A Robust Statistical Color Edge Detection for Noisy Images

Mina Alibeigi*
Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran
minaalibeigi@gmail.com
Niloofar Mozafari
Department of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran
mozafari@cse.shirazu.ac.ir
Zohreh Azimifar
Department of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran
azimifar@cse.shirazu.ac.ir
Mahnaz Mahmoodian
Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran
m.mahmoodian@ut.ac.ir

Received: 02/July/2014 Revised: 27/Jan/2015 Accepted: 25/Feb/2015

Abstract

Edge detection plays a significant role in image processing and performance of high-level tasks such as image segmentation and object recognition depends on its efficiency. It is clear that accurate edge map generation is more difficult when images are corrupted with noise. Moreover, most of edge detection methods have parameters which must be set manually. Here we propose a new color edge detector based on a statistical test, which is robust to noise. Also, the parameters of this method will be set automatically based on image content. To show the effectiveness of the proposed method, four state-of-the-art edge detectors are implemented and the results are compared. Experimental results on five of the most well-known edge detection benchmarks show that the proposed method is robust to noise. The performance of our method for lower levels of noise is very comparable to the existing approaches, whose performances highly depend on their parameter tuning stage. However, for higher levels of noise, the observed results significantly highlight the superiority of the proposed method over the existing edge detection methods, both quantitatively and qualitatively.

Keywords: Edge Detection; Color; Noisy Image; RRO Test; Regression; Pratt's Figure of Merit.

1. Introduction

One of the most important processes in low-level image processing is edge detection, and the performance of high-level image processing tasks is highly dependent on this process.

Novak and Shafer [1] claim that at most 90% of the information in color images is similar to that of gray images, which means there still remains information that may not be detected in gray images. On the other hand, in most applications, images are often corrupted by noise. Generation of accurate edge map becomes more critical and complicated in the presence of noise. Thus, the detector must be robust against noise.

In this paper, we present a robust color edge detector for automatic edge detection in the presence of additive noise. The main advantage of our detector is that the parameters are set automatically based on image content. The proposed approach adopts a statistical test called Robust Rank-Order (RRO) test [2] to detect edges. In order to detect fewer false edges in the images with higher levels of noise, lower significance levels are employed with the RRO test. Unlike the existing methods, the proposed algorithm adapts its parameters based on image content automatically. Experimental results for lower levels of noise show that the proposed edge detector is comparable to the existing approaches; for higher levels of noise, the proposed method performs significantly better, both quantitatively and qualitatively.

The rest of this paper is organized as follows: Section 2 reviews the related works. In Section 3 the proposed algorithm for edge detection is explained. The experimental results are given in Section 4 and Section 5 presents conclusion and future work.

2. Related Works

The problem of edge detection saw its pioneering work at least as early as 1986 by the work of Canny [3]. Since then, various edge detection algorithms have been developed [4,5,6,7,8,9, and 10]. Most of these algorithms are based on computation of the intensity gradient vector, which are, in general, sensitive to noise. In order to reduce this problem, some spatial averaging is combined with differentiation process, such as the Laplacian of Gaussian operator and zero-crossing detection [5]. As stated before, one of the well-known methods of this category is Canny edge detector [3] which finds edges by looking for local maxima of the gradient of image. The gradient is calculated using the derivative of a Gaussian filter. Canny edge detector includes the weak edges in the

* Corresponding Author

output only if they are connected to strong edges. This method is therefore less likely to be fooled by noise, and more likely to detect true weak edges than those detectors named above.

Also in recent years, many soft computing algorithms were proposed for edge detection [11,12,13,14,15, and 16]. In most of these algorithms, the problem of edge detection is seen as an optimization problem. For example, in [12, 13, and 14] bio-inspired optimization algorithms are used to address the edge detection problem. In [14] the authors studied the effect of different topologies in PSO-based edge detection techniques. These algorithms are more robust against noise compared to gradient based methods. However, they are much more time consuming. Also, their performance depends on implementation details and parameters settings. Therefore, their results might not be easily reproducible.

Statistical edge detection is an alternative to deterministic edge detection. Bovik et al. [17] proposed several nonparametric statistical tests for edge detection in noisy images. However, their work can only detect edges in four directions and no quantitative comparison was reported for this method. Variance tests were also applied for edge detection by Aron and Kruz [18] and Kruz and Benteftifa [19]. Recently, Lim and Jang [20] have compared three statistical two-sample tests for edge detection in noisy images. These edge operators are based on a test statistic, computed using the modified grey level values. In another study, Lim [2] proposed a new operator based on RRO test for detecting edges in all possible orientations in noisy images. The RRO detector has been designed for grey images and requires two user defined parameters: the edge height (δ) and the significance level (α) of statistical test, which is often set to 0.05. Since different values of these parameters may lead to different results, adaptive specification of these parameters based on image content could be interesting.

As mentioned before, color images contain more information than grey ones. In order to benefit from all the information present in color images, proposing a color edge detection method becomes indispensable. Due to importance of color information, many color edge detection methods were proposed [21, 22, 23, and 24]. Russo and Lazzari [23] proposed edge detection in color images corrupted by Gaussian noise, which adopts a multi-pass preprocessing stage that gradually reduces the amount of noise in each R, G and B channels of image, independently.

Since statistical approaches to edge detection are more robust in the presence of noise, in this study we propose a statistical color edge detector which is completely adaptive and operates robustly in the presence of Gaussian noise, the most common type of noise encountered during image acquisition [25].

3. The Proposed Detector: Robust Color Edge Detector

The proposed algorithm for color edge detection includes three steps: noise estimation, prefiltering, and edge detection (see Figure 1). The details of these steps are discussed in the next subsections.

3.1 Noise Estimation

In the first step, noise estimation, the standard deviation of noise must be estimated. There exist several well-known methods to estimate the level of noise [23,26]. In this paper we use median noise estimation method proposed by Pizurica et al. [27], due to its efficiency and better accuracy compared to other methods. In median noise estimation, the median of wavelet coefficients in high-high band of finest scale is used to calculate the standard deviation of noise according to the following formula:

$$\sigma_n = Median(|\lambda_{h,h,1}|)/0.6745 \quad (1)$$

Where $\lambda_{h,h,1}$ is the wavelet coefficient in high-high band of the first scale of noisy image.

3.2 Prefiltering

In the second step, we adopt a prefiltering to reduce the effect of noise and outliers. Let us define noisy pixels as pixels that are corrupted by noise with amplitude not too different from their neighbors, and outliers as pixels that are corrupted by noise with amplitude much larger than their neighbors [23]. We consider two types of filters to reduce the effect of these pixels. Figure 1 illustrates the block diagram of these filters. Type A filter is to remove noisy pixels. It computes the difference of each pixel with a predefined function of its neighbors as follows:

1. Small difference between the current pixel and its neighbors is due to noise and must be decreased.
2. Large difference between the current pixel and its neighbors is due to edge and must be preserved.

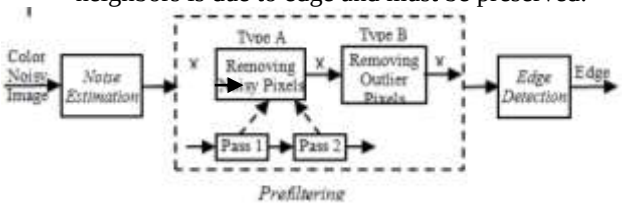


Fig. 1. The framework of the proposed edge detector.

We adopt a two-pass procedure to increase the effectiveness of the smoothing action [23]. This procedure is defined according to following equations:

$$x_{Pass}^{(c)}(i,j) = x_{in}^{(c)}(i,j) + \frac{1}{8}\sum_{m=-1}^{1}\sum_{n=-1}^{1}\zeta_1^{(c)}(x_{in}^{(c)}(i+m,j+n), x_{in}^{(c)}(i,j)), (m,n) \neq (0,0) \quad (2)$$

$$x_{A2B}^{(c)}(i,j) = x_{pass}^{(c)}(i,j) + \frac{1}{8}\sum_{m=-1}^{1}\sum_{n=-1}^{1}\zeta_2^{(c)}(x_{pass}^{(c)}(i+m, j+n), x_{pass}^{(c)}(i,j)), (m,n) \neq (0,0) \quad (3)$$

Where, $c = 1, 2$ or 3 indicates the channel number for R, G and B channels, respectively. ζn(c) is a parameterized nonlinear function and is defined as:

$$\zeta_n^{(c)}(u,v) = \begin{cases} u-v & |u-v| \le a_n^{(c)} \\ ((3a_n^{(c)} - u + v)/2)sgm(u-v) & a_n^{(p)} < |u-v| \le 3a_n^{(c)} \\ 0 & |u-v| > 3a_n^{(c)} \end{cases} \quad (4)$$

Where $a_n^{(c)}$ is an integer bounded at $0 < a_n^{(c)} < 256$, (for $c = 1, 2, 3$) [26].

According to (2), the smoothing filter is first applied to each channel of the color image $x_{in}^{(1)}$, $x_{in}^{(2)}$ and $x_{in}^{(3)}$ to produce intermediate results $x_{pass}^{(1)}$, $x_{pass}^{(2)}$ and $x_{pass}^{(3)}$. Then the second filtering pass is applied on these intermediate components and yields to $x_{A2B}^{(1)}$, $x_{A2B}^{(2)}$ and $x_{A2B}^{(3)}$, respectively.

As Figure 1 shows, the type B prefiltering is meant to remove outlier pixels, for which the difference between the processed pixel and all its neighbors is very large. Equation (5) illustrates this prefiltering:

$$x_{out}^{(c)}(i,j) = x_{A2B}^{(c)}(i,j) - (255)\Delta(i,j) \quad (5)$$

Where

$$\Delta(i,j) = MIN\left\{\mu_{LA}\left(x_{A2B}^{(c)}(i,j), x_{A2B}^{(c)}(i+m, j+n)\right)\right\} - MIN\left\{\mu_{LA}\left(x_{A2B}^{(c)}(i+m, j+n)\right), x_{A2B}^{(c)}(i,j)\right\} \quad (6)$$

$\mu_{LA}(u,v)$ is the membership function describing the relation of *"u is much larger than v"*.

$$\mu_{LA}(u,v) = \begin{cases} \frac{u-v}{L-1} & 0 < u - v \le L - 1 \\ 0 & u - v \le 0 \end{cases} \quad (7)$$

Here L is the maximum possible intensity value.

After applying these filters on R, G and B channels independently, a simple averaging is performed.

3.3 Edge detection

In third step, to detect edge pixels we consider a small square subimage of size 5×5 centered at each pixel. We divide the neighboring pixels centered by this square mask into two groups of size 12, as shown in Figure 2. In other words, we consider a set of $N = m + n$ independent observations, excluding the center pixel, which are divided into $X = (X_1, X_2, \dots, X_n)$ and $Y = (Y_1, Y_2, \dots, Y_m)$. It is assumed that the first set of observations comes from a continuous distribution named $F(x - \mu_x)$ and the second one comes from another continuous distribution $G(y - \mu_y)$, with μ_x and μ_y as shift parameters. Note that the two distributions F and G are not identical. In order to create our statistical model, we define two modified sets of observations $\{A_i\}$ and $\{B_i\}$ as follows:

$$A_i = \begin{cases} X_i + \delta & X_i \epsilon X \\ Y_i & Y_i \epsilon Y \end{cases}$$
$$B_i = \begin{cases} X_i - \delta & X_i \epsilon X \\ Y_i & Y_i \epsilon Y \end{cases} \quad (8)$$

Where δ is edge height parameter. This parameter determines the minimum grey level difference which leads to labeling the center pixel as an edge pixel, according to the following statistical test:

$$H_0^{\uparrow}: \mu_x + \delta \ge \mu_y \quad versus \quad H_1^{\uparrow}: \mu_x + \delta < \mu_y$$

and

$$H_0^{\downarrow}: \mu_x - \delta \le \mu_y \quad versus \quad H_1^{\downarrow}: \mu_x - \delta > \mu_y$$

Since the distributions F and G are not identical, theoretically statistical tests such as Wilcoxon test [17] may not be appropriate [2]. Thus, in this study for testing $H_0^{\uparrow}\,(or\ H_0^{\downarrow})$ against $H_1^{\uparrow}\,(or\ H_1^{\downarrow})$ on $\{A_i\}$ (or $\{B_i\}$), a statistical test named RRO test [2, 18] is used.

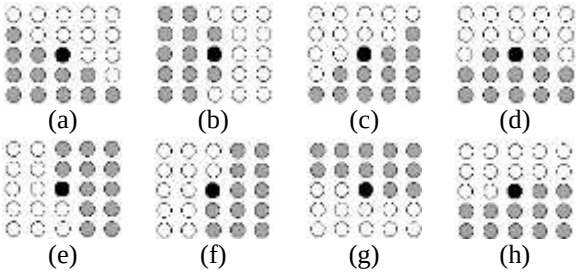


Fig. 2. Partitioning the window in eight different ways where the grey areas represent X and the white areas represent Y.

For each $X_i + \delta$, $X_i \in X$ in $\{A_i\}$, we count the number of lower-valued observations Y_i's in $\boldsymbol{Y}$. This number is denoted as $U(Y, X_i + \delta)$. Then the mean value of the $U(Y, X_i + \delta)$ values is calculated using equation (9). Similarly, we find the number of lower-valued observations $X_i + \delta$'s in $\boldsymbol{X}$ for each Y_i and will denote this counting by $U(X + \delta, Y_i)$. Then the mean value, $\overline{U(X + \delta, Y)}$, is evaluated via equation (10), as:

$$\overline{U(Y, X + \delta)} = \frac{1}{m}\sum_{i=1}^{m} U(Y, X_i + \delta) \quad (9)$$

$$\overline{U(X + \delta, Y)} = \frac{1}{n}\sum_{i=1}^{n} U(X + \delta, Y_i) \quad (10)$$

Next, two variables are defined to demonstrate the variability of $U(X + \delta, Y)$ and $U(Y, X + \delta)$ as $V_{x+\delta}$ and $V_{y+\delta}$ according to the following formulas:

$$V_{x+\delta} = \sum_{i=1}^{m}[U(Y, X_i + \delta) - \overline{U(Y, X + \delta)}]^2$$
$$V_{y+\delta} = \sum_{i=1}^{n}[U(X + \delta, Y_i) - \overline{U(X + \delta, Y)}]^2 \quad (11)$$

Finally, the test statistics for observations $\{A_i\}$ is given by:

$$U_A = \frac{n \cdot \overline{U(X + \delta, Y)} - m \cdot \overline{U(Y, X + \delta)}}{2\sqrt{V_{x+\delta} + V_{y+\delta} + \overline{U(Y.X + \delta)} \cdot \overline{U(X + \delta, Y)}}} \quad (12)$$

For $\{B_i\}$ observations, the values $U(Y, X - \delta)$, $U(X - \delta, Y)$, $V_{x-\delta}$ and $V_{y-\delta}$ are defined exactly analogous to those defined for $\{A_i\}$ observations:

$$\overline{U(Y, X - \delta)} = \frac{1}{m}\sum_{i=1}^{m} U(Y, X_i - \delta)$$

$$\overline{U(X-\delta,Y)} = \frac{1}{n}\sum_{i=1}^{n} U(X-\delta,Y_i)$$

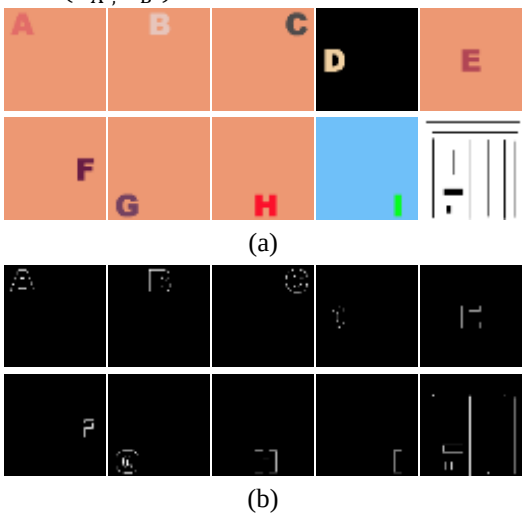
$$V_{x-\delta} = \sum_{i=1}^{m} [U(Y,X_i-\delta) - \overline{U(Y,X-\delta)}]^2$$

$$V_{y-\delta} = \sum_{i=1}^{n} [U(X-\delta,Y_i) - \overline{U(X-\delta,Y)}]^2$$

Thus the test statistic will be given by:

$$U_B = \frac{m\,.\,\overline{U(Y\,,X-\delta)} - n\,.\,\overline{U(X-\delta\,,Y)}}{2\sqrt{V_{x-\delta}+V_{y-\delta}+\overline{U(Y,X-\delta)}.\,\overline{U(X-\delta\,,Y)}}} \quad (13)$$

Having determined U_A and U_B, we have defined $U^* = \max(U_A, U_B)$.



(a)

(b)

Fig. 3. (a) The synthetic images used for training parameters of RRO test (b) Edge maps for the above synthetic images.

Accordingly, $H_0^{\uparrow}\,(or\ H_0^{\downarrow})$ will be rejected for large values of U^* *[2]*. If $U^* > u_\alpha$, for a specified threshold u_α at a significance level α, then an edge is detected. Flinger and Pollicello [28] present the critical values of U^* for small sample sizes up to 12. This threshold is constant in RRO detector with a pre-specified significance level which is usually set to 0.05, for all different levels of noises presents in images.

RRO test has been utilized for edge detection [2] and appropriate results have been reported. We, however, argue that for higher levels of noise, it is more desirable to detect only stronger edges in order to reduce the effects of false edges; therefore, we adjust the threshold in RRO test at different significance levels based on the standard deviation of the Gaussian noise. With higher levels of noise, we will detect edges only with higher confidence. In order to make the test completely adaptive, we also must set the edge height parameters of RRO test. The next subsection presents the procedure of adaptive tuning of these parameters.

3.4 Adaptive Tuning of Parameters

In order to drive the parameters, significance level *(α)* and edge height *(δ)*, adaptively, we used a set of synthetic images as training samples. These training samples are generated artificially to cover a wide range of colors and contrasts. We used alphabets (A, B, C, D, G, H, I), lines and squares in these images to cover a wide range of edges in different orientations which may appear frequently in a real image. Figure 3 shows the synthetic images and their corresponding true edge maps used in this study, all of the size 256×256 pixels.

In definition, edge divides two regions. In creating true edge maps for these images we follow the idea that when moving from one region to the next, the bordering pixels we first meet in the first region are edge pixels (outer edges) but the bordering pixels in the new region (inner edges) are not considered as edge pixels.

These training images (Figure 3) are used to derive the functions which determine the optimal values of δ and u_α for various levels of noise. To find the optimal values of δ and u_α, a number of experiments was performed on these images by adding Gaussian noise with different standard deviations ($\sigma_n = 0, 10, 20, 30, 40, 50$ and 60). After finding the edge maps of these images by our proposed edge detector for all possible values of δ and u_α, the error defined in equation (14) is computed for each value. Then for each particular noise level, those values of δ and u_α with lowest error value, will be selected to be applied in RRO test.

$$\text{Error} = \sum_{i\,\in \text{Sample images}} (4 * (\text{number of missed edge points})_i + (\text{number of new faulty edge points})_i) \quad (14)$$

In (14) the missed edges (pixels which are present in true edge map but not in the edge map provided by the proposed edge detector) are 4 times more penalized than the new faulty edge points (pixels which are not in the true edge map of the image but present in the edge map provided by the proposed edge detector). This is the rational that why our edge detector is robust to noise and will remove noisy pixels from edge map in highly corrupted images, though it may lead to missing some weaker edge points, like other existing edge detectors. However, existing detectors may also detect noisy pixels as edge pixels.

Table 1 gives the optimal values calculated for δ and u_α with different levels of additive Gaussian noise. Parameters δ_m and α_m are used to refer to optimal values. As it can be seen in Table 1, these parameters are functions of the noise standard deviation (σ_n). Thus, by knowing the standard deviation of the noise, the optimal threshold values can be determined according to estimation formulas for $\tilde{\delta}_m$ and $\tilde{\alpha}_m$, which are defined as:

Table 1. Optimal values founded for each of the parameters, δ and uα

Standard Deviation (σn)	*Edge height (δ)*	*Threshold (u)*
0	9	6
10	11	1
20	13	1
30	17	1
40	14	3
50	19	3
60	15	10

$$\tilde{\delta}_m\,(\sigma_n) = \text{round}(0.1250 * \sigma_n + 10.25) \quad (15)$$

$$\tilde{\alpha}_m(\sigma_n) = \max(1.704, (0.0058 + 0.0599 * \sigma_n)) \quad (16)$$

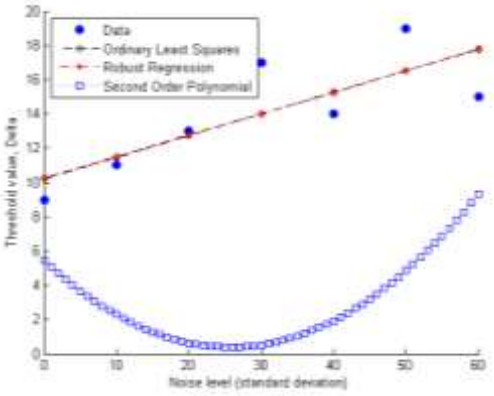
Where σ_n is the estimated standard deviation of noise computed according to (1).

In order to estimate the continuous functions, all of the optimal values at different noise levels are used as sampling points to find the relationships between both δ_m and α_m with σ_n using regression methods, as shown in Figures 4(a) and 4(b). With enough sample points, both functions can be estimated effectively.

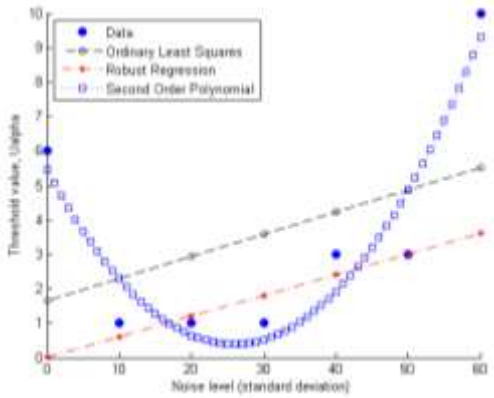
We use two first-order polynomials to estimate $\tilde{\delta}_m$ and $\tilde{\alpha}_m$ according to (15) and (16). We use robust regression [29], which has the minimum mean-square error (LMMSE) [21] in true values of sample points in comparison with other regression methods considered.

It must be noted that for proper edge detection, according to [28] the lowest value of significance level is allowed to be 0.05 which leads to $u_\alpha = 1.704$. Thus the final value for u_α will be the maximum of 1.704 and the one estimated via (16). In other words, in highly corrupted images, only those pixels are labeled edge that we obtain at least 95% confidence about them.

Once the estimated functions are calculated, given an image corrupted by Gaussian noise, the standard deviation of the noise is estimated according to (1). Then, using the estimated functions which are presented in Equations (15) and (16), the proper threshold values are calculated.



(a)



(b)

Fig. 4. Regression plot for (a) δ and (b) u_α.

4. Experimental Results and Discussions

In this section, the performance of our proposed method is compared with five successful detectors, particularly RRO detector [2] which uses the same statistical test as we considered in our study. Since Lim [2] has reported better performance for RRO detector in comparison with two other existing statistical detectors, T detector [20] and Wilcoxon detector [17,20], we only compared our algorithm with RRO detector. We use the 5×5 difference-of-boxes as was shown in Figure 2 for our method as well as for RRO detector. Moreover, we compare our method with well-known edge detectors, Canny [3] and Sobel [5] as well as the recent color edge detector proposed by Russo and Lazzari [23]. Both synthetic and natural images are used in these experiments. The synthetic images allow the use of Pratt's figure of merit [30] (PFOM) as a quantitative evaluation measure, while the natural images might be more feasible for real applications.

We used a synthetic ideal step image that makes the use of PFOM evaluation measure possible; a square synthetic image of the size 256×256 with two vertical and two horizontal edges, as shown in Figure 5. To evaluate the performance of the proposed detector in noisy images, seven different levels of noise (zero-mean Gaussian noise with standard deviations 0, 10, …, 60) are added to the synthetic image. Some of these images are shown in Figure 5 (first column).

Figure 5 also shows the results of applying four different edge detectors to these synthetic images. The results illustrate that the proposed detector performs better in highly corrupted images. In absence of noise, it produces thicker edges compared to Canny and Sobel detectors, the filters that perform poorly in the presence of noise.

It is also observed that our edge detector produces less speckle noise in comparison with RRO detector [2]. From Figure 5, it is also clear that our algorithm performs better than the method of Russo and Lizzari [23], when tested with all noise levels. Our filter not only produces thinner edges but also removes noise and creates much less artifacts.

We adopt the following definition [2] to compute PFOM:

$$R = \frac{1}{\max(I_I, I_A)} \sum_{i=1}^{I_A} \frac{1}{1 + \beta e(i)^2} \quad (17)$$

where I_I and I_A are the number of ideal and actual edge points, respectively, $e(i)$ is the pixel miss distance of the ith pixel which was detected as edge, and $\beta = \frac{1}{9}$ [31] is a scaling constant.

Table 2 illustrates PFOM values of all five edge detectors for a variety of Gaussian noise distributions. We reported the PFOMs of RRO detector for different values of $\delta = 10, 15$ and 20.

The PFOM values show that without any noise, the performance of our detector is the same as Russo and Lazzari's and better than Canny and Sobel detectors, while it is worse than RRO's for all three height parameter values. However, it can be seen that as the standard deviation of noise increases, the performance of our detector improves.

In another experiment, we considered five different images (Lenna, Peppers, F16, Building and Baboon) which are commonly used in literature. Figures 6-10 show the results of applying the above edge detectors on these images for different noise levels. In all of these experiments, the edge height parameter of the RRO detector was set to $\delta = 15$.

As the results indicate, for original noiseless images, Canny detector is the best choice due to its thin and connected results. Considering the quality of detected edges, the second best detector is Sobel; while its performance in terms of detecting connected edges, is quantitatively lower than all other detectors. For original noiseless images, the performance of RRO, Russo and Lazzari's and our detector are not distinguishable.

For higher levels of noise, Canny detector nearly fails; as most of the present noisy pixels are detected as edge pixels. Sobel will also fail, but in a different way; by not detecting most of the true edges. However, the other three edge detectors can find most of the true edges in these situations. Albeit RRO and Russo and Lizzari's detectors will detect most of the noisy pixels as edge pixels as well, and this problem will be emphasized for higher levels of noise. On the contrary, our proposed method performs well on both original and severely noisy images. Our algorithm classifies fewer noisy pixels as edge pixels and focuses on detecting the edges for which we obtain more confidence according to (16).

Table 2. The measured values of PFOM for the edge detectors

Noise STD	Edge Detectors					
	Canny	Sobel	Russo and Lazzari's detector	Δ	RRO	The Proposed
0	0.36869	0.37154	0.70708	10 15 20	**0.87139** **0.87139** **0.87139**	0.7070
20	0.74238	0.38126	0.74653	10 15 20	0.80045 0.83835 0.86129	**0.87042**
40	0.74196	0.62123	0.75389	10 15 20	0.76462 0.79103 0.81360	**0.83688**
60	0.74120	0.80033	0.69580	10 15 20	0.74559 0.77764 0.79784	**0.80394**

5. Conclusion and Future Work

We proposed a new edge detector based on the RRO test, which is a useful alternative to the Wilcoxon test. In order to detect edges in noisy images, we used a multi-pass prefiltering stage to reduce the effect of additive Gaussian noise in each channel of a color image independently. Afterwards, we applied RRO test using a window of size r×r to detect edges in all possible orientations. Since the RRO detector is not adaptive and it needs user specified parameters for detecting edges, we purposed to make this detector adaptive in a systematic approach based on image content.

We investigated the performance of our proposed detector in comparison with well-known Canny and Sobel edge detectors as well as RRO and Russo and Lazzari's detectors for both synthetic and natural images. From the experimental results, it was observed that in highly corrupted images, the proposed edge detector performs much better than other edge detectors both quantitatively and qualitatively in a completely adaptive manner.

For future works, we can modify the RRO test statistic to handle R, G and B channels, simultaneity. Furthermore, it is interesting to introduce a prefiltering which is able to reduce the effect of different types of noise, such as impulsive noise. However, the current proposed detector can handle all types of noisy images but its performance is significantly better for Gaussian noise. Also, we intend to investigate the performance of proposed method in HIS color space. Our preliminary results on HIS color space are promising.

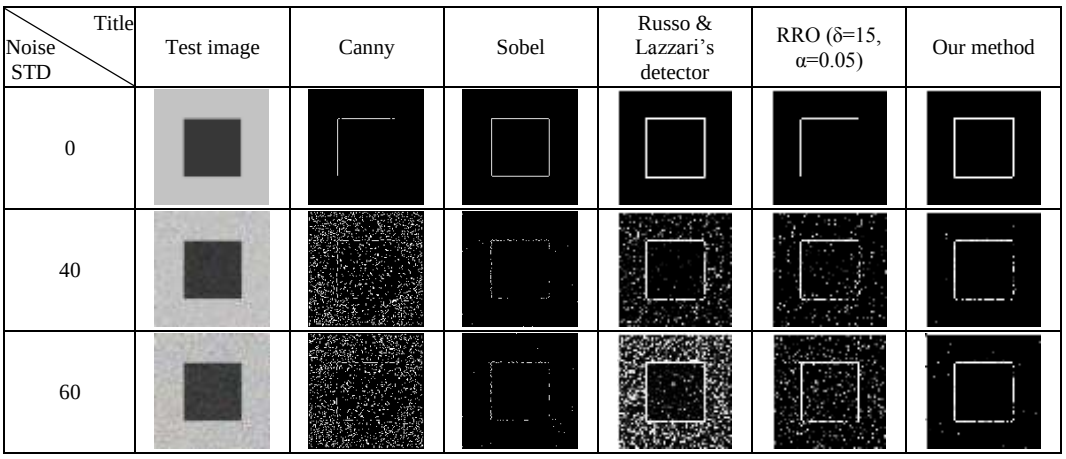


Fig. 5. Edge detector results for square synthetic images.

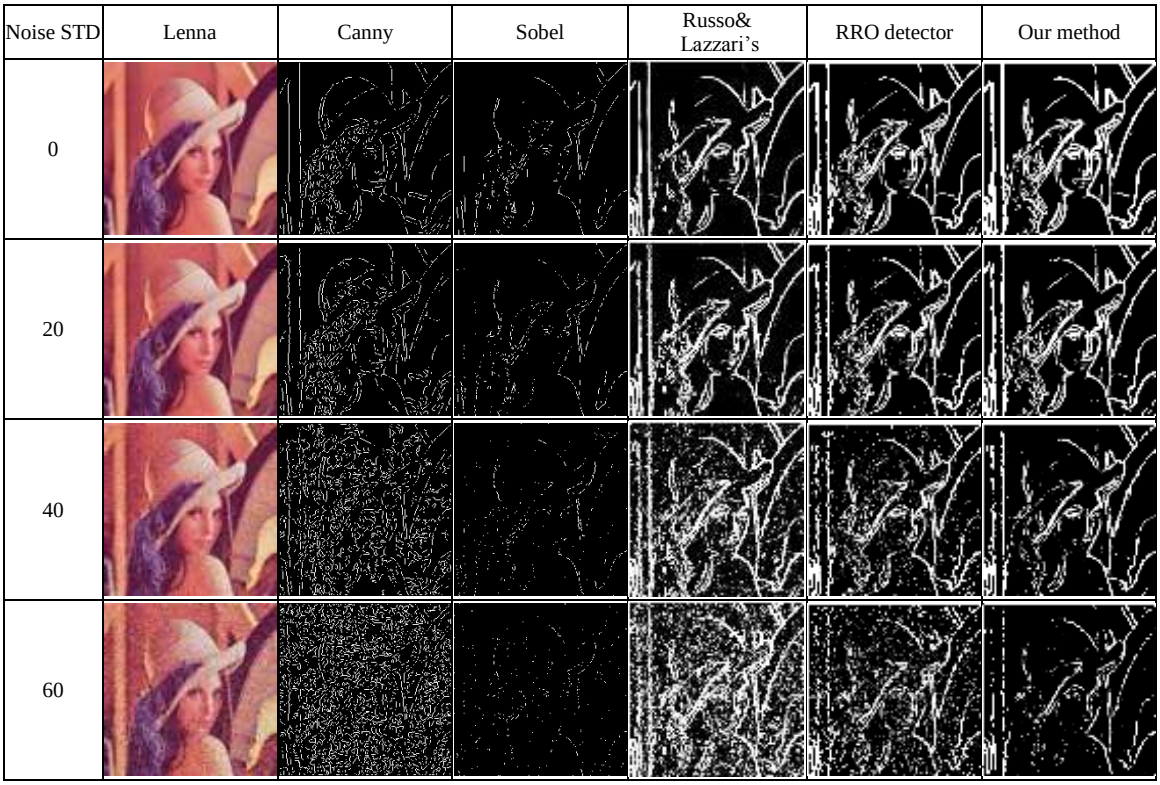


Fig. 6. Edge detection results for Lenna image.

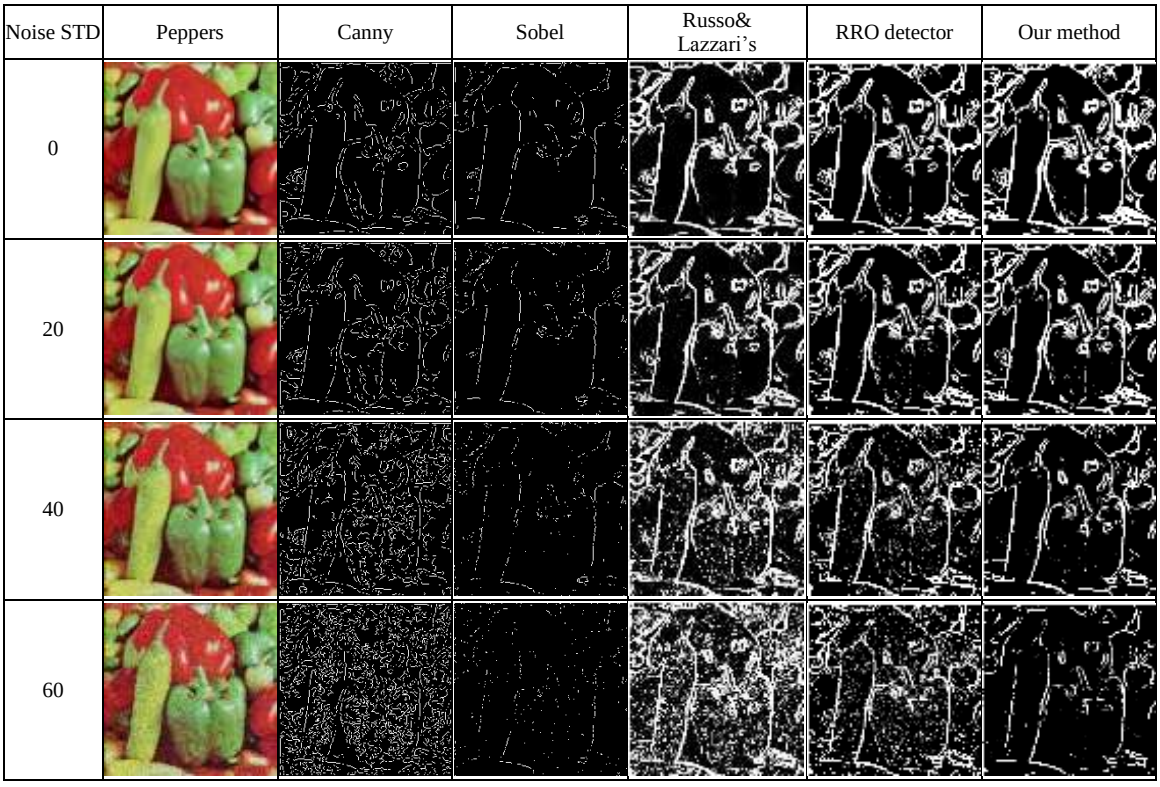


Fig. 7. Edge detection results for Peppers image.

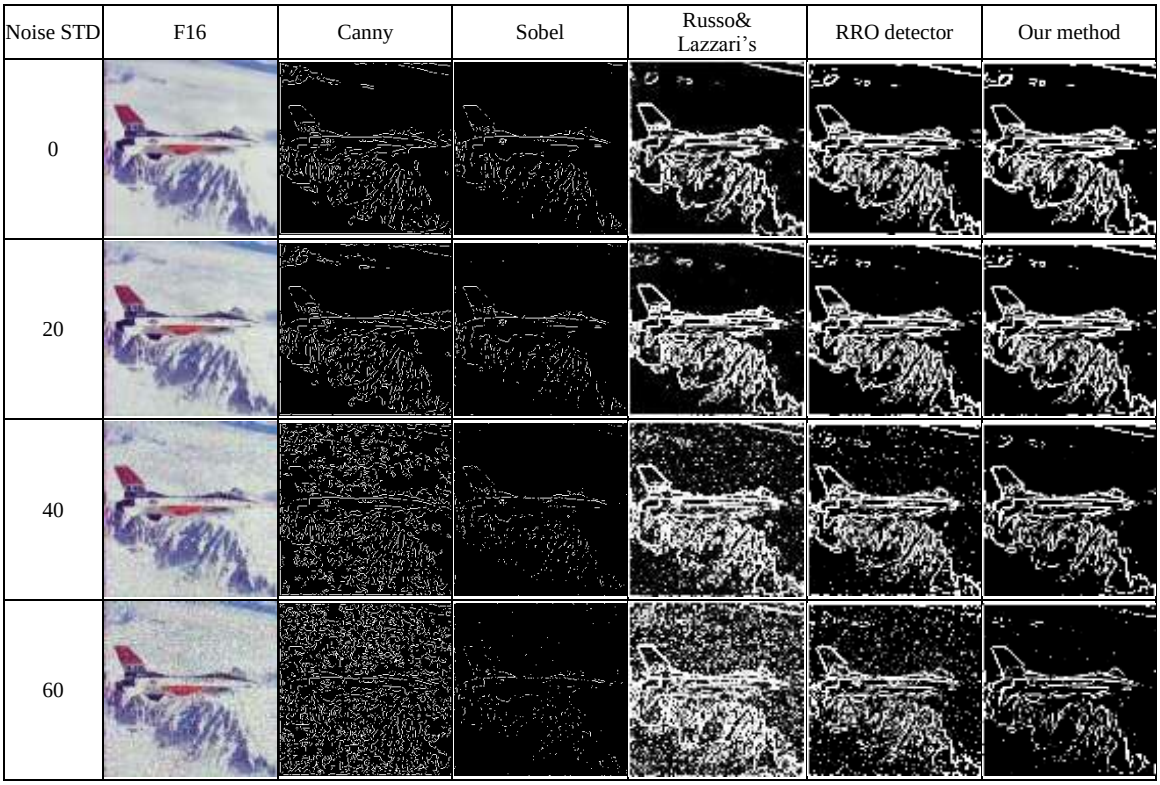


Fig. 8. Edge detection results for F16 image.

Fig. 9. Edge detection results for Building image.

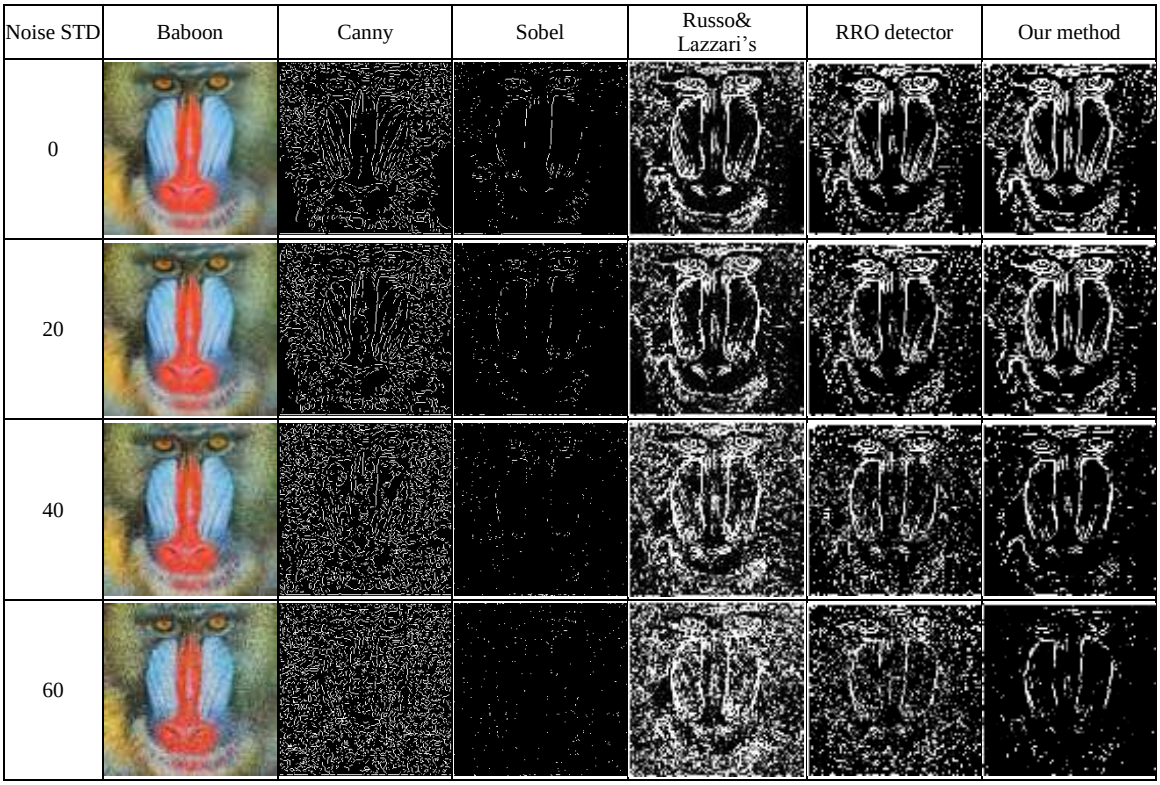


Fig. 10. Edge detection results for Baboon image.

References

[1] C. L. Novak, and S. A. Shafer, "Color edge detection", In proc. of DARPA image understanding workshop 1, 1987, pp. 35-37.

[2] D. H. Lim, "Robust edge detection in noisy images", Computational Statistics and Data Analyses, Vol. 50, No. 3, 2006, pp. 803-812.

[3] J. Canny, "A computational approach to edge detection", IEEE Trans. Pattern Anal. Machine Intell. PAMI, vol. 8, No. 6, 1986, pp. 679–698.

[4] M. Sharifi, and M. Fathy, and M. T. Mahmoudi, "A classified and comparative study of edge detection algorithms", in Proceedings of International Conference on Information Technology: Coding and Computing, 2002, pp. 117–120.

[5] R. C. Gonzalez, and R.E. Woods, Digital Image Processing, third ed., Addison-Wesley, NewYork, 1992.

[6] A. Rosenfeld, and A. Kak, Digital Picture Processing, Second ed., Academic Press, New York, 1982.

[7] M. D. Heath, and S. Sarkar, and T. A. Sanocki, and K. W. Bowyer, "Robust Visual Method for Assessing the Relative Performance of Edge-Detection Algorithms", IEEE Trans. Pattern Anal. Mach. Intell., Vol. 19, No. 12, 1997, pp. 1338-1359.

[8] Q. Ying-Dong, and C. Cheng-Song, and C. San-Ben, and L. Jin-Quan, "A fast subpixel edge detection method using Sobel–Zernike moments operator", Image and Vision Comput., Vol. 23, No. 1, 2005, pp. 11-17.

[9] M. Basu, "Gaussian-Based Edge-Detection Methods—A Survey", IEEE Transactions on Systems, Man, and Cubernetics—Part C: Applications and Reviews, Vol. 32, No. 3, August 2002.

[10] E. Nadernejad, and S. Sharifzadeh, and H. Hassanpour, "Edge detection techniques: evaluations and comparisons". Applied Mathematical Sciences, Vol. 2, No. 31, 2008, pp. 1507-1520.

[11] N. Senthilkumaran, and R. Rajesh, "Edge detection techniques for image segmentation - a survey of soft computing approaches", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, 2009, pp. 250-254.

[12] O. P. Verma, et al. "A novel fuzzy system for edge detection in noisy image using bacterial foraging", Multidimensional Systems and Signal Processing, Vol. 24, No. 1, 2013, pp. 181-198.

[13] M. Setayesh, and M. Zhang, and M. Johnston, "Edge detection using constrained discrete particle swarm optimisation in noisy images", IEEE Congress on Evolutionary Computation (CEC), 2011.

[14] M. Setayesh, and M. Zhang, and M. Johnston, "Investigating particle swarm optimisation topologies for edge detection in noisy images", AI 2011: Advances in Artificial Intelligence. Springer Berlin Heidelberg, 2011, pp. 609-618.

[15] H. Li, , X. Liao, Ch. Li, H. Huang, and Ch. Li. "Edge detection of noisy images based on cellular neural networks." Communications in Nonlinear Science and Numerical Simulation vol. 16, no. 9, 2011, pp. 3746-3759.

[16] M. A. El-Sayed, Y. A. Estaitia, and M. A. Khafagy. "Automated Edge Detection Using Convolutional Neural Network." International Journal of Advanced Computer Science & Applications (IJACSA) vol. 4, no. 10, 2013, pp. 11-17.
[17] A.C. Bovik, and T.S. Huang, and D.C. Munson, "Nonparametric tests for edge detection in noise", Pattern Recogn, Vol. 19, No. 3, 1986, pp. 209–219.
[18] J. Aron, and L. Kurz, "Edge detection using ANOVA techniques", Int. Symposium on Inf. Theory, Israel, 1973.
[19] L. Kurz, and M.H. Benteftifa, Analysis of variance in statistical image processing, Cambridge University Press, Cambridge, 1997.
[20] D. H. Lim, and S. J. Jang, "Comparison of two-sample tests for edge detection in noisy images", J. R. Stat. Soc. Ser. D-Statist, Vol. 51, No. 1, 2002, pp. 21–30.
[21] A. Koschan, "A Comparative study on color edge detection", In Proc. 2nd Asian Conf. on Computer Vision, Vol. 3, 1995, pp. 574-578.
[22] A. Mittal, S. Sofat, and E. Hancock. "Detection of edges in color images: A review and evaluative comparison of state-of-the-art techniques." Autonomous and Intelligent Systems, Springer Berlin Heidelberg, 2012, pp. 250-259.
[23] F. Russo, and A. Lazzari, "Color edge detection in presence of Gaussian noise using nonlinear prefiltering", IEEE Trans. On Instrum. And Meas., Vol. 54, No. 1, 2005, pp. 352-358.
[24] A. N. Evans, "Nonlinear Edge Detection in Color Images", in Advances in Nonlinear Signal and Image Processing, edited by: Stephen Marshall and Giovanni L. Sicuranza. EURASIP Book Series on Signal Processing and Communications, Vol. 6, 2006, pp. 329-356.
[25] F.V. Heijden, Image-Based Measurement Systems, Wiely, New York, 1994.
[26] F. Russo, "A method for estimation and filtering of Gaussian noise in images", IEEE Trans. On Instrum. and Meas. Vol. 52, 2003, pp. 1148-1154.
[27] A. Pizurica, and W. Philips, and I. Lemahieu, and M. Acheroy, "A joint interand intrascale statistical model for Bayesian wavelet based image denoising", IEEE Trans. On Image Processing, Vol. 11, No. 5, 2002, pp. 545-557.
[28] M. A. Fligner, and G. E. Pollicello, "Robust rank procedures for the Behrens–Fisher problem", J. Am. Stat. Assoc., Vol. 76, 1981, pp. 162–168.
[29] P. W. Holland, and R. E. Welsch, "Robust regression using iteratively reweighted least-squares", Commun. in stat.: theory and methods, Vol. 6, pp. 813-827.
[30] W. Pratt, Digital Image Processing,Wiley, NewYork, 1978.
[31] D. Malah, and T. Peli, "A study of edge detection algorithms", Computer Graphics Image Process., Vol. 20, 1982, pp. 1-21.

Mina Alibeigi received her B.Sc degree in Software Engineering (2008) from Shiraz University and M.Sc degree in Artificial Intelligence (2010) from that university. She is currently a Ph.D student in Machine Intelligence and Robotics at University of Tehran. Her research interests include imitation learning, human-computer interaction, data mining and image processing.

Niloofar Mozafari received the B.Sc degree in Software Engineering (2007) from Shiraz University and M.Sc degree in Artificial Intelligence (2010) from that university. She is currently Ph.D student in CSE and IT department of Shiraz University, Iran. Her area research interests include social network analysis, image processing and data mining.

Zohreh Azimifar received the B.Sc degree in Computer Science and Engineering from Shiraz University, Shiraz, Iran, in 1994, and her Ph.D degree in Systems Design Engineering from the University of Waterloo, Waterloo, Canada, in 2005. In 2005, she was a postdoctoral fellow in medical biophysics at the University of Toronto, Toronto, Canada. Since 2006 she has been a faculty member and director of computer vision and pattern recognition lab in computer science and engineering at Shiraz University. Currently, Dr Azimifar is chair of the Shiraz university research center for intelligent image and vision processing. Her research interests include statistical pattern recognition, machine vision and cognitive vision.

Mahnaz Mahmoodian received B.Sc degree in Computer Engineering (2008) from Shiraz University and M.Sc degree in Computer Engineering, Artificial Intelligence (2012) from University of Tehran. She is currently working at Traffic and Transportation Research Center (TTRC), Shiraz University as a researcher. Her current fields of interest are Machine Vision, Image Processing and Intelligent Transport Systems (ITS).