Intelligent Edge Detector Based on Multiple Edge Maps

Mohammed Qasim, Wei Lee Woon, Zeyar Aung, and Vinod Khadkikar
Masdar Institute of Science and Technology
Abu Dhabi, United Arab Emirates
{mmqasim, wwoon, zaung, vkhadkikar}@masdar.ac.ae

Abstract— **An intelligent edge detection method is proposed. The method is based on the use of pattern recognition and machine learning techniques to combine the outputs of multiple edge detection algorithms. In this way, the limitations of the individual edge detectors can be overcome and performance enhancement is achieved. Two widely used classification algorithms, the Naive Bayes Classifier and the Multi-layer Perceptron, were selected for the learning task. The proposed system was evaluated on artificial and real images. A simple class labeling system based on the output of all edge detectors is suggested to provide controllability between detection sensitivity and noise resistance. Principal Component Analysis was performed to reduce computational burden and improve detection accuracy. The method is shown to achieve a practical compromise between detection sensitivity, computational complexity, and noise immunity.**

Keywords—*edge detection; machine learning, naïve Bayes classifier, multi-layer perceptron, principle component analysis*

I. Introduction

Edge detection is one of the most fundamental operations in image processing and computer vision. It is defined as the process of locating the boundaries of objects or textures depicted in an image (i.e. *edges*) [1]. Knowing the positions of these boundaries is critical in the process of image enhancement, recognition, restoration and compression.

Since edges are often characterized by discontinuities or abrupt changes in the image brightness or intensity, the principle of edge detection relies broadly upon detecting sudden changes in the image grey level intensity. Unfortunately, abrupt changes in image intensity also often correspond to noise, which is characterized by the presence of high frequency components. Because edge detectors are commonly based on gradient operators, the noise-related high frequency components will also be amplified, resulting in distortions to the resulting edge maps. To alleviate this problem, low pass filtering is often applied prior to the gradient operators. However, this in turn might result in loss of information regarding detailed features in the images. As such, designing a generalized edge detector that can perform well in all contexts is very challenging [2].

There are many commonly used edge detectors such as the Sobel [3], Roberts [4], Canny [5] and Laplacian-of-Gaussian [6] (LoG) edge detectors. Most of them depend on first- or second order gradient operators but they differ in that different differentiation operators or smoothing stages are used. For example, the Sobel operator estimates a smoothed gradient of the image intensity by convolving the image with an integer valued filter in horizontal and vertical direction. In mathematical terms two 3×3 kernels Gx and Gy in the horizontal and vertical directions are convolved with the image to estimate the image derivatives. If the test image is denoted by V, then the intensity computation can be written as follows:

$$\mathrm{Gx} = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \times \mathrm{V} \quad (1)$$

$$\mathrm{Gy} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \times \mathrm{V} \quad (2)$$

The Roberts operator is similar to the Sobel detector; however, no smoothing stage is applied prior to the computation of derivatives and search for edges is in the diagonal rather than the vertical and horizontal directions.

On the other hand, the Canny edge detector, which is considered one of the best edge detectors with respect to noise tolerance, applies Gaussian kernel filtering to smooth the image.

Finally, the Laplacian of Gaussian (LoG) is a second order edge detector. The noise effects which have been exaggerated by the first order derivative will be further compounded when using second order edge detectors. Therefore, the LoG detector utilizes a prior smoothing stage where the image is convolved with a Gaussian kernel smoothing filter. The edges correspond to the zero crossings of the LoG of the image. Further details of the above mentioned edge detectors are provided in [3]-[6]. Other edge detection algorithms not described here include the Prewitt Operator [7] (which is similar to Roberts [4]) and the Zero Crossing Operator [8].

The new trend is to enhance the noise-resistance capability of traditional edge detectors. With the evolution of machine learning and pattern recognition algorithms, it has become feasible to achieve intelligent edge detection with improved noise tolerance. For example, a novel genetic-neuro-fuzzy system was proposed for edge detection of ultrasound images in [9]. In this work, the author utilized competitive neural networks (NN) as the edge classifier. An on-line genetic algorithm (OGA) was used to optimize and regulate the system parameters. The method resulted in improved performance in terms of noise resistance compared to conventional edge detectors.

In [10], a novel compound edge detector has been proposed to improve the noise-resistance capability. The method is based on the fusion of multiple edge detectors and utilizing their outputs in creating an intelligent noise-resistant edge detector. The merits of this approach rely upon combining the advantages of different edge detectors to produce a more intelligent and noise-resistant edge detector. K-Nearest Neighbors (KNN), Naive Bayes Classifier (NBC) and Artificial Neural networks (ANN) have been utilized in the edge classification and compared with the Canny edge detector under different noise levels.

In [11], the authors proposed an edge detection method that uses K-means clustering and different properties of image pixels as features. The qualities of the different clustering results obtained using different number of clusters were analyzed using silhouette indexes in order to choose the best number of clusters. The results exhibited higher noise resistance compared to Canny and Sobel methods.

It was concluded that there is no general edge detector that can perform perfectly in all contexts [1]. Therefore, an edge detector which combines multiple edge detectors has the potential to perform better in a variety of contexts. In this paper, a classification-based edge detector based on the merging multiple edge detectors is proposed.

The main objective of the proposed method is to fuse multiple edge detectors and use them to formulate a classification-based detection method which intelligently detect edges and improve noise immunity. The fused output of six edge detectors, namely, Sobel [3], Roberts [4], Canny [5], LoG [6], Perwitt [7], and zero-crossing [8] edge detectors, has been used for that purpose. The output of each edge detector was used as a feature in the feature extraction process. Two classifiers were adopted to perform the edge classification, namely ANN and NBC. Due to the sparseness of the image data, and in order to reduce the computational burden that results from such large images, Principle Component Analysis (PCA) was adopted to reduce the dimensionality of the data, and to add further intelligence to the proposed method. The proposed method was tested under different levels of noise added to the image and compared with the Canny edge detector.

The results have proved to be superior in terms of noise resistance and sensitivity to Canny edge detector which is considered to be the most immune to noise among other conventional edge detectors. The proposed method was implemented using the edge detection algorithms and statistics tool box in the Matlab [12] environment.

II. THEORETICAL BACKGROUND

A. Naïve Bayes Classifier (NBC)

Naïve Bayes Classifier is a probabilistic classifier used to estimate the posterior distribution over the target classes. To simplify this process, the NBC makes the assumption of feature independence. Therefore the joint likelihood can be written as:

$$P(X_1, X_2, \dots, X_d / C_m) = P(X_1/C_m).P(X_2/C_m) \dots P(X_d/C_m) \quad (3)$$

The posterior distribution can then be written as:

$$P(C_m/X_1, X_2, \dots, X_d) \propto P(X_1/C_m) \dots P(X_d/C_m).P(C_m) \quad (4)$$

where $P(X_i/C)$ terms are the individual independent likelihoods, d is the number of features and P (C_m) is the class prior. In our edge detection problem we have six features and two classes, specifically edge pixel and non-edge pixel classes.

B. Multi Layer Perceptron

Multi Layer Perceptron (MLP) which is also known for the Artificial Neural Networks is a classification technique inspired by the neurons in the human nervous system. Unlike single perceptron network, Multi-layer Neural Networks can learn non-linearly separable data.

The first layer of the ANN is called the input layer and is where the input samples are presented. Each neuron at the input layer represents one feature. Therefore, the size of the input layer is equivalent to d (dimension of the data). The second layer is called the hidden layer. This additional layer adds the non-linearity to the conventional single layer perceptron algorithm by a nonlinear activation function.

Choosing the size of the hidden layer is not trivial. In the proposed edge detection system, the size of the hidden layer was increased until no further improvement was observed. Finally, the output layer consists of the output neurons where each neuron represents a class.

In this paper, an MLP with six neurons at the input layer, 10 neurons at the hidden layer (best performance) and one neuron at the output layer was used. The output of the ANN is either 1 or 0, indicating edge or non-edge pixels. The ANN toolbox in Matlab [12] was utilized to perform the experiments.

C. Principal Component Analysis (PCA)

PCA is a feature combination or dimensionality reduction technique that assumes linear mapping. The idea behind PCA is to map the data linearly (linear transformation) into a new set of features that preserve the largest variances of the data. The intuition is to find the direction of the maximum variance which corresponds to the direction of the Eigen vector with the largest Eigen value. The proof of this intuition is beyond the scope of this paper and can found in [13].

The basic PCA performs the following steps to project the data into the new vector space:

1. Shift each sample point by the mean of all sample points corresponding to one feature:

$$z_{im} = x_{im} - \mu_d \quad (5)$$

where x_{im} is the i^{th} observation corresponding to the feature m.

2. Find the Eigen vectors and Eigen values of $(n-1) \times$ covariance matrix:

$$\Sigma = 1/(n-1)\,(z_{im} z_{im}^{T}) \quad (6)$$

where Σ is the covariance matrix and n is the number of sample points (number of pixels).

3. The new vector space is then:

$$W = E^{t}z \tag{7}$$

where $E = [e_1\ e_2 \ldots e_d]$ is the vector of the Eigen vectors.

PCA was computed using the PCA command in Matlab. The algorithm returns the coefficients (principle components), the mapped data into the new vector space and the Eigen values arranged in descending order.

III. THE PROPOSED METHOD

A. *Classification Problem Formulation*

The methodology of the proposed method is depicted in Figure 1. The image is processed by six component Edge Detectors (labeled ED1-ED6 in the figure). The output of each detector is converted into a vector where each pixel of the resulting vector is an observation or a sample point in the $N \times M$ edge detection windows. The resulting vectors are combined to form the classification matrix as shown in Figure 1. The classification matrix has $N \times M$ rows and 6 features (edge operators), where N is the number of rows and M is the number of columns of the target picture's matrix.

B. *Label Assignment*

The mechanism of labeling the pixels as edge or non-edge pixels was based on the feedback from all edge detectors. Each edge detector returns logical 1 if the pixel is an edge and returns logical 0 if otherwise. In order to provide a reliable labeling system, a threshold limit was used to control the process of labeling. In other words, if the threshold limit was set to 3, it means that at least 3 or more of the edge detectors should detect the pixel as an edge for the labeling system to assign 1 to the corresponding pixel; otherwise, a label of 0 is assigned to the pixel.

This mechanism allows the user to control the detection sensitivity and noise resistance. For example, if the threshold is set to 1, only one edge detector would have to detect the corresponding pixel as an edge; this supports the detection of weak or faint edges, which may be detected by the Canny operator but not by the Roberts operator. This enables the fusion based edge detector to sweep among the six edge detectors with respect to the desired application. Henceforth, this threshold will be denoted by α.

C. *Classifiers Used*

From the abovementioned steps A and B, the classification matrix with the corresponding pixel labels is obtained. That completes the formulation of the classification problem where different pattern recognition techniques are used to learn the different edge patterns and intelligently classify edge and non-edge classes. Two classifiers with different properties and levels of complexity were employed in the edge classification problem, namely Naïve Bayes Classifier and Artificial Neural Networks. Theoretical backgrounds of both classifiers were reviewed in Section II.

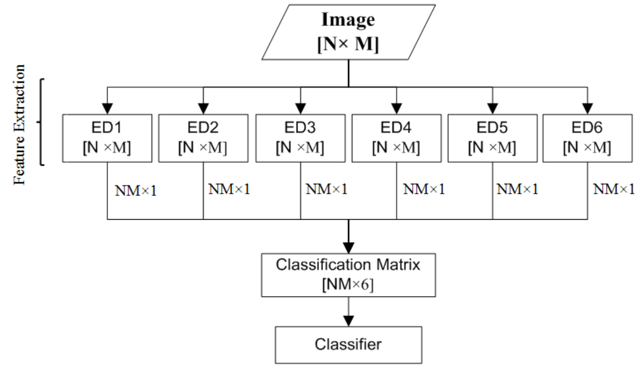

Figure 1. Schematic of the proposed method.

IV. DATA AND MODEL PARAMETERS

A. *Data*

The training images were generated by rotating an equilateral square in three steps of 22.5° clockwise. This would train the classifier to learn the different types of edges, corners and lines including horizontal, vertical and diagonal lines. Two sets of target edges were produced with threshold (α) of 3 and 4, respectively. The target edges produced using a threshold α = 3 and α = 4 are shown in Figures 2 and 3. Inner edges (squares) are observed in the target edges due to the slightly thick borders of the original square. Those inner edges can be removed by tightening the thickness of the borders; however, it is not necessary since the main objective is to train the classifier to different kinds of edges.

White Gaussian noise with 10% standard deviation was added the training images to improve noise immunity of the trained classifier. Testing data were generated by contaminating a 30° rotated square with white Gaussian noise of 5% to 40% standard deviation in increments of 5%.

The results presented in the later sections were produced using α = 3 in some parts and α = 4 in other parts. This was done to present the effect of α on the proposed edge detector.

B. *Model Parameters*

A detailed study of model and parameter selection is beyond the scope of this paper. However, tests on the best parameter values including the value of α, number of hidden units in the MLP and the optimum number of dimensions after the application of PCA is described.

1) *Threshold limit α:*

Different values of α were tested ranging from 1 to 6. It is found that reducing α would increase the sensitivity of edge detection at the expense of noise tolerance. In our experiments, we observed that setting α = 3 or α = 4seemed to result in an acceptable compromise between sensitivity and noise resistance.

2) *Number of hidden units:*

Hidden units are defined by activation functions or neurons in the hidden layer. Number of hidden units was varied until the best performance of the network was obtained. Ten hidden units (Nh = 10) were adopted throughout the experiments as there was no significant performance change beyond 10 units.

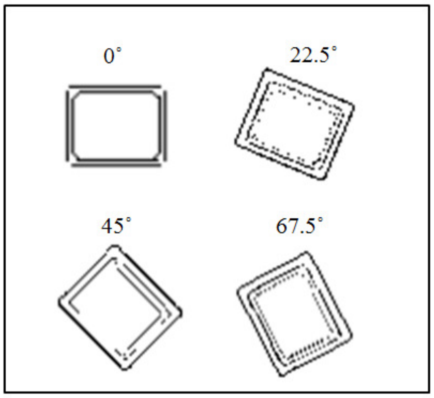

Figure 2. Edge targets for the training data depicted with threshold = 3.

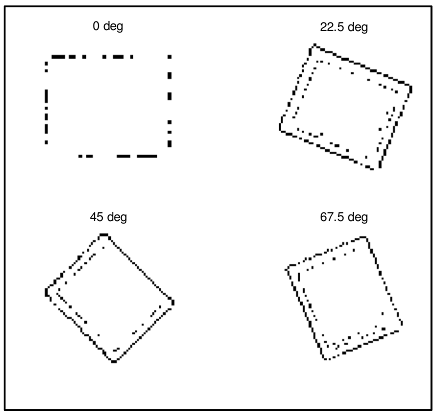

Figure. 3. Edge targets for the training data depicted with threshold = 4.

3) Dimension:

The result of PCA is a new set of features returned in descending order according to their importance (largest variance). It was found that the first two features which correspond to the maximum variances are the most discriminative ones and yield almost perfect classification. Therefore, the two features corresponding to the maximum variance were adopted throughout the paper.

V. Results and Discussions

A. Examination of Classifiers Immunity to Noise

To study the effect of noise on the NB classifier and ANN, their classification error was computed for different experiments of distorted test images with multiple levels of white Gaussian noise. The classification error was defined as the ratio of total misclassified pixels to the total number of pixels. Misclassified pixels could be edge pixels classified as non-edge pixels or vice versa. For the sake of comparison, the noise tolerance capabilities of both classifiers were compared with the canny edge detector as shown in Figure 4.

As mentioned earlier a 30˚ rotated test image was distorted with white noise of 5 to 40% standard deviation. It is worth mentioning that the depicted figure was obtained at α = 4 and a different value of α could give yield to different results. As mentioned earlier, the selection of α was based on a trade-off between sensitivity and noise immunity.

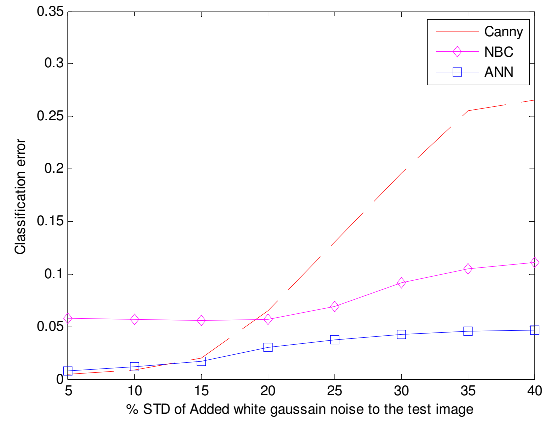

Figure 4. Edge targets for the training data depicted with threshold = 4.

Figure 5. Real image employed in testing.

It can be noticed from Figure 4 that although canny edge detector performs the best under low noise levels, it has the highest classification error under high noise levels. On the other hand, ANN has the highest overall accuracy where it performs almost equally to the Canny at low noise levels and has the lowest classification error at high noise levels. Therefore, it is considered to be the most appropriate classifier in terms of sensitivity and noise immunity. Finally, the NBC, although it is considered to be computationally simpler than the ANN, performs reasonably satisfactory under high noise levels; however, it has the lowest accuracy (detection sensitivity) at lower noise levels.

B. Tests on Real Images

To verify the results obtained in Figure 4, the proposed fusion-based edge detectors (NBC and ANN) were tested on real life images corrupted with different levels of noise, and with the same threshold level (α = 4). Naturally, depending on the scene being portrayed, real images can differ greatly in their complexity. An intermediate-complexity real image (source: http://www.pdfguides.com) was employed in the testing as shown in Figure 5. The image's source is NBC and ANN were compared with the Canny detector under 0%, 5%, 10% and 15% noise levels added to the depicted real image. The resulting edge maps of Canny, ANN, and NBC are shown in Figures 6, 7, and 8, respectively.

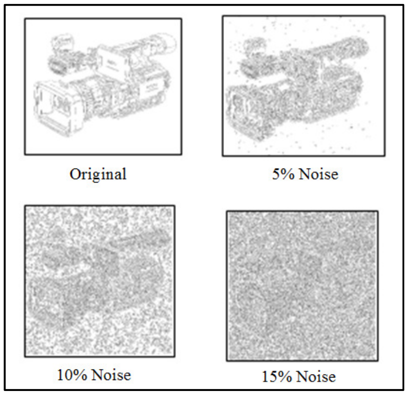

Figure 6. Canny fusion edge detector applied to the real image.

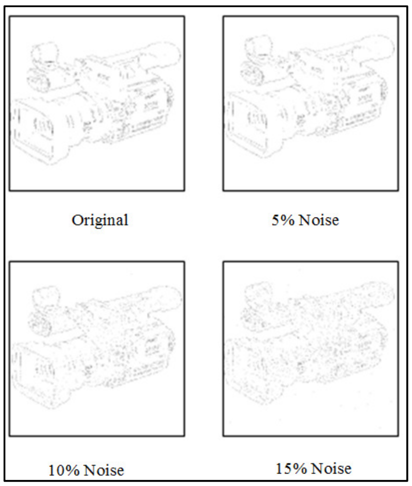

Figure 7. ANN fusion edge detector applied to the real image.

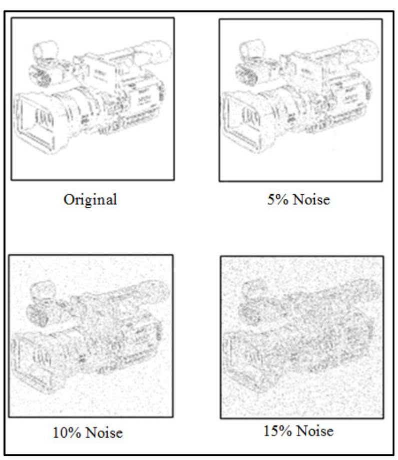

Figure 8. NBC fusion edge detector applied to the real image.

The following observations can be made from the presented results:

1. It can be noticed that the canny edge detector gives the best edges for the no noise case; however as the noise levels were increased the performance degraded fairly quickly. This proves the results presented in Figure 4.
2. Figure 7 verifies the fact that the ANN-based fusion edge detector has the highest immunity to noise. However, while the graph in Figure 4 seemed to indicate that ANN should have similar performance to the canny edge detector at low levels of noise, this did not turn out to be the case.
3. The NBC classifier shows reasonable sensitivity for the noiseless case and performs much better than the Canny at high noise levels.

It can be concluded that there is always a trade-off between sensitivity and noise immunity. NBC shows a fair compromise between these two criteria. Nonetheless, ANN performs much better than both NBC and Canny at high noise levels. Therefore, for highly distorted images, Canny and NBC would most likely fail to detect the edges whereas ANN could at least detect the sharp edges. It is also worth mentioning that the results for the real image were produced based on $\alpha = 4$. Different value of α would affect one detection parameter, either sensitivity or noise tolerance. For example ANN would be more sensitive if α was chosen to be 3 or 2. However, it would inversely affect its noise immunity.

C. Dimensionality Reduction using PCA

In the proposed method, each edge detector is considered as a feature and since we have six edge operators, the extracted data is six-dimensional with each pixel as a sample point (observation). Although six features could be considered as a reasonable dimensionality size, in case of large images dimensionality reduction could be vital to reduce the computation burden, and yet preserve the desired classification accuracy or even improve it. In addition to the size of the images, most images suffer from matrix sparseness.

In this paper, PCA was adopted to reduce the dimensionality of the classification problem. As mentioned earlier in Section II, PCA preserves largest variances in the data. As a proof-of-concept, the test image was processed via PCA and tested only on the NBC. The artificial testing image corrupted with 5% was applied to the PCA algorithm. The PCA returns the data transformed into a new vector space with new six features.

Only the first two features which correspond to the direction of the largest variances were used to train the NBC. The Receiver Operator Curve was developed for two cases, namely NBC without PCA and NBC with PCA as shown in Figure 9. The classification error and Area Under the Curve (AUC) are illustrated in Table I.

As seen in Figure 9, the ROC curve for the NBC with the PCA processed data is superior in performance than that of the NBC case with the original data. This is promising in the sense that a lot of applications tolerate some reduction on accuracy as long as computation burden is improved. In our results this accuracy tolerance is not even needed to be considered since the new 2-dimensional classification case performs better than the 6-dimensional original data. This proves the fact that some features are redundant and impotent with respect to classification improvement.

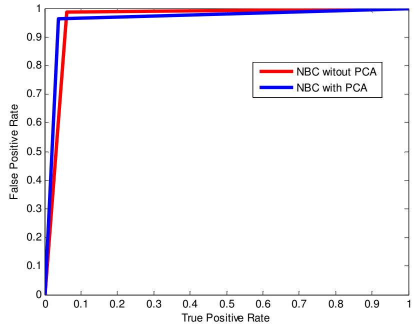

Figure 9. ROC curve of NBC without PCA (red) and ROC curve of NBC with PCA (blue) with 5% noise added to the test image.

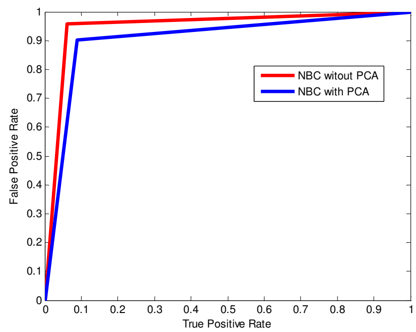

Figure 10. ROC curve of NBC without PCA (red) and ROC curve of NBC with PCA (blue) with 20% noise added to the test image.

TABLE I. COMPARISON OF AUC AND CLASSIFICATION ERROR VALUES OF NBC WITHOUT PCA AND NBC WITH PCA.

	AUC	Classification Error (%)
Without PCA	0.9620	5.9822
With PCA	0.9632	3.7964

It should be noted that increasing the noise level can potentially affect the resulting eigenspace, which in turn might necessitate the use of more PCA dimensions. Another analysis is that the direction of the largest variance may not be useful for the classification problem since noise is added and the whole classification problem might change. These comments are subjective and further investigation is needed to verify this.

Figure 10 shows the effect on the PCA after increasing the Gaussian noise level to 20% standard deviation using the same two features used previously. It can be noticed that in this case the NBC without PCA performs better than the NBC with PCA (larger AUC). As mentioned earlier, this is because the level of noise corruption has increased and different set of features will be required to improve the accuracy. Although this is the case with high noise level, the PCA still provide a reasonable accuracy and faster computation.

VI. CONCLUSIONS

In this paper, an intelligent edge detection approach is proposed using the fusion of multiple edge detectors. The proposed edge detector using ANN was proved to be the most noise resistant compared to Canny and NBC. However, NBC has been proven to detect edge with acceptable compromise between detection sensitivity and noise immunity.

PCA was applied as proof-of-concept regarding the advantage of dimensionality reduction. It was tested on artificial data and the results were promising in terms of classification accuracy and computation complexity. The justification regarding the superior performance of the PCA based NBC could be summarized due to the fact that some features are redundant and could be considered as null in the classification process.

The labeling system has added the advantage of controlling the objective of the fused intelligent edge detector as it provides more flexibility with respect to the application under experiment. That is, sensitivity and noise resistance could be controlled by adjusting the threshold value. This could be seen from the fact that changing the threshold is nothing but tuning the proposed edge detector to work as one of the six detectors. Therefore, by reaching the best compromise of threshold level, more informative data are fed to the intelligent classifiers (NBC and MLP), which in turn provide superior results in terms of sensitivity and noise immunity compared to the conventional detectors.

ACKNOWLEDGMENT

This research was sponsored by the Government of Abu Dhabi, United Arab Emirates through its funding of the Masdar Institute of Science and Technology's research project on "Fault Detection and Classification for Smart Grids with Islanded Capability".

REFERENCES

1. D. Ziou and S. Tabbone, "Edge detection techniques – an overview," *International Journal of Pattern Recognition and Image Analysis*, 8: 537–559, 1998.
2. W. Xiao and X. Hui, "An improved edge detection method for image corrupted by Gaussian noise," in *Computer and Computing Technologies in Agriculture, Volume II*, pp. 1153–1159, 2009.
3. E. Sobel, *Camera Models and Machine Perception*, PhD thesis, Electrical Engineering Department, Stanford University, California, USA, 1970.
4. L. G. Roberts, "Machine perception of three-dimensional solids," *Optical and Electro-optical Information Processing*, J. T. Tippett *et al.* (Editors), MIT Press, 1965.
5. J. Canny, "A computational approach to edge detection," *IEEE Transaction Pattern Analysis Machine Intelligence*, 8: 679–698, 1986.
6. D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London*, 207: 187–217, 1980.
7. J. M. S. Prewitt and M. L. Mendelsohn, "The analysis of cell images," *Annals of the New York Academy Science*, 128: 1035–1053, 1966.
8. J. R. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley and Sons Inc., 1997.
9. A. Rafiee, M. H. Moradi, and M. R. Farzaneh, "A novel genetic-neuro-fuzzy filter for speckle reduction from sonography images," in *Proceedings of the 6th WSEAS International Conference on Evolutionary Computing,* pp. 285–290, 2005.
10. W. L. Woon, P. Liatsis, and K. D. Wong, "Fusion of multiple edge maps for improved noise resistance," in *Proceedings of the 7th MMU International Symposium on Information and Communication Technologies*, pp. 1–8, 2006.
11. B. Neupane, Z. Aung, and W. L. Woon, "A new image edge detection method using quality-based clustering," in *Proceedings of the 10th IASTED International Conference on Visualization, Imaging, and Image Processing*, pp. 20–26, 2012.
12. Matlab v 7.11.0.584 (R2010b). Computer software. MathWorks Inc., Natick, Massachusetts, USA, 2010.
13. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd Edition, Wiley Inc., 2001.