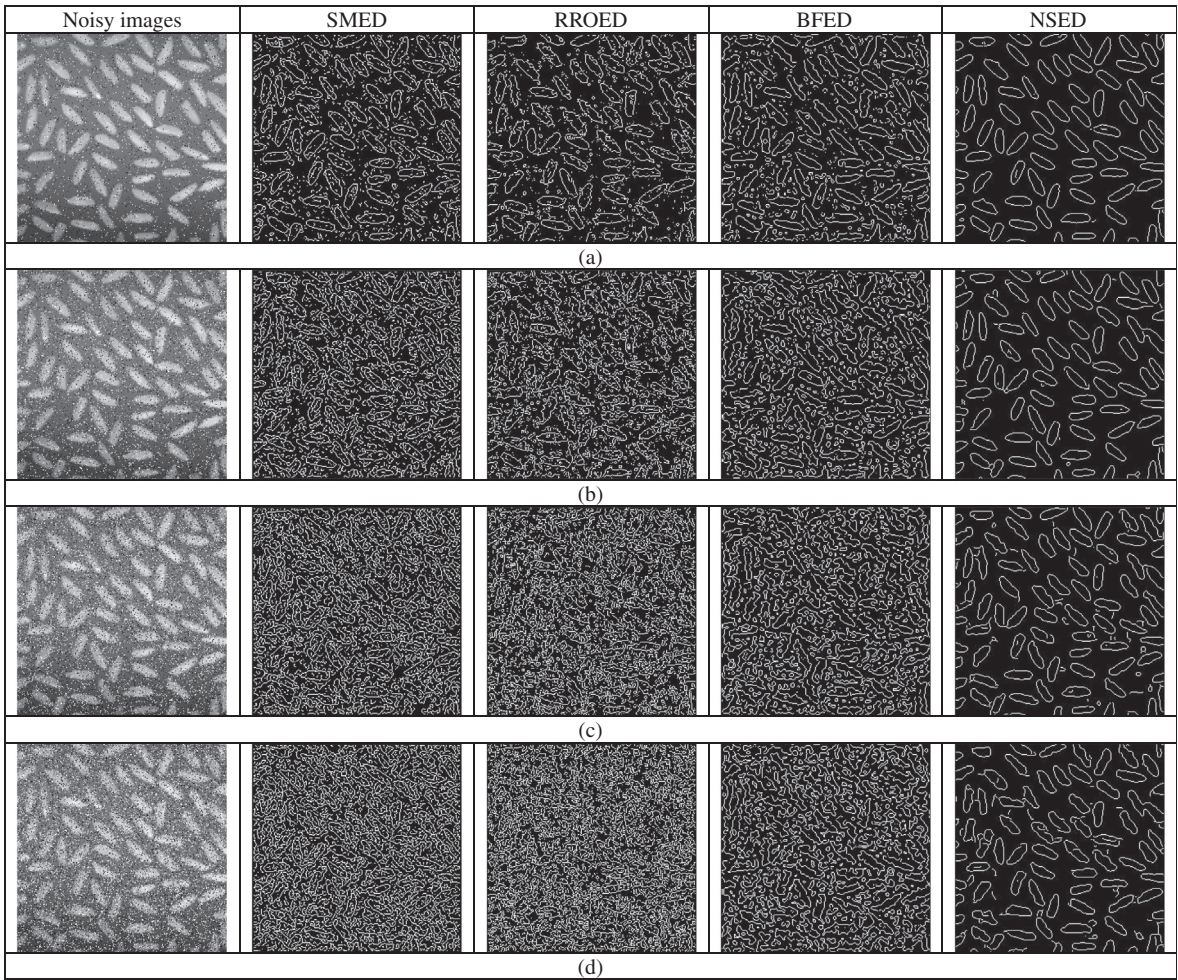**Fig. 6.** Edge detection performance comparison on *Rice* image (a) *E* comparison (b) *FOM* comparison.

Noisy images	SMED	RROED	BFED	NSED

(a)

(b)

(c)

(d)

Fig. 7. Edge detection result comparison on *House* image with different Gaussian noise levels (a) results on *House* image with Gaussian noise (σ = 5) (b) results on *House* image with Gaussian noise (σ = 10) (c) results on *House* image with Gaussian noise (σ = 15) (d) results on *House* image with Gaussian noise (σ = 20).

Fig. 8. Edge detection performance comparison on *House* image (a) *E* comparison (b) *FOM* comparison.

Fig. 9. Edge detection result comparison on *Eight* image with different Gaussian noise levels (a) results on *Eight* image with Gaussian noise (σ = 5) (b) results on *Eight* image with Gaussian noise (σ = 10) (c) results on *Eight* image with Gaussian noise (σ = 15) (d) results on *Eight* image with Gaussian noise (σ = 20).

Fig. 10. Edge detection performance comparison on *Eight* image (a) *E* comparison (b) *FOM* comparison.

Fig. 11. Edge detection result comparison on *Alumgrns* image with different Gaussian noise levels (a) results on *Alumgrns* image with Gaussian noise (σ = 5) (b) results on *Alumgrns* image with Gaussian noise (σ = 10) (c) results on *Alumgrns* image with Gaussian noise (σ = 15) (d) results on *Alumgrns* image with Gaussian noise (σ = 20).

Fig. 12. Edge detection performance comparison on *Alumgrns* image (a) *E* comparison (b) *FOM* comparison.

Fig. 13. Edge detection result comparison on *Rice* image with different Salt and Pepper noise levels (a) results on *Rice* image with Salt and Pepper noise (*d* = 0.05) (b) results on *Rice* image with Salt and Pepper noise (*d* = 0.1) (c) results on *Rice* image with Salt and Pepper noise (*d* = 0.15) (d) results on *Rice* image with Salt and Pepper noise (d = 0.2).
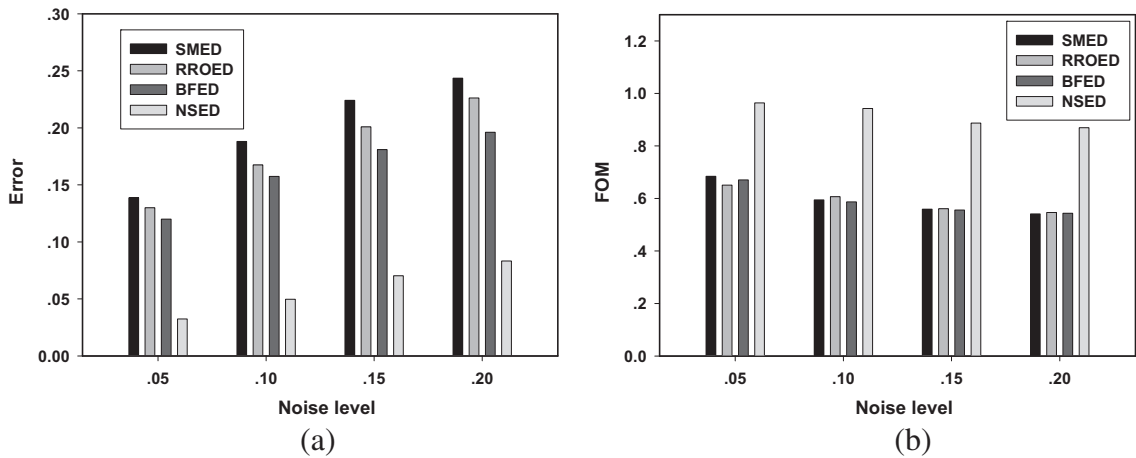
Fig. 14. Edge detection performance comparison on *Rice* image with Salt and Pepper noise (a) *E* comparison (b) *FOM* comparison.

At first, the proposed method is compared with SMED, RROED and BFED on the images without noise. The detection results are shown in Fig. 4. In Fig. 4, the second column is the ground truth edge image, which is obtained on the noise free images using the traditional Canny edge detector. The results by different methods are shown in different columns.

Then, four original images are added to Gaussian noise with mean 0 and four standard deviations (σ = 5, 10, 15, 20). The noisy images and result images using different methods are shown in Figs. 5, 7, 9 and 11. In the first column of Figs. 5, 7, 9 and 11, the corrupted images have different standard deviations, and the second, third, fourth and fifth columns are the edge detection results using SMED, RROED, BFED and NSED methods, respectively.

Fig. 5 is the results of *Rice* image having different noise levels. There are many false edges and noise in the results of SMED. In the results of NSED, most edges of rice grains are extracted and no noise appears. BFED missed some edges when the noise level increases. In Fig. 7, *House* image, the edges of house obtained by NSED are detected correctly and localized precisely. NSED does not find many false edges when the noise standard deviation is high. The similar comparison results happen in Figs. 9 and 11.

From Figs. 4, 5, 7, 9 and 11, it can be seen that the NSED method achieves very good results with few false edges and high localization accuracies. At different noise levels, the NSED detects the most shapes of the objects, while the detection performances of others are affected by the noise and some objective boundaries are lost and disconnected. Compared with the results by other methods at all noise levels, the detection results by the NSED are smoother and more connected, and edge position and orientation are more accurate, which can be demonstrated in the last column of Figs. 5, 7, 9 and 11. The outperformance benefits from the facts the NSED approach handles the indeterminacy of the images well and DAM operation provides more directional information in neutrosophic set domain.

To compare the edge detection performance of the methods on real images quantitatively, the *E* and *FOM* values of the result images of these methods are calculated at different noise levels. The comparison results are shown in Figs. 6, 8, 10 and 12, in which the result values of different methods are displayed using different bars, respectively. It is seen NSED obtains the less *E* and bigger *FOM* than other methods at all noise levels.

In Fig. 6, the average *E* of the proposed method is 0.0221, much less than that of SMED (0.0877), RROED (0.04262) and BFED (0.06586). The average *FOM* of the proposed method is 0.9761, much higher than that of the SMED (0.8828), RROED (0.8377) and BFED (0.7918). There are similar results in Figs. 8, 10 and 12. It is not difficult to see NSED has better performances than SMED, RROED and BFED.

In addition, the *Rice* image is added with Salt & Pepper noise with different noise density (noise density *ds* = 0.05, 0.1, 0.15, 0.2). The noisy images and result images are shown in Fig. 13, and the values of *E* and *FOM* are shown in Fig. 14. From Fig. 14, it is seen that NSED performs better than others on the edge detection and localization with higher *FOM* values and less *E* values.

The statistical analysis is also taken on the real images to test the performance of comparison on metrics *E* and *FOM* via a paired *t*-test. The *p*-values are shown in Table 4. The results are similar to Table 3, and all *p*-values are less than 0.05. The statistical analysis results also demonstrate that the improvement is statistically significant comparing the NSED with SMED, RROED and BFED on the real images.

We also employ more real images to test the performance of NSED. The results on original images are show in Fig. 15, and results on the images with Gaussian noise and on the images with different parameters and noise type are displayed in Figs. 16 and 17, respectively.

From the comparisons and experiments on the visual effect and quantity measures in Figs. 4–17, the proposed method is more effective on the edge details detection and noise restraining. It not only detects most true edges properly and accurately and achieves the better localization accuracy, but also has higher *E* and *FOM* values with different noise levels.

An experiment is taken to compare the time consumption of NSED, SMED, RROED and BFED methods. The NSED takes less than 21 s per image on average for an AMD Phenom (tm) 9500 Quad-core Processor, 2.2 GHz. Table 5 compares the average computational time on an image for different algorithms. The NSED takes the similar CPU times as SMED and RROED methods, and higher speed than BFED method.

In the NSED method, the directional information was employed and a directional α-mean operation was adopted to overcome the drawback of the traditional α-mean, which could blur the image and make the detected edges inaccurate. Here, an experiment is taken to compare the NSED method with the method using the traditional α-mean operation (MNSED) on edge detection. The comparison results are illustrated in Fig. 18. Because the MNSED method blurs the edges on the image, the detected edge by MNSED is smooth and inaccurate, shown in Fig. 18(b). Especially, some small detailed edges are missed in Fig. 18(c and d).

Table 4
Statistical comparison on real images using *E* and *FOM*.

Metric	*p*-Value		
	SMED vs. NSED	RROED vs. NSED	BFED vs. NSED
E	4.0540×10^{-5}	0.0016	0.0001
FOM	0.0001	8.582×10^{-6}	5.1612×10^{-5}

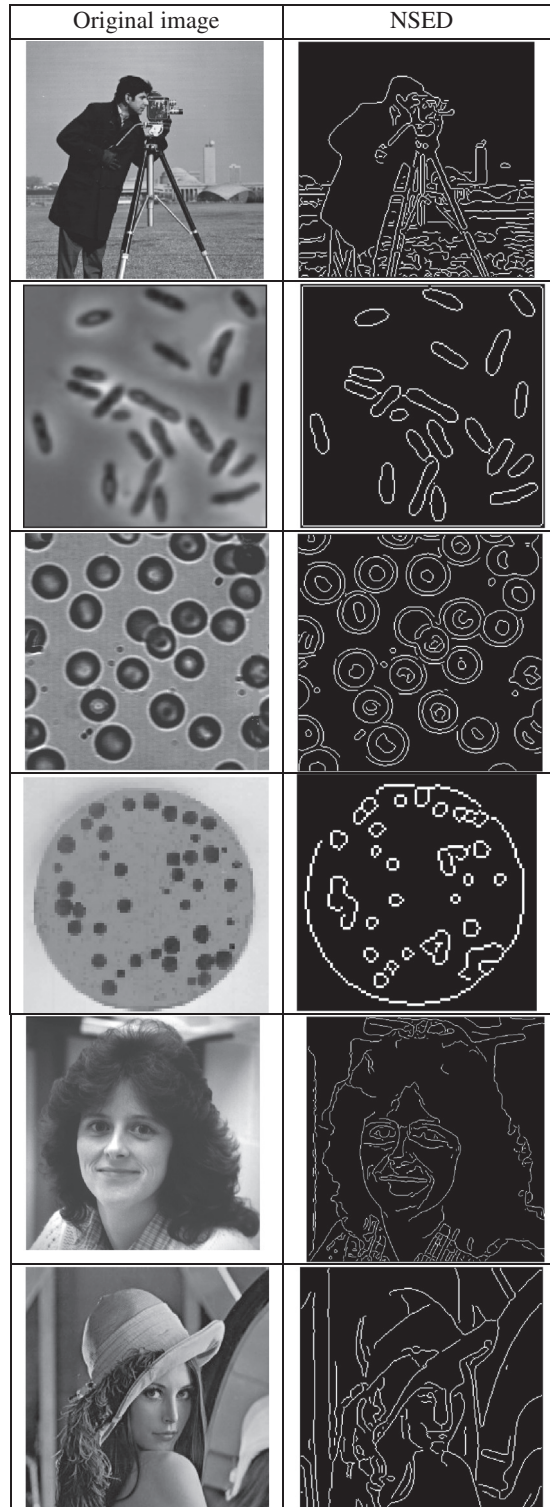| Original image	NSED

Fig. 15. Edge detection results on six original images using NSED.

4.3. Comparison with anisotropic diffusion algorithm

A new experiment was taken to compare the DAM filtering operation with the anisotropic diffusion algorithm [23] on the edge detection. The DAM filtering operation was replaced using the original anisotropic diffusion filter (ADF), and other steps
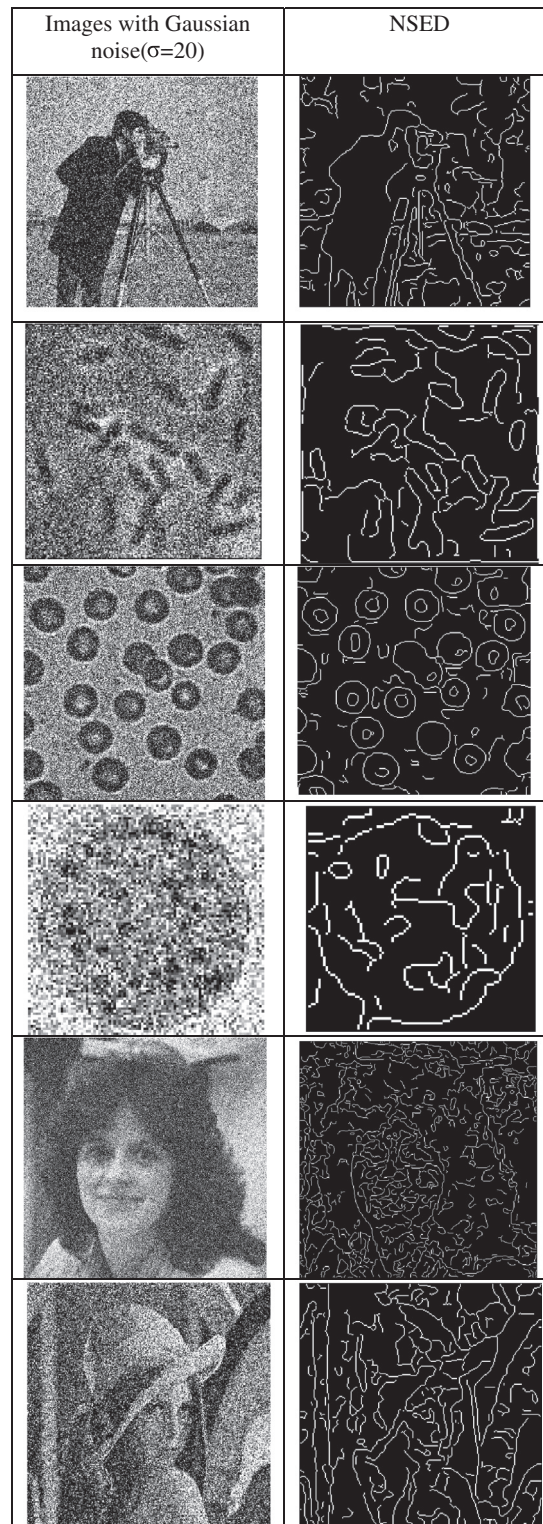
Fig. 16. Edge detection results on six images with Gaussian noise (σ = 20) using NSED.

and parameters were same as the NSED method. The edge detection results evaluated the *FOM* quantitatively. The results of images are shown in Fig. 19 and the values of *FOM* are listed in Table 4. The results in Fig. 19 show the NSED method has better performance than the method based on ADF. In Table 6, the *FOM* results of the NSED method are higher than those

of the ADF based method, which also demonstrate the DAM filtering is superior to the original ADF processing. The superiority of DAM leads to the better performance of the edge detection.

4.4. Experiment on the convergence

In NSED, the iteration process has been used to detect the edge on the image. To test the convergence of the edge detection in NSED, we perform an experiment to investigate its edge detection performance with different iteration times. In this experiment, the noisy images are used to investigate the relationship between the numbers of the detected edge pixels and the iteration numbers. The results are shown in Figs. 20 and 21. From the curves in Figs. 20 and 21, it shows the number of detected edge pixels becomes stable after iteration number is greater than 20 times. The performance of edge detection NSED converges and is stable.
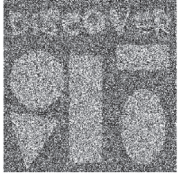
Fig. 17. Edge detection results using NSED on the image with different types of noise.

Table 5
Average CPU time for different algorithms.

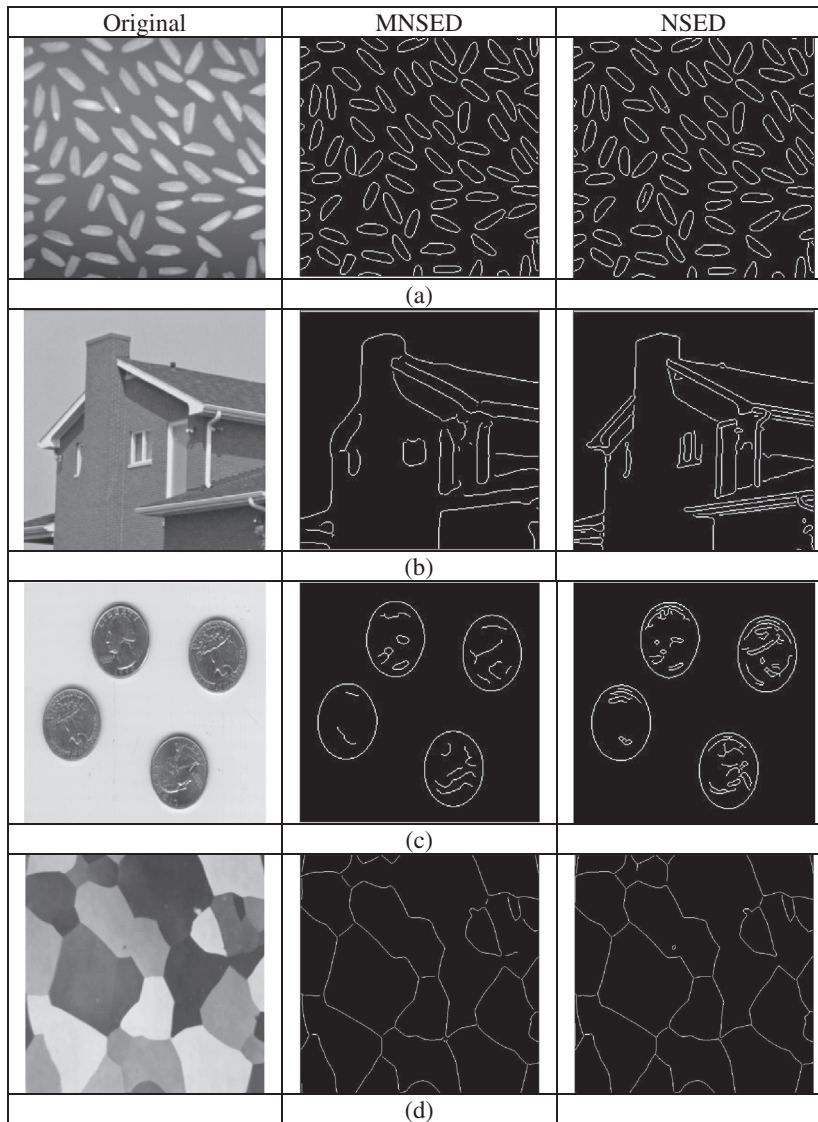| Algorithm	CPU time (s)
SMED	19
RROED	25
BFED	180
NSED	21

Fig. 18. Edge detection result comparison on four original images using MNSED and NSED (a) results on *Rice* image (b) results on *House* image (c) results on *Eight* image (d) results on *Alumgrns* image.

4.5. Experiment on the parameters

The proposed method also needs three constant thresholds (w, α and ∇_{th}) as parameters. The edge detection performance becomes constant when these parameters are optimized. We employ an experiment on noisy images to show the performance of the NSED with different values of parameters. During these parameters, w is the local window size and usually assigned as 3, 5 or 7. In the experiment, we only consider the performance at different values of α and ∇_{th}, and the value of *FOM* is used to evaluate the edge detection performance. Figs. 22 and 23 display the relationship between *FOM* and α and ∇_{th}. From those two figures, we can see the values of *FOM* become constant when these parameters are close to the optimized values ($\alpha = 0.15$ and $\nabla_{th} = 0.04$). The experimental results demonstrate that the edge detection performance of NSED is constant when the parameters are optimized.

4.6. Experiment on the large scale image database

We further test the performance of NSED in the large scale image dataset and conduct an experiment on a publicly available dataset namely Sowerby image database [24]. This database contains one hundred segmented images of road scenes in countryside and their associated labels. The proposed method is also evaluated using two metrics *E* and *FOM*, and compared

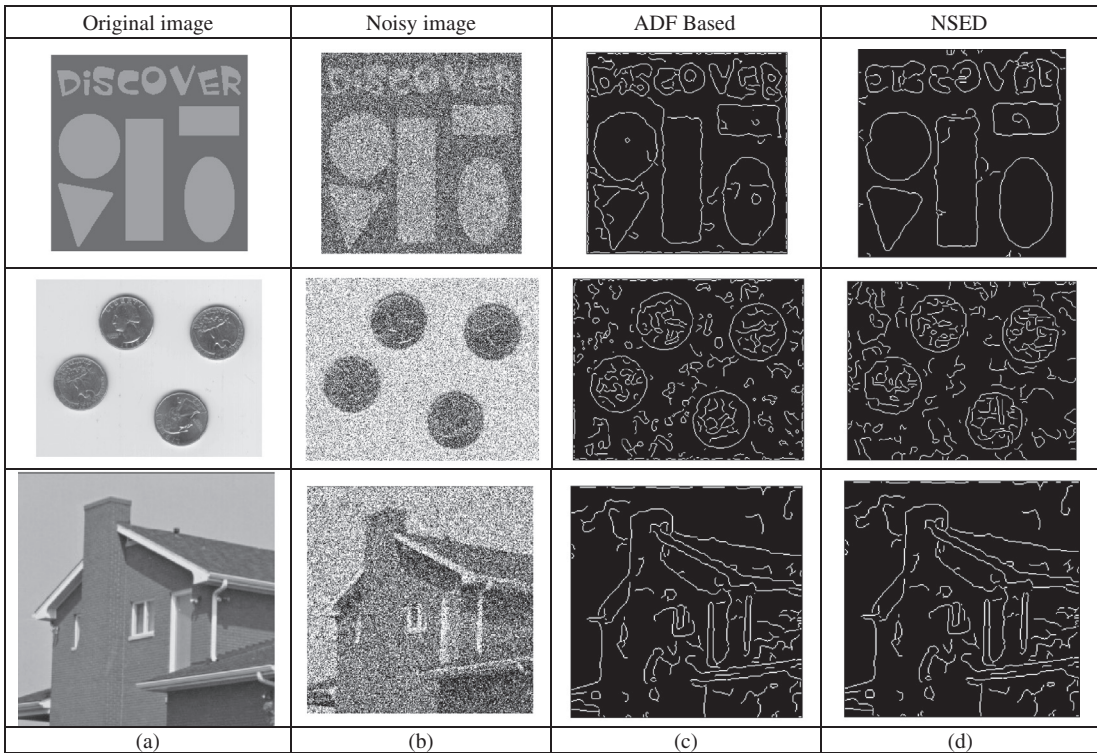| Original image	Noisy image	ADF Based	NSED
(a)	(b)	(c)	(d)

Fig. 19. Edge detection results on different noisy images (a) original images (b) noisy images with Gaussian noise ($\sigma = 25$) (c) results of the method based on ADF (d) results of the NSED.

Table 6
FOM values on different images.

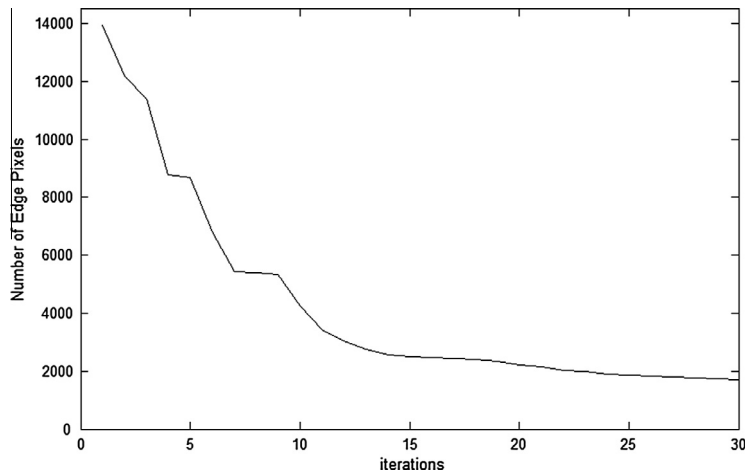| Images	ADF based method	NSED
Discover	0.6064	0.7812
Eight	0.4081	0.4788
House	0.5101	0.5136
Average	0.5082	0.5912

Fig. 20. The relation between number of detected edge points and iteration numbers on the *Eight* image with Gaussian noise ($\sigma = 25$).
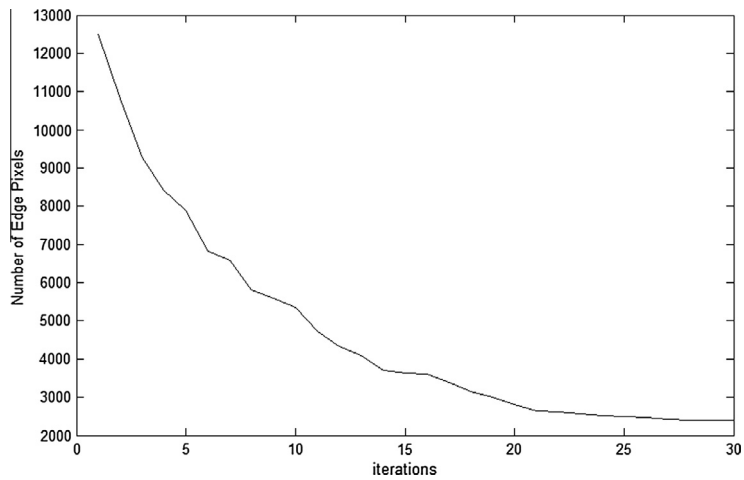
Fig. 21. The relation between number of detected edge points and iteration numbers on the *Discover* image with Gaussian noise (σ = 25).
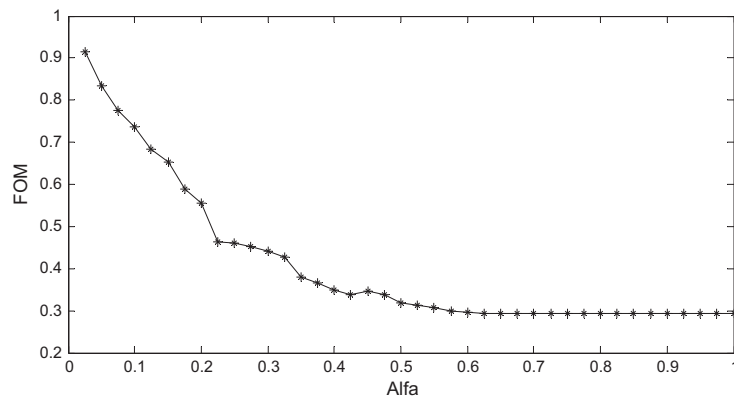

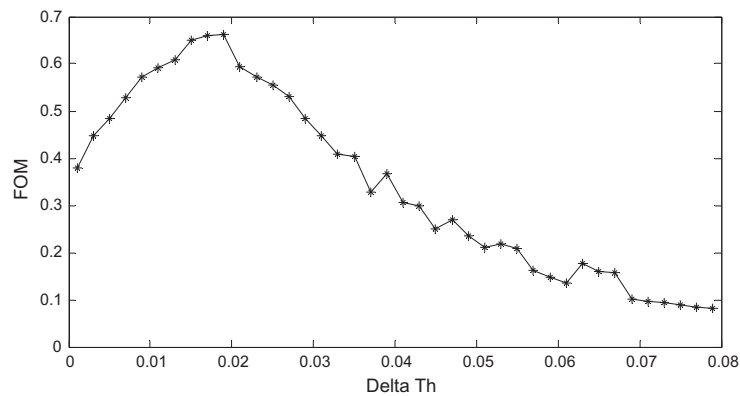**Fig. 22.** The relationship between the parameter Alpha and *FOM*.

Fig. 23. The relationship between the parameter delta$_{th}$ and *FOM*.

with the SMED method. The average *E* value for all images is 0.0143 for the SMED and 0.0078 for the NSED methods, and the average *FOM* value for all images is 0.5491 for the SMED and 0.6651 for the NSED methods, respectively. The comparison is tested on *E* and *FOM* metrics using the paired *t*-test. The *p*-values of the test on *E* and *FOM* are 2.0823×10^{-55} and 0.042, respectively. Both *p*-values are less than 0.05. The statistical analysis demonstrates that the improvement is statistically significant comparing the NSED with SMED on the images in the Sowerby database.

5. Conclusions

A novel edge detection approach which combines the NS theory and directional α-mean operation is introduced in this paper. According to the NS theory, the image is described by using three membership sets, T, I and F. The directional α-mean operation is employed to reduce the image's indeterminacy. We have carried out extensive experiments with various types of noise and noise levels on various types of images, investigating the performance of the proposed method. Moreover, extensive comparisons with SMED, RROED and BFED methods are carried out for showing the superiority of our proposal. The comparison with the ADF method is also considered in the experiments. The obtained experimental results are evaluated quantitatively with *FOM* and *E* values. In addition a comparative experiment is considered according to the average CPU times. According to the experimental results, the proposed method can not only perform better on clear and simple images, but also on the noisy and complex images, due to the fact that the proposed approach can handle the indeterminacy of the images well. In addition, the run time of our method is quite close to the compared methods. A statistical test is also considered for further analyzing the experimental results, and the statistical test results show that the proposed method has a significant improvement. We also test the performance of our proposal in the large scale image dataset and conducted an experiment on a publicly available dataset namely the Sowerby image database. The obtained results indicate the better performance of our proposal. According to various evaluation criterions, the proposed method can be used successfully in many image processing applications. In the future, the NSED is planned for edge detection in medical images, and it will also find a wide application in this area.

References

[1] Ziou D, Tabbone S. Edge detection techniques – an overview. Pattern Recogn Image Anal 1998;8:537–59.
[2] Chang CY. Contextual-based Hopfield neural network for medical image edge detection. Opt Eng 2006;45:037006.
[3] Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 1986;8:679–98.
[4] Bao P, Zhang L, Wu X. Canny edge detection enhancement by scale multiplication. IEEE Trans Pattern Anal Mach Intell 2005;27:1485–90.
[5] Tang G, Zhong X, Zhang F, Jin Z. X-ray image edge detection based on wavelet transform and lipschitz exponent. In: Proc. 2009 intelligent information technology and security informatics. Second international symposium on IITSI'09; 2009. p. 195–7.
[6] Shih MY, Tseng DC. A wavelet-based multiresolution edge detection and tracking. Image Vis Comput 2005;23(4):441–51.
[7] Chacon-M MI, Prieto RC, Sandoval R, Alejandro Rodriguez R. A soft image edge detection approach based on the time matrix of a PCNN. In: Proc. 2008 IEEE international joint conference on neural networks, IJCNN 2008; 2008. p. 463–9.
[8] Emin Yüksel M. Edge detection in noisy images by neuro-fuzzy processing. AEU – Int J Electron Commun 2007;61:82–9.
[9] Verma O, Hanmandlu M, Sultania A, Parihar A. A novel fuzzy system for edge detection in noisy image using bacterial foraging. Multidim Syst Signal Process 2011;22:1–18.
[10] Singh B, Singh AP. Edge detection in gray level images based on the Shannon entropy. J Comput Sci 2008;4:186–91.
[11] Bhardwaj K, Mann PS. Adaptive Neuro-Fuzzy Inference System (ANFIS) based edge detection technique. Int J Sci Emerg Technol Latest Trends 2013;8(1):7–13.
[12] Zhang F, Zhang X, Cao K, Li R. Contour extraction of gait recognition based on improved GVF Snake model. Comput Electr Eng 2012;38:882–90.
[13] Zhou Y, Liu H, Song E, Huang Z. Image bilevel thresholding based on multiscale gradient multiplication. Comput Electr Eng 2012;38(4):853–61.
[14] Wang H, Sunderraman R, Smarandache F, Zhang YQ. Interval neutrosophic sets and logic: theory and applications in computing, infinite study; 2005.
[15] Guo Y, Cheng HD. New neutrosophic approach to image segmentation. Pattern Recogn 2009;42:587–95.
[16] Guo Y, Cheng HD. A new neutrosophic approach to image denoising. New Math Nat Comput 2009;5(3):653–62.
[17] Sengur A, Guo Y. Color texture image segmentation based on neutrosophic set and wavelet transformation. Comput Vis Image Understand 2011;115(8):1134–44.
[18] Guo Y, Cheng HD, Tian J, Zhang Y. A novel approach to speckle reduction in ultrasound imaging. Ultrasound Med Biol 2009;35.
[19] Black MJ, Sapiro G, Marimont DH, Heeger D. Robust anisotropic diffusion. IEEE Trans Image Process 1998;7:421–32.
[20] Lim DH. Robust edge detection in noisy images. Comput Stat Data Anal 2006;50:803–12.
[21] Pratt WK. Digital image processing. John Wiley & Sons; 1978. pp. 429–432.
[22] Ma L, Staunton RC. A modified fuzzy C-means image segmentation algorithm for use with uneven illumination patterns. Pattern Recogn 2007;40:3005–11.
[23] You YL, Kaveh M. Fourth-order partial differential equations for noise removal. IEEE Trans Image Process 2000;9:1723–30.
[24] Collins D, Wright WA, Greenway P. The Sowerby image database. In: Seventh international conference on image processing and its applications, vol. 1; 1999. p. 306–10.
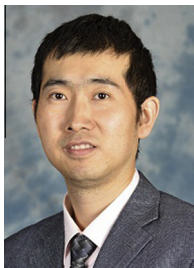
Yanhui Guo received his Ph.D. degree in the Department of Computer Science at Utah State University in 2010. He is currently an assistant professor in the School of Science at Saint Thomas University, Miami Gardens, FL. His research interests include image processing, pattern recognition, medical imaging processing, computer aided detection/diagnosis, and computer assisted surgery.

Abdulkadir Şengur obtained his Ph.D. degree from the Department of Electric and Electronics Engineering at Firat University, Turkey, in 2006. He is an associated professor Dr. in the Technology Faculty, Department of Electrical and Electronics Engineering at Firat University. His research interest areas include pattern recognition, machine learning and image processing.