Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 2789 – 2805
© TÜBİTAK
doi:10.3906/elk-1403-286

A new approach for edge detection in noisy images based on the LPGPCA technique

Şahin IŞIK, Kemal ÖZKAN*
Computer Engineering Department, Eskişehir Osmangazi University, Eskişehir, Turkey

Received: 27.03.2014 • **Accepted/Published Online:** 06.12.2014 • **Final Version:** 15.04.2016

Abstract: In this study, a new approach to edge detection on images, corrupted with Gaussian and impulsive noise, is presented. The concept, under the decomposition of image with its principal component being an analysis on local pixel grouping for noise suppression, called LPGPCA based denoising, is adopted in order to obtain noiseless gradient maps for edge extraction. As a result, an algorithm has been developed called LPGPCA-ED. Firstly, horizontal and vertical gradient images are computed; then the gradient images are decomposed into a noiseless phase by applying the LPGPCA algorithm. Once a single gradient map has been obtained, a smart nonmaximum suppression operation is carried out to obtain a binary edge map. To show the accuracy of the proposed edge detector objectively, F-measure and PFOM results of the proposed edge detector on images with different Gaussian and impulsive noise are compared with the results of traditional and certain recently published edge detectors. Objectively, the experimental results on RUG and receiver operating characteristic (ROC) curve databases show that our method has better performance than other corresponding edge detectors. Moreover, subjective experiments on a variety of noise contaminated images show that the LPGPCA-ED algorithm is more robust under high level noise conditions, and is able to reveal well-linked lines and also preserve the structural form of a processed image.

Key words: Edge detection, LPGPCA, noise suppression, nonmaximum suppression

1. Introduction

Edge detection is an important and fundamental task in computer vision and image processing. The global definition of an edge is the boundary between two different regions where intensity abruptly changes caused by geometric and nongeometric events. Some examples for such geometric events are discontinuity in depth, texture, surface color, and orientation, while nongeometric events can be summarized as mirror reflections, shadows, and interre?ections along a specific direction for an image. Constitutively, the main aim behind edge detection is the extraction of important features from the edges of an image (such as lines, corners, and curves) for computer vision tasks, such as measuring the size, shape, and location of objects to facilitate registration [1], restoration [2], and classification [3]. Moreover, the edge detection process provides sufficient information about an image by reducing the amount of presence data by transforming it into more suitable formats in terms of computation time. Since the performance of further stages depends on the characteristics of edge detection, the obtained edge segments (edgels) should be in a continuous, thin, and well-localized format [4]. Although the traditional gradient based operators present good performance in noiseless images, they usually fail in the presence of noise. This is because the variation in intensity occurred from various sources, such as the effects

*Correspondence: kozkan@ogu.edu.tr

of size, heat, and the level of the fill factor of preferred sensors in cameras. To address the limitations of edge detectors that are sensitive to noise, several edge detection algorithms have been proposed to detect edges in noisy images [5] such as Gaussian-based [6], statistical-based [7], and image transform-based [8].

Several statistical-based methods have been developed to determine edges in noisy images. The best known are edge detectors based on the Wilcoxon test [9], median test [10], t-test [11], and robust rank-order (RRO) test [12]. Recently, new ones have been developed, such as adaptive statistical thresholding (AST) using the gray level co-occurrence matrix (GLCM) in order to determine the threshold value in the wavelet domain ()[13]. In addition, a local thresholding method that depends upon the statistical variability of the gradient vector at a pixel is utilized to determine the eligibility of a given pixel to be an edge pixel [14]. Another proposed method is a robust statistic method [7] that at first determines an edge structure with a robust one-way model and then localizes it by a contrast test.

Moreover, a variety of edge detection methods attempt to detect edges by utilizing a transform function on images that are corrupted with a large amount of noise. This group is known as scale space theory based edge detectors [5]. Some examples for this group are the wavelet transform method [15], mathematical morphological method [16], fuzzy method [17], neural network method [18], and genetic algorithm [19]. Unfortunately, they are expensive and are sensitive to noise. In addition, the obtained map includes jagged and broken edges. Moreover, there are no general edge detection methods that perform well in all contexts and stages; it is widely accepted that different edge detectors are better suited to different tasks.

In another work [5], the particle swarm optimization (PSO) algorithm was introduced by Kennedy and Eberhart in 1995 as a computational method for solving global optimization problems, and was designed for edge detection in noisy images. To obtain thin edges instead of broken and jagged edges, two PSO-based algorithms are used for the optimization of the proposed model. In simple terms, one of the PSO-based algorithms maximizes the distances between pixels in two regions (interset distance) and minimizes the distance between the pixels within each region (intraset distance) by determining an area to include the best curve as a preservation method. The second PSO-based algorithm handles constraints by adopting nonstationary and multistage penalty fitness functions until an optimized particle (curve) is obtained, based on a stopping criterion. Although the results seem good, the algorithm suffers from a long running time.

Furthermore, a new noise-robust edge detector (NRED) [20] has been developed to obtain edge maps by combining a small-scale isotropic Gaussian kernel and large-scale anisotropic Gaussian kernels (ANGKs). The local directional variation is determined by deriving the anisotropic directional derivatives (ANDDs) from ANGKs. Next, a noise-robust ANND-based edge strength (gradient) map (ESM) is constructed, considering the scale alone and its edge resolution by the ratio of the scale to the anisotropic factor. The edge stretch effect in anisotropic smoothing is also identified. Later, a fused noise-robust ESM with high edge resolution and limited edge stretch is obtained by combining the ANDD-based ESM and the gradient-based ESM with a small-scaled isotropic Gaussian kernel. Finally, a binary edge map is acquired by conducting the Canny method on fused ESM.

The question is how to develop an edge detector method that is robust, insensitive to noise, and has the ability to extract a high quality edge map from noisy images. For this purpose, we have proposed a new edge detector method, called LPGPCA-ED, that produces thin, clear, and continuous edges even if the pixels of image are corrupted and deformed by a higher noise level. We are inspired by the 'two-stage image denoising by principal component analysis with local pixel grouping' idea, abbreviated to LPGPCA in the study [21]. According to the referred to study, a local pixel grouping process should be performed prior to transformation with PCA. To achieve this, similar pixels are first grouped in order to form a homogeny class. Next, a PCA transformation is applied to the homogeneous region. In this way, the image is represented with noisy PCA coefficients. Therefore, the transformed domain is denoised with the linear minimum mean square-error estimation (LMMSE) in order to regulate noisy pixels. After the denoising procedure, the noiseless data are again reconverted into the image domain using the same PCA transformation. As a result, noiseless horizontal and vertical gradient maps of processed images are acquired by applying the obtained PCA transformation. Later, the obtained noiseless images are put forward as input to the edge detection stage where nonmaximum suppression is carried out. This procedure facilitates the extraction of clear continuous edges, and the preservation of the structure of the objects situated in the strong noise image, by adapting a noise removal method called the LPGPCA procedure.

2. LPG-PCA-ED edge detector

2.1. Obtaining training set with local pixel grouping

The noisy image can be expressed as $I_v = I + v$, where I is the original noiseless image and v is the imposed additive white Gaussian noise with zero mean and σ standard deviation. It is assumed that the image I and noise v are uncorrelated. The main aim of a denoising algorithm is to estimate the noiseless image from the noisy image using certain information obtained from the image itself and the characteristics of the noise. Since the edges in the image are the meaningful components, most image denoising algorithms attempt to protect them from the negative effects of the denoising process. Although the pixels in the same region exhibit similar characteristics on certain properties, such as color, obtained from the image itself and the characteristics of the noise, adjacent regions differ significantly from each other. Therefore, it is vital to apply the local pixel grouping technique before the denoising procedure. Since the noise is assumed to be uncorrelated (location invariant and ergodic), a similar expression for the noisy image patch can be written as $\vec{a}_v = \vec{a} + v$. The image that is presented in Figure 1 is taken from the indicated reference [22]. As demonstrated in Figure 1, in order to denoise a specific pixel, a KxK window is centered on noisy image I_v and expressed as a column vector $\vec{a}_v = [a_1...a_n]^T$, where $n = K^2$. This window is moved over the LxL patch to construct a training set, where $L > K$ and a_v is at the center of that window, as shown in Figure 1. Thus, a total of $(L - K + 1)^2$ training samples are extracted for each patch of $\vec{a}_v$. Unfortunately, for the regions that contain edges, patches of size KxK within LxL windows would resolve to an incorrect covariance matrix because the patches would be dissimilar, causing the patches to belong to different classes. Therefore, the patches that are similar to the moving window are selected to build the class. This is a classification problem that can be solved using various techniques, such as block-matching or K-means clustering. Since the actual intention is not the classification, the block matching technique is employed for grouping similar blocks.

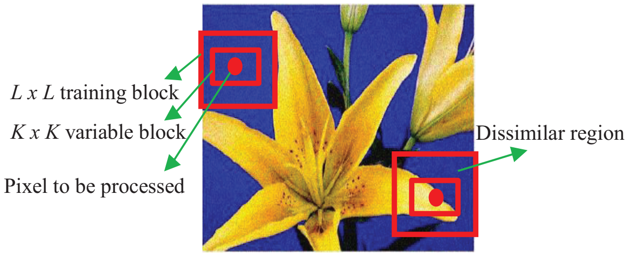

Figure 1. Illustration of the modeling of a class selection.

2.2. LPG-PCA procedure

To make a decision about whether a pixel is an edge or not, a KxK region (in this study $K = 3$, $L = 7$) around the pixel is determined and the pixels substituted in this region are shown in vector form as $\vec{x}$. As mentioned above, the training blocks are constructed from the LxL around the region in the neighborhood of the processed pixel. Later, the blocks that are similar to the $\vec{x}$ vector are extracted from the obtained training blocks using the LPG and are combined together to form the X matrix. If the constructed X matrix is corrupted with noise then it can be written as

$$X_V = X + V \tag{1}$$

If our goal was to eliminate the noise, then X_V should be centralized as emphasized in reference work [21]. According to the referred to work, the centralized data are in the form of $Z_c = X_V - \mu_{X_V}$ and can be written as $Z_c = (X - \mu_x) + (V - \mu_V)$. In addition, the centralized matrix can be expressed as $Z_c = X_c + V$ by accepting the noise with a zero mean. Here X_c refers to the form of the centralized noiseless data. However, since our main aim is edge detection and not denoising, we do not have to centralize the input matrix. For this reason, only the covariance matrix of X, denoted by Ω_X, is computed and the PCA transformation matrix P_X is obtained with respect to Ω_X. Since the available dataset X_V is corrupted with noise, Ω_X cannot be directly computed. Therefore, the covariance matrix of noisy data Ω_{X_v} should be computed without centralizing the dataset X_V and can be formulized as

$$\Omega_{X_V} = \frac{1}{n} X_V X_V^T = \frac{1}{n}\left(XX^T + XV^T + VX^T + VV^T\right) \tag{2}$$

Herein, because X and V are uncorrelated, the items XV^T and VX^T will be almost zero matrices and thus

$$\Omega_{X_V} \approx \frac{1}{n}\left(XX^T + VV^T\right) = \Omega_X + \Omega_V, \text{ where } \Omega_X = (1/n)\, XX^T \text{ and } \Omega_v = (1/n)\, VV^T. \tag{3}$$

The components of the covariance matrix $\Omega_V(i,j)$ show the correlation between v_i and v_j. Since v_i and v_j are uncorrelated for $i \neq j$, it is expected that Ω_v is an MxM diagonal matrix and the diagonal components should constitute σ^2. With respect to this, Ω_v can be written as $\sigma^2 I$, where I is the identity matrix. Intuitively, it can be said that the PCA transformation matrix P_X associated with Ω_X is the same as the PCA transformation matrix associated with Ω_{X_V}. This is proved by later equations. Therefore, Ω_X can be decomposed as

$$\Omega_X = \Phi_X \Lambda_X \Phi_X^T, \tag{4}$$

where Φ_X is the MxM orthonormal eigenvector matrix and Λ_X is the diagonal eigenvalue matrix. Since Φ_X is an orthonormal matrix, we can write Ω_V as

$$\Omega_V = \left(\sigma^2 I\right) \Phi_X \Phi_X^T = \Phi_X \left(\sigma^2 I\right) \Phi_X^T \tag{5}$$

Thus we have

$$\begin{aligned} \Omega_{X_V} = \Omega_X + \Omega_V &= \Phi_X \Lambda_X \Phi_X^T + \Phi_X \left(\sigma^2 I\right) \Phi_X^T \\ &= \Phi_X \left(\Lambda_X + \sigma^2 I\right) \Phi_X^T = \Phi_X \Lambda_{X_V} \Phi_X^T \end{aligned} \tag{6}$$

where $\Lambda_{X_V} = \Lambda_X + \sigma^2 I$. When Eq. (17) is observed, we can infer that Ω_{X_V} and Ω_X have the same eigenvector matrix Φ_X. Therefore, practically, we can directly compute Φ_X by decomposing Ω_{X_V} instead of Ω_X, and then the orthonormal PCA transformation matrix for X is set as

$$P_X = \Phi_X^T \tag{7}$$

Applying P_X to dataset X_V, we have:

$$Y_V = P_X X_V = P_X X + P_X V = Y + V_Y, \tag{8}$$

where $Y = P_X X$ is the decorrelated dataset for X and $V_Y = P_X V$ is the transformed noise dataset for V. Since Y and noise V_Y are uncorrelated, we can easily derive that the covariance matrix of Y_V is

$$\Omega_{Y_V} = \frac{1}{n} Y_V Y_V^T = \Omega_Y + \Omega_{V_Y}, \tag{9}$$

where $\Omega_Y = \Lambda_X$ is the covariance matrix of decorrelated dataset Y. As a result, the covariance matrix of noise dataset V_Y becomes $\Omega_{V_Y} = P_X \Omega_V P_X^T$.

Thus, most energy of noiseless dataset Y concentrates on the several most important components of the PCA transformed domain Y_V, whereas the energy of noise V_Y distributes much more event. The noise in Y_V can be eliminated using the LMMSE technique.

2.3. Edge detection based on LPGPCA

To explain the algorithm in more detail, we can illustrate a simple algorithm for edge detection with the LPGPCA denoising approach. Considering Figure 2, the algorithm can be summarized as follows:

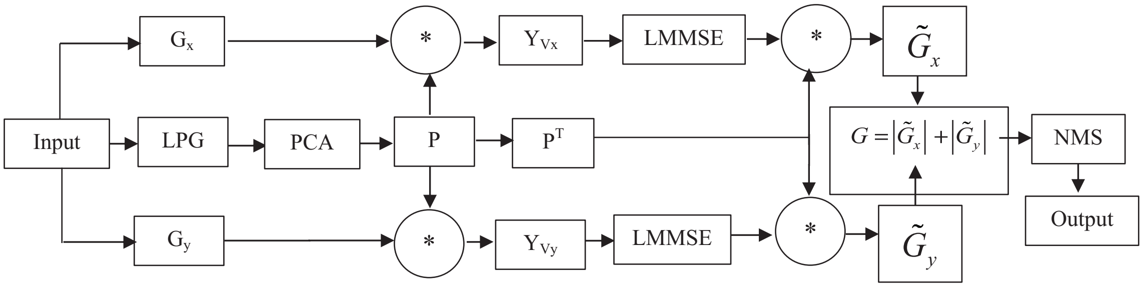

Figure 2. An overview of the proposed algorithm.

(1) Firstly, for each noisy input block, the LPG procedure with PCA is employed in order to transform into the PCA domain. Hence, the projection matrix and inverse projection matrix are obtained.

(2) Next, the reference block is decomposed into two distinct gradients using the Canny's derivative operator, horizontal (G_x) and vertical (G_y), respectively.

(3) The G_x and G_y gradients are transformed into the PCA domain by multiplication with the projection matrix.

(4) In the PCA domain, the transformed gradient coefficients are denoised using the LMMSE technique and the denoised domain is reconverted into an image domain where the noiseless form of reference image is presented.

(5) Later, the magnitude or edge strength of the gradients is estimated using the formula $G = \left|\tilde{G}_x\right| + \left|\tilde{G}_y\right|$.

(6) Finally, in the postprocessing stage, nonmaximum suppression is carried out, based on an optimized threshold selection method that is presented in Ray's work .[23].

As noted above, the fundamental contribution of this study is in determining edges in a noisy image by adopting a denoising method, named LPG-PCA and used in the reference work [21]. For this purpose, the projection matrix of the input block is first obtained. Then the vertical (G_y) and horizontal (G_x) derivatives of the input image are computed by operating Canny's edge operator. The obtained orthonormal PCA transformation (P_X) from the noisy block is applied to both G_x and G_y, so that the horizontal and vertical gradient maps are transformed into the PCA domain. To suppress noise from these gradient maps, the LMMSE technique is performed in the PCA domain and, by using the transpose of the same PCA transformation, the denoised data are reconverted into the image domain. In this way, the noiseless horizontal ($\tilde{G}_x$) and vertical ($\tilde{G}_y$) derivative matrices are obtained in order to extract a noisy free edge map.

The main reason for choosing the LPGPCA algorithm for edge detection in a noisy image can be expressed with the special characteristic of the algorithm for preserving the edges by eliminating the noise in vectoral form instead of handling single pixels. In this particular way, a model from the nearest neighbors of the processed pixel is constructed as

a vectoral form and the noise level is minimized by applying a noise reduction method. With regard to this property, we have endeavored to make certain improvements to the algorithm for acquiring a noiseless gradient map. As a significant contribution to the proposed method, we decided to ignore the centralizing procedure of the input matrix as fulfilled in the LPGPCA algorithm. The other modification is that the information stated on the noisy derivative is reduced by conducting the LMMSE technique on projected horizontal and vertical gradients. The LPG procedure is only utilized to obtain the projection matrix. For this reason, the modified version of the algorithm could not be considered as a noise removal method for the proposed work.

Once the denoised gradient maps are obtained, the gradient magnitude G at a pixel is then obtained by the formula $G = \left|\tilde{G}_x\right| + \left|\tilde{G}_y\right|$. The next step of the LPGPCA is to extract edge points using a nonmaximum suppression procedure based on an optimal threshold selection method. For this purpose, we used an optimized threshold, proposed in Ray's work [23], based on two parameters: the total number of obtained pixels after NMS and the total number of pixels in the gray image.

$$D_E = \frac{\text{Total Number of Black Pixels after NMS}}{\text{Total Number of Pixels in the Image}} \tag{10}$$

To remove the hysteresis thresholding that was used in Canny's work, a new approach has been put forward to estimate an automatic optimized threshold, as given in Eq. (21), according to Ray's work. Furthermore, it is emphasized that the proposed approach has certain desirable advantages. Firstly, it has the ability to increase/decrease monotonically or remain as it is, and secondly, it can be easily modified according to the desired connectivity and edge segment's size. In Ray's work, a histogram of black pixels, varying with respect to a series of threshold values D_E, starting from min to max (e.g., 0 to 255), was plotted in order to optimize the threshold value. As is the nature of the global threshold, it can be observed that the percentage of black pixels decreases when the value of the threshold is increased. Visually, a marked rapid decline is recognized in the amount of detail available in the edge map with an increasing threshold and, beyond a certain level, almost no usable detail is available. To alleviate such problems, it has been considered that standardizing the data obtained with D_E is necessary. The formula in Eq. (22) is standardized using the concept of the z-score normalization:

$$Z = \frac{x-\mu}{\sigma} \tag{11}$$

In Eq. (22), Z and x refer to the standardized and prestandardized data, while μ and σ indicate the sample mean and standard deviation. The values of μ and σ are calculated by taking data until $D_E >= 1$. The generated edge map reveals that the general structure of real images and edges are well localized for linking. Moreover, the generated edge map obtained from the aforementioned threshold can easily be incorporated with two considerations: connectivity and object size. An overview of the nonmaximum suppression that is used in this work is given in Eq. (23).

$$\begin{aligned} \forall\left(i,j\right), I\left(i,j\right) &= no\_\text{ edge if} \begin{cases} I\left(i,j\right) < NMS\_\text{Threshold} \\ I\left(i,j\right) < I\left(i+n_x, j+n_y\right) \quad \forall\left(i,j\right), I\left(i,j\right) \\ I\left(i,j\right) < I\left(i-n_x, j-n_y\right) \end{cases} \\ &= \text{ edge otherwise where } -1 \leq n_i \leq 1, i = x, y \end{aligned} \tag{12}$$

From Eq. (23), we can easily derive that the performance of the proposed method depends on the threshold selection stage. This is because threshold selection directly affects the performance of an algorithm and also plays a major role in the emergence of satisfactory results. By looking at the objective and subjective experimental results, we noted that the performed optimized threshold is more than adequate for this study.

Using the LPGPCA procedure prior to NMS thresholding is a crucial step in order to obtain better performance and to reduce the running time in the case of edge determination. Furthermore, the performance of algorithms such as Canny depends on an applied and selected NMS procedure. While these methods realize an edge linking procedure in order to connect edge pixels one by one, we can ignore this step and find the obtained results sufficient for the proposed method.

3. Experimental results

Although many measures have been proposed to evaluate the performance of edge detection, it still remains an open problem due to the fact that there is no a generally agreed definition of ‘edge’. For example, Canny considers three criteria, min error rate, well localization, and uniqueness, for assessment of his edge detection performance. According to the survey performed by the referred to work [24], the proposed measures for the evaluation of the performance of an edge detector can be divided into three different groups: quantitative, qualitative, and hybrid measures. The first group uses objective measures to compute performance by assigning one or more numerical values to an edge image, whereas the second group makes a subjective comparison depending on human interpretation. A third group emerges from a combination of the first and the second group.

To make a benchmark evaluation, the quantitative and qualitative measures are used for comparison purposes. For this purpose, firstly a subjective evaluation is carried out by comparing the visual outputs with the NRED method. Later, a quantitative evaluation is performed with traditional edge detectors and the NRED method using F-measure metric. Finally, another quantitative evaluation is performed with PSOs, Canny, and NRED edge detectors using Pratt’s figure of merit (PFOM) metric [25].

For all the experiments, the parameters of PSO1, PSO2, RRO, and Canny are considered as explained in the study of the PSO edge detector [5]. Moreover, the results of the NRED method are obtained by considering the optimal parameters in the referred to paper [20]. In case of the NRED method, the scale of the anisotropic Gaussian kernels is 16, low and high thresholds are 0.5 and 0.70, and the number of anisotropic directional derivative filters is 16. For the noise-free images, the standard deviation of Gaussian white noise is considered as 0, but for noisy images the level of imposed noise is forwarded as a noise ratio into the algorithm.

It should be noted that all the experiments given in this paper were run on the same hardware (Intel core i5-3210M with 2.5 GHz CPU and 4 GB memory) with software implemented on MATLAB 2013b. Here the standard MATLAB implementations of Canny, Sobel, and Prewitt algorithms are performed.

3.1. Qualitative evaluation with traditional methods

A set of subjective results, obtained through the proposed and NRED methods, over ten randomly chosen images from the website of the LSD [26] database (http://demo.ipol.im/demo/gjmr_ line_ segment_ detector/archive), are presented in Figures 3 and 4 in terms of visual evaluation. One important condition is that a developed edge detection method should have a strong and comprehensive capability of working for all types of image.

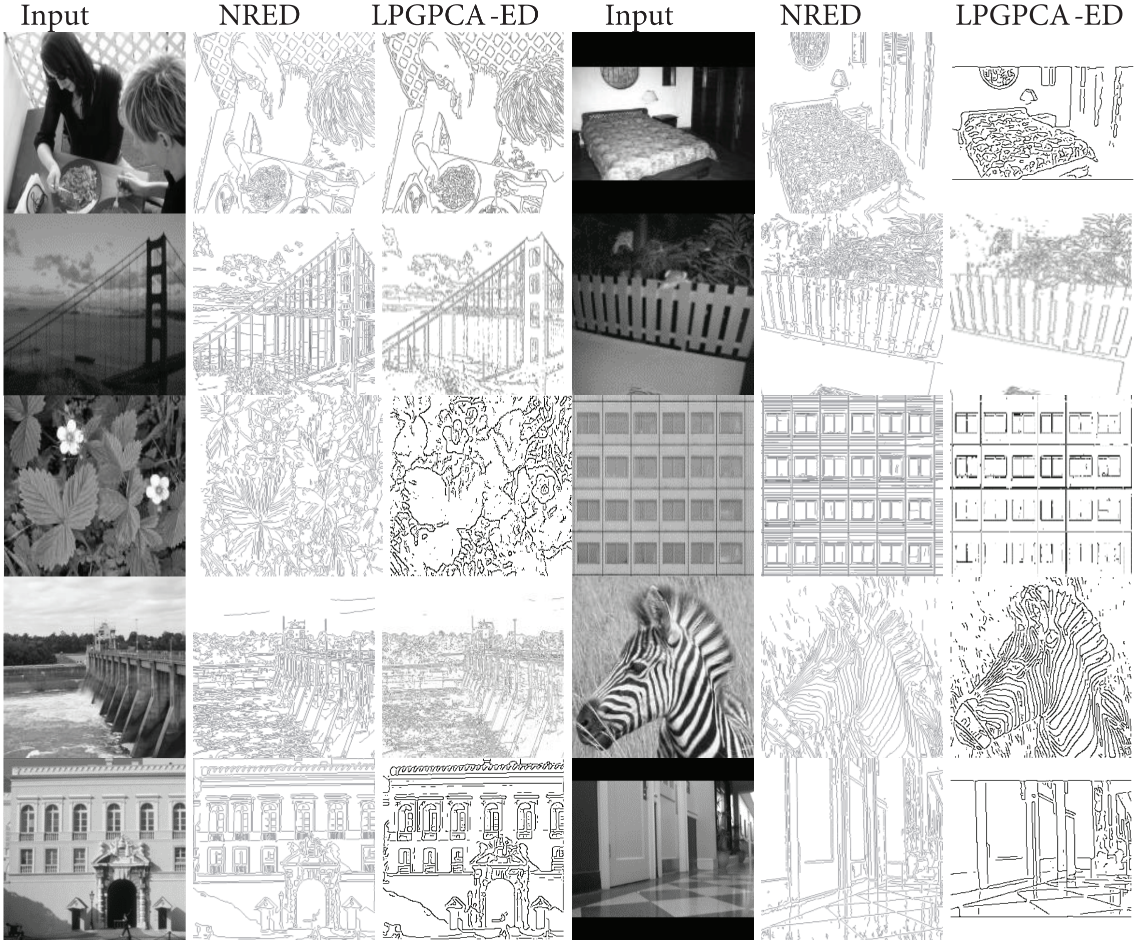

Figure 3. The ten randomly chosen noiseless images.

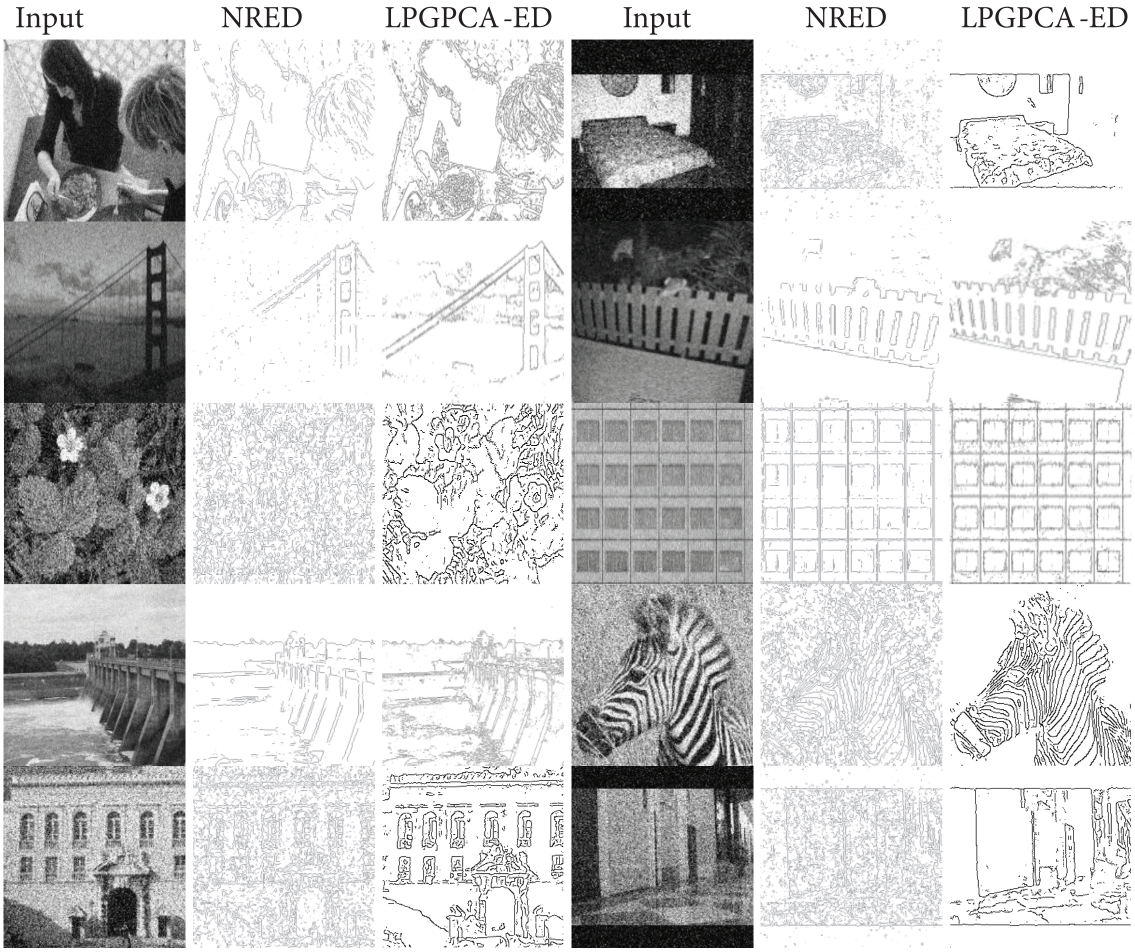

Figure 4. The edge maps obtained from noisy versions of the images given in Figure 3. Each image is corrupted with Gaussian 50 dBs noise.

In this respect, we made a direct comparison with the NRED's results in order to expose the performance of the proposed method on both noiseless and noisy images. The proposed method only uses a nonmaximum suppression procedure based on an optimal threshold selection method.

Figure 3 presents the ten noise free images, while Figure 4 presents corrupted images with Gaussian 50 dB noise. In the given figures, the second and fifth columns show the NRED's outputs, whereas the third and sixth columns refer to the output of our proposed method. From the output edge maps in Figure 4, we can see that the LPGPCA-ED algorithm is clearly superior to the NRED method on the images with Gaussian 50 dBs noise. For a visual evaluation, it can be seen that the structure of the images are hidden in the edge maps of the NRED's edge detector, in particular for the noisy version the 'flower' image. On the other hand, the LPGPCA-ED is able to extract edge maps that are more apparent and resistance to noise. At first glance, although the NRED can produce clearer lines in noise-free images, the same performance does not proceed for images corrupted with strong noise. In fact, for almost all edge detection methods, a low-pass filter is applied on the processed image aiming to attenuate the high frequencies of related noisy information. However, it is widely known that using a low-pass filter in order to reduce noise may result in a degradation of grayscale pixel values, thereby destroying the edges. Therefore, a crowded set of filters has emerged in the literature, preserving the edges and smoothing the image at the same time. A visual comparison of certain filters can be found in the website of paper [27]. However, edge extraction is not a fundamental goal of these filters, and they are used as a preprocessing step in computer vision and image processing applications. Although the noise removing procedure is carried out as an individual application, it is sometimes the best choice for utilization in the preprocessing step. In this regard, the procedures for edge detection from a noisy image consist of two stages, which are smoothing the image followed by edge map extraction. In this study, we have combined these stages as a single procedure by obtaining a noise-free gradient map.

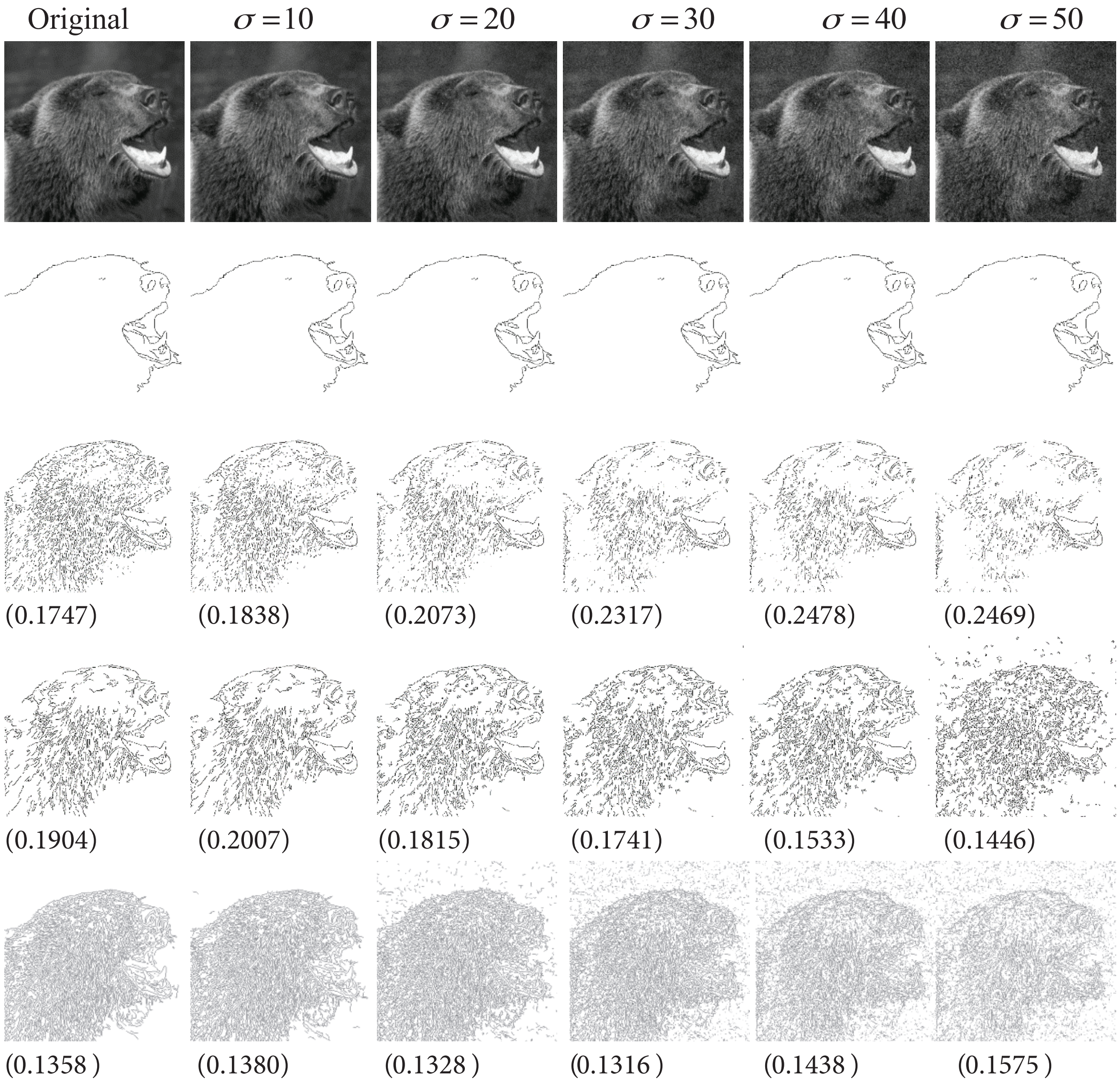

Figure 5. The obtained F-measure values on Bear image.

3.2. Quantitative evaluation using F-measure

To evaluate the performance of the proposed method objectively, we have used the F-measure (also known as F-score and F-metric) by concentrating on test images of the ROC (http://figment.csee.usf.edu/edge/roc) and RUG (http://www.cs.rug.nl/˜imaging/APD/rug/ rug.html) databases. Since the F-metric is only computed with edge pixels, available source codes of the relevant edge detectors are greatly needed to obtain precision and recall values using pixels. Due to a lack of codes proposed for edge extraction on the given datasets, we made comparison with traditional methods and the NRED method for their simple implantation and their fast run times. Generally, the F-measure is used to crosscheck accuracy by considering precision p and recall r to compute a score. In the context of the F-measure, while 1 indicates a high score, 0 refers to the worst one. Moreover, the F-measure has been employed in several distinct problems, such as boundary detection at a single scale [11] and multiscale [9], performance evaluation of the segmentation method [28] and [29], information retrieval [30], classification [31], and clustering [29]. The traditional F-measure is the harmonic mean of precision and recall, and computed with the following formulas:

$$\text{precison } = \frac{TP}{TP+FP} \tag{13}$$

$$\text{recall } = \frac{TN}{TN+FN} \tag{14}$$

$$\text{F-measure } = 2x\frac{\text{precision } \times \text{ recall}}{\text{precission } + \text{ recall}} \tag{15}$$

As demonstrated in Table 1, remarkable results are obtained using the images of the ROC and RUG databases. The ROC database consists of 60 real images, 50 of general objects and 10 of aerial scenes, and their manually specified ground truth segmentation data, whereas the RUG database includes 40 real images and corresponding manually specified ground truth images. Thus, 100 images were used to evaluate the performance of the proposed method in terms of the F-measure.

Table 1. The obtained average, min and max values of F-measure from experimented 100 images (downloaded from RUG and ROC databases) under different noise with the varying variance range.

METHOD	F-SCORE	ORIGINAL	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$
LPGPCA-ED	Average	**0.5978**	**0.5925**	**0.5776**	**0.5611**	**0.5467**	**0.5348**
	Min	0.1747	0.1838	**0.2073**	**0.2113**	**0.2155**	**0.2216**
	Max	0.8075	0.7934	0.7745	0.7607	0.7512	0.7330
SOBEL	Average	0.5776	0.5668	0.5066	0.4337	0.3706	0.3205
	Min	0.1544	0.1620	0.1642	0.1520	0.1355	0.1167
	Max	0.8474	0.8247	0.7719	0.7079	0.6254	0.5492
PREWITT	Average	0.5794	0.5697	0.5144	0.4431	0.3795	0.3287
	Min	0.1585	0.1670	0.1653	0.1529	0.1388	0.1125
	Max	0.8489	0.8276	0.7824	0.7111	0.6362	0.5653
CANNY	Average	0.5961	0.5914	0.5736	0.5377	0.4908	0.4470
	Min	0.1904	0.2007	0.1815	0.1741	0.1414	0.1224
	Max	0.8281	0.8216	0.8171	0.8083	0.7922	0.7628
NRED	Average	0.4227	0.3812	0.3628	0.3735	0.3936	0.4111
	Min	0.1251	0.1163	0.1055	0.1251	0.1364	0.1522
	Max	0.8024	0.7821	0.7452	0.6870	0.6522	0.6433

The ROC database [32] was developed to objectively evaluate edge detectors by accurately rating the pixel on the edge map with respect to a specific task. ROC uses real images and considers three values to generate a ground truth, namely, edge, nonedge, or do not count. In other words, the segmented regions and edge pixels included in ground truth (GT) have distinct meanings, i.e. the black pixels represent the edges, the gray represents no edges, and the white represents 'don't care'. The areas classified by neither edge detector nor by *GT*, are called 'don't care' regions. If a detector marks an edge pixel based on a predefined tolerance or a pixel is marked as edge by either of the two, it is counted as a true positive (TP). If an edge pixel is decided as an 'edge' in a 'nonedge' region, it is considered as a false positive (FP). The 'don't care' regions, which are marked white, are not taken into account. In addition, the points that are nonedge in GT, and are also marked as nonedge in the obtained edge map, are called true negatives (TN). If a pixel is considered as 'nonedge' by GT, but are reported as 'edge' in the output of the detector, then it is accepted as a false negative (FN). Hence, if a point falls in one of the classes (TP, FP, TN and FN), then the count of the observed class is increased by 1. In the case of the TP and FN classes, the percentage rate is derived in terms of the total of edge pixels in GT. For the FP and TN, the percentage rate is computed in terms of the corresponding total number of pixels in the image.

To investigate the performance of the proposed method under a noisy environment, the experimental images were corrupted by adding independent identically distributed Gaussian noise. As can be seen from Table 1, the variances of the imposed Gaussian noise are 10, 20, 30, 40, and 50, for the columns from 4 to 8, respectively. In addition, images in the third column contain no noise. From the 100 noisy images, the performance of four algorithms is compared in respect to the obtained average, min, and max F-measure values.

Undoubtedly, it is clear that the proposed method is superior to the Sobel, Prewitt, Canny, and NRED methods, when the average F-measure (denoted as F-score in Table 1) values are taken into account. In Table 1, min indicates the lowest obtained value of the F-measure, whereas max shows the highest obtained value. At first glance, although in the case of the max values the Canny algorithm is able to extract attractive results and presents amazing competition, the proposed method outperforms the Canny for min and average values in most cases. Furthermore, it can be clearly seen that the performances of the Sobel, Prewitt, Canny, and NRED methods decrease rapidly, whereas the performance of the proposed falls off slowly when the density of noise is increased. This is because our edge detector is more resistant to noise, since the derivative procedure is not carried out. Although the Canny method performs significantly better than the Sobel, Prewitt, and NRED methods, the detected edges are in jagged form, and also many noise spots are produced, even with nonmaximum suppression and dual thresholding, on the postprocessing. Therefore, it shows that the Canny method is still extremely sensitive to Gaussian noise. As a result, the obtained results suggest that the proposed method can be effectively used either to detect edges in noiseless images or in the case of high noise levels.

Finally, our min values and Canny's related scores method, given in Table 1, are presented in Figure 5. Canny's max values and our related scores, given in Table 1, are presented in Figure 6. Again, the last rows of Figures 5 and 6 present the visual results of the NRED algorithm on the given images. In both figures, the variances of imposed Gaussian noise are 10, 20, 30, 40, and 50, for columns 2 to 6, respectively. However, images in the first column contain no noise. The second row indicates ground truth images. The third and fourth rows present the edge maps obtained from the LPGPCA_ ED and Canny edge detectors, respectively. Moreover, the results of the NRED are given in the last row of Table 1 in order to make a benchmark evaluation. The obtained F-measure results are given under the related image. Obviously, it can be seen that our method is resistant to high level noise ratio, whereas other methods are unable to retain their quality in high level noise conditions.

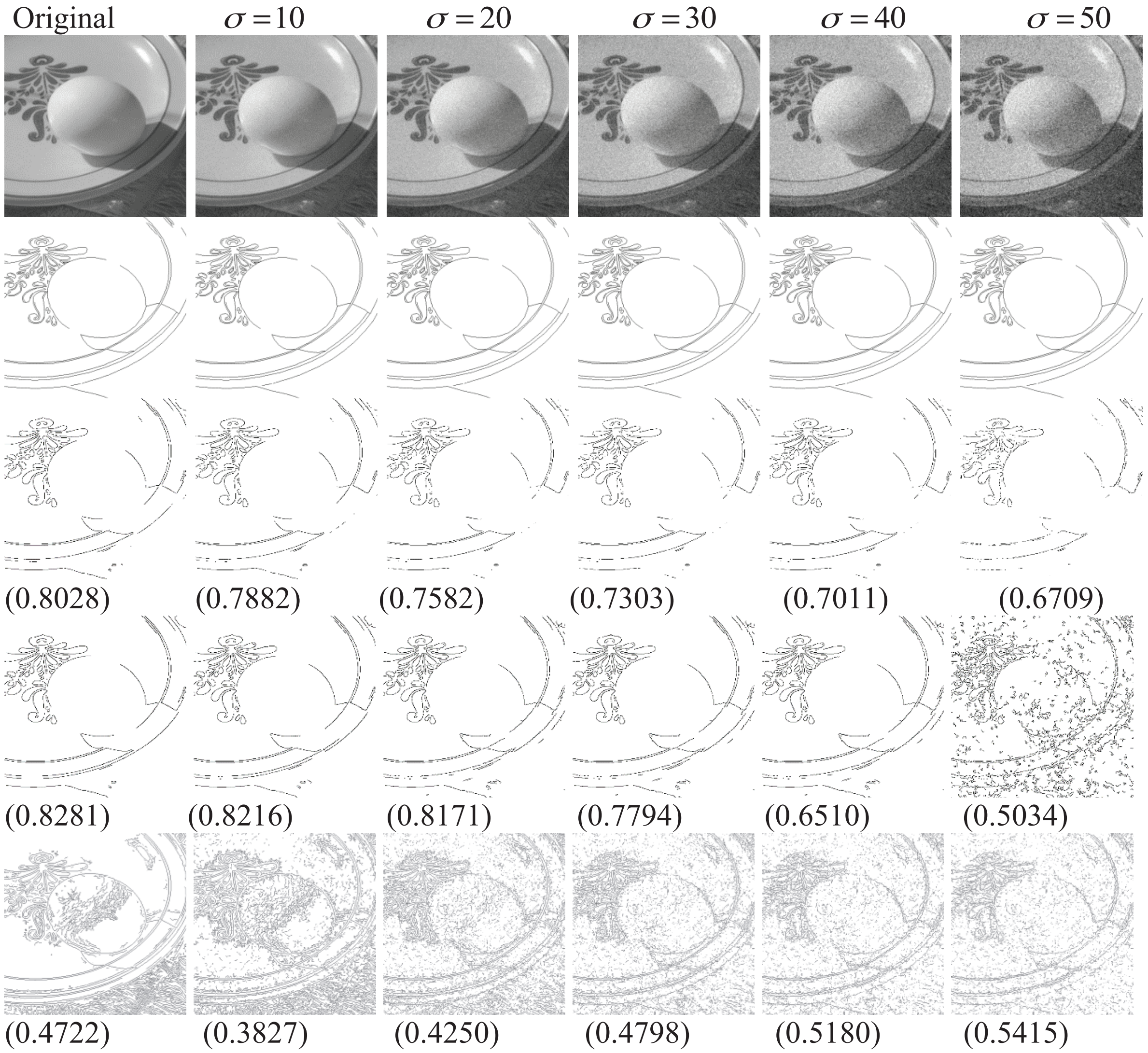

Figure 6. The obtained F-measure values on the Egg image.

For the bear image in Figure 5, LPGPCA-ED is able to produce very clear edges, but a large portion of the edges were missing in the images with less noise level. When the pattern of the face of the bear in Figure 5(a) is considered, the edges detected by Canny are jagged in the case of high level noise (especially around the nose); however, with our LPGPCA method, the edges are preserved. Moreover, the quality of output produced by the NRED method worsens when the noise level is increased. For the egg image in Figure 6, the edge maps obtained from Canny contain certain rough edges. Therefore, it can be said

that the proposed method is robust and useful for noisy images. Again, the performance of the NRED method reaches min level in the case of high level noise.

3.3. Quantitative evaluation using PFOM metric

For the comparison stage, the results of LPGPCA-ED are compared with recently proposed noisy edge detectors, namely the PSOs, RRO, and NRED methods, and the Canny method, and are exhibited in Table 2. The PFOM values of the PSOs, RRO, and the Canny methods have been taken from referenced study [5]. PFOM is widely used for assessing the performance of edge detector methods. The formula of PFOM is

Table 2. Comparison in terms of PFOM values.

Image	*Noise*	*LPGPCA-ED*	*PSO1*	*PSO2*	*Canny*	*RRO*	*NRED*
Saturn	G22	0.852371	0.772790	0.772799	0.852345	0.812005	0.278551
Saturn	G18	0.896157	0.851451	0.853361	0.822624	0.813742	0.277149
Saturn	G14	0.908028	0.780282	0.784602	0.840852	0.824204	0.290917
Saturn	G10	0.912640	0.885272	0.883163	0.841772	0.766146	0.317660
Saturn	G6	0.921499	0.767548	0.767398	0.838813	0.828318	0.402304
Cube	G22	0.823747	0.617938	0.618242	0.187473	0.401117	0.449262
Cube	G18	0.847549	0.644613	0.646577	0.222038	0.360251	0.488735
Cube	G14	0.848558	0.517665	0.516603	0.207251	0.406565	0.547031
Cube	G10	0.851307	0.632640	0.633265	0.194032	0.399380	0.617938
Cube	G6	0.846304	0.589924	0.589206	0.203340	0.395320	0.642070
Wall	G22	0.713264	0.832500	0.746579	0.654331	0.680672	0.209741
Wall	G18	0.770090	0.747364	0.746972	0.652029	0.673407	0.202100
Wall	G14	0.811573	0.791053	0.791264	0.652318	0.675357	0.208258
Wall	G10	0.783910	0.806529	0.806284	0.630157	0.669606	0.225366
Wall	G6	0.765425	0.780092	0.780462	0.635119	0.671810	0.237669
Street	G22	0.746696	0.810434	0.809075	0.652529	0.741287	0.376883
Street	G18	0.867099	0.743296	0.743951	0.663258	0.698976	0.350605
Street	G14	0.815116	0.746577	0.746826	0.591021	0.662430	0.328896
Street	G10	0.757834	0.637710	0.641211	0.581495	0.664391	0.336404
Street	G6	0.756993	0.750670	0.750181	0.638049	0.722894	0.386468
Saturn	I0.1	0.410059	0.419754	0.421777	0.374629	0.389246	0.668164
Saturn	I0.2	0.386997	0.468760	0.470071	0.114122	0.394962	0.634414
Saturn	I0.3	0.355234	0.484417	0.483590	0.008486	0.365243	0.502715
Saturn	I0.4	0.380362	0.344146	0.191153	0.003533	0.249544	0.376460
Saturn	I0.5	0.644947	0.191539	0.192462	0.001750	0.007544	0.247030
Cube	I0.1	0.600244	0.570007	0.569811	0.238533	0.451012	0.720317
Cube	I0.2	0.624673	0.534157	0.535551	0.066385	0.430736	0.706199
Cube	I0.3	0.549491	0.535441	0.534368	0.005257	0.393973	0.686087
Cube	I0.4	0.591714	0.406655	0.406561	0.002173	0.263651	0.507045
Cube	I0.5	0.618456	0.291388	0.291420	0.001052	0.009412	0.378604
Wall	I0.1	0.617216	0.474320	0.477228	0.612840	0.394115	0.291174
Wall	I0.2	0.677795	0.485948	0.488712	0.116294	0.388994	0.285974
Wall	I0.3	0.598300	0.581962	0.582185	0.005233	0.369926	0.288990
Wall	I0.4	0.504426	0.438475	0.440016	0.003154	0.241493	0.249565
Wall	I0.5	0.497344	0.254742	0.256399	0.002110	0.008363	0.194187
Street	I0.1	0.560255	0.542130	0.542094	0.492784	0.384163	0.474044
Street	I0.2	0.565659	0.503819	0.381383	0.147636	0.381919	0.498648
Street	I0.3	0.573907	0.456459	0.456501	0.005876	0.364330	0.530457
Street	I0.4	0.568247	0.413806	0.413252	0.004350	0.244949	0.388670
Street	I0.5	0.600755	0.274330	0.275494	0.002438	0.008015	0.275413

$$PFOM = \frac{1}{\max(I_E, I_{GT})} \sum_{i=1}^{I_{GT}} \frac{1}{1 + \kappa e(i)^2}, \tag{16}$$

where I_E and I_{GT} refer to the total number of detected edge points from the proposed method and that already given in related ground truth image, respectively, $e(i)$ indicates the value of error in terms of edge point localization (i.e. the distance between an edge pixel and the nearest edge pixel of ground truth), and κ denotes the scale constant, typically chosen as 1/9. As is emphasized in [12], the larger value of PFOM indicates better performance and the ideal value is 1. Table 2 shows the obtained PFOM values from the test images after NRED, PSO1, PSO2, and RRO. The experimental images and their ground truths were taken from the website given in [33] and have been tested in study [34]. To obtain experimental results, different levels of Gaussian and impulsive noise have been imposed on four different images, named saturn, cube, wall, and street. In Table 2, G6, G10, G14, G18, and G22 refer to peak signal noise ratios (PSNR) from 6 dB to 22 dB for

Gaussian noise and I0.1, I0.2, I0.3, I0.4, and I0.5 refer to noise probability from 0.1 to 0.5 in the case of impulsive noise. By considering the quantitative results from Table 2, it can be inferred that the performance of the LPGPCA-ED method is substantially better than that of the others. Moreover, the performance of all the detectors in the case of Gaussian noise is much better than for impulsive noise when considering the numerical results. Interestingly, specifically for the Gaussian noise, the performance of PSO1, PSO2, RRO, and NRED decreases in parallel to the noise ratio, whereas the performance of LPGPCA-ED rises when the noise level increases. This is because the proposed method is not based on a derivative procedure that makes the displacement on the location of pixels and is sensitive to noise. It can be clearly seen that for Gaussian noise the performance of LPGPCA-ED is readily observable. Although the NRED method presents better results than the other methods in the case of low level impulsive noise, for high level noise, the results of the proposed method are better than those of NRED.

The obtained superior results could be attributed to the employment the LPGPCA procedure for edge strength detection for which noisy pixels are recovered by cleaning noisy coefficients in the PCA domain. In contrast to our method, the others proposed to extract edges from noisy images do not make an effort as realized in our method and directly attempt to reveal edge pixels with two consecutive stages, namely edge map extraction and thresholding. However, since imposing noise onto an image changes the regulation among pixels, it is essential to restore the pixels that are exposed to the deformation and corruption by external means. Therefore, we applied the LMMSE procedure to regulate noise level and extract good quality edge maps. Additionally, the NMS procedure is based on the histogram of block pixels as described in the thresholding stage. In this respect, the PFOM values may vary with respect to different noise levels.

Additionally, the running times of the LPGPCA-ED, PSOs, and NRED methods are compared for the sake of speed evaluation. According to the study of the PSO edge detector [5], the execution time of PSO1 was usually between 50 and 70 s, while that of PSO2 was between 40 and 50 s. In addition, the proposed method shows similar periods in average execution time for images corrupted with Gaussian and impulsive noise at about 60 and 50 s, respectively. However, the elapsed running time for the NRED method is between 10 and 15 s. Although the running time of the proposed method is longer than that of the NRED method, new ways for reducing computation time will be investigated, implementing the algorithm on new platforms with different programming languages, i.e. OPENCV and C ++, or by optimizing the code. Generally, most of the elapsed time is wasted whilst using the block matching technique for grouping similar pixels. With block matching, different homogeneous regions are constructed in order to suppress noise without ruining edges. As a result, when considering the execution time, if the size of the blocks is reduced then the running time will similarly decrease.

4. Conclusion

In this paper, we propose a new edge detection model based on the concept behind LPGPCA based noise suppression. The experimental results indicate that the proposed edge detection method works efficiently on images influenced by noise and presents the best performance in terms of similarity to the desired edges. Consequently, after a comparative evaluation with traditional and recently proposed edge detectors, the advantages of the proposed method are that LPGPCA-ED is robust against noise, extracts clear continuous edges, preserves the structure of objects situated in the processed image, and gives remarkable results in terms of F-measure and PFOM metrics.

References

[1] Kim KB, Kim JS, Lee S, Choi JS. Fast image registration using pyramid edge images. International Journal of Intelligent Engineering and Systems 2009; 2: 1-8.

[2] Lakshmi A, Rakshit S. Image restoration: an edge detection based regularization. Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG); 15–17 December 2011; Hubli, Karnataka, India: IEEE. pp. 114-117.

[3] Ma X, Grimson WEL. Edge-based rich representation for vehicle classification. Computer Vision (ICCV); 17–21 October 2005; Los Alamitos, CA, USA: IEEE. pp. 1185-1192.

[4] Topal C, Akinlar C. Edge drawing: a combined real-time edge and segment detector. Journal of Visual Communication and Image Representation 2012; 23: 862-872.

[5] Setayesh M, Zhang M, Johnston M. A novel particle swarm optimisation approach to detecting continuous, thin and smooth edges in noisy images. Information Sciences 2013; 246: 28-51.

[6] Canny J. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1986, 8: 679-698.

[7] Hou Z, Koh T. Robust edge detection. Pattern Recognition 2003, 36: 2083-2091.

[8] Fesharaki M, Hellestrand GR. A new edge detection algorithm based on a statistical approach. Speech, Image Processing and Neural Networks; 13–16 April 1994; Hong Kong, China: IEEE. pp. 21-24.

[9] Ren X. Multi-scale improves boundary detection in natural images, 10th European Conference on Computer Vision; 12–18 October 2008; Marseille, France: Springer. pp. 533-545.

[10] Bovik AC, Huang TS, Munson DC Jr. Nonparametric tests for edge detection in noise. Pattern Recognition 1986; 19: 209-219.

[11] Kokkinos I. Boundary detection using F-measure, filter-and feature-(F3) boost. 11th European Conference on Computer Vision; 5–11 September 2010; Heraklion, Crete, Greece. Springer: pp. 650-663.

[12] Lim HD. Robust rank-order test for edge detection in noisy images. Nonparametric Statistics 2006; 18: 333-342.

[13] Padmavasavi K, Kumar NU, Rao EK, Madhavilatha M. Performance evaluation of adaptive statistical thresholding based edge detection using GLCM in wavelet domain under noisy conditions. International Journal on Graphics, Vision and Image Processing 2010; 10: 35-44.

[14] Rakesh RR, Chaudhuri P, Murthy C. Thresholding in edge detection: a statistical approach. IEEE Transactions on Image Processing 2004; 13: 927-936.

[15] Srivastava GK, Verma R, Mahrishi R, Rajesh S. A novel wavelet edge detection algorithm for noisy images. International Conference on Ultra Modern Telecommunications & Workshops; 12–14 October 2009; St. Petersburg, Russia: IEEE. pp. 1-8.

[16] Lee J, Haralick R, Shapiro L. Morphologic edge detection. IEEE Journal of Robotics and Automation 1987; 3: 142-156.

[17] Micheli DE, Caprile B, Ottonello P, Torre V. Localization and noise in edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1989; 11: 1106-1117.

[18] Aizenberg IN, Aizenberg NN, Vandewalle J. Precise edge detection: representation by Boolean functions, implementation on the CNN. Fifth IEEE International Workshop on Cellular Neural Networks and Their Applications Proceedings; 14–17 April 1998; London, England: IEEE. pp. 301-306.

[19] Gudmundsson M, El-Kwae EA, Kabuka MR. Edge detection in medical images using a genetic algorithm. IEEE Transactions on Medical Imaging 1998, 17: 469-474.

[20] Shui PL, Zhang WC. Noise-robust edge detector combining isotropic and anisotropic Gaussian kernels. Pattern Recognition 2012; 45: 806-820.

[21] Zhang L, Dong W, Zhang D, Shi G. Two-stage image denoising by principal component analysis with local pixel grouping. Pattern Recognition 2010; 43: 1531-1549.

[22] Priya GH, Venkatesh B, Srivani A, Sowmya G, Rajasekhar N. Noise removal in image using LPG-PCA (Local Pixel Grouping-Principle Component Analysis) algorithm. Pattern Recognition 2010; 43: 1531-1549.

[23] Ray R. Unsupervised edge detection and noise detection from a single image. Pattern Recognition 2013; 46: 2067-2077.

[24] Lopez CM, Baets BD, Bustince H. Quantitative error measures for edge detection. Pattern Recognition 2013; 46: 1125-1139.

[25] Pratt WK. Digital Image Processing. New York, NY, USA: Wiley, 1978.

[26] Von Gioi RG, Jakubowicz J, Morel JM, Randall G. LSD: a line segment detector. Image Processing On Line 2012; 2: 35-55.

[27] Papari G, Petkov N, Campisi P. Artistic edge and corner enhancing smoothing. IEEE Transactions on Image Processing 2007; 16: 2449-2462.

[28] Rafiee G, Dlay S, Woo W. Unsupervised segmentation of focused regions in images with low depth of field. IEEE International Conference on Multimedia and Expo (ICME), 15–19 July 2013; San Jose, CA, USA: IEEE. pp. 1-6.

[29] Derraz F, Taleb AA, Beladgham M, Khelif M. Evaluation of segmentation quality for the geometrical active contour method. Image and Signal Processing and Analysis, 16–17 September 2005; Zagreb, Croatia: IEEE. pp. 377-382.

[30] Dembczynski K, Jachnik A, Kotlowski W, Waegeman W, Huellermeier E. Optimizing the F-Measure in multi-label classification: plug-in rule approach versus structured loss minimization, Proceedings of the 30th International Conference on Machine Learning; 16–21 June 2013; Atlanta, GA, USA: IEEE. pp. 1130-1138.

[31] Kandefer M, Shapiro S. An F-measure for context-based information retrieval, Commonsense 2009; pp. 79-84.

[32] Bowyer K, Kranenburg C, Dougherty S. Edge detector evaluation using empirical ROC curves. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 23–25 June 1999; Fort Collins, Colorado, USA: IEEE.

[33] Fernández García NL, Carmona Poyato A, Medina Carnicer R, Madrid Cuevas FJ. Automatic generation of consensus ground truth for the comparison of edge detection techniques. http://www.uco.es/˜ma1fegan/Comunes/investigacion/imagenes/ground-truth.html.

[34]] Fernández García NL, Carmona Poyato A, Medina Carnicer R, Madrid Cuevas FJ. Automatic generation of consensus ground truth for the comparison of edge detection techniques. Image and Vision Computing 2008; 2: 496-511.