Intelligent Edge Detector Based on Multiple Edge Maps

M. Qasim, W.L. Woon, Z. Aung

Technical Report DNA #2012-10

May 2012

Data & Network Analytics Research Group (DNA)
Computing and Information Science Program,
Masdar Institute of Science and Technology,
PO Box 54224, Abu Dhabi, UAE.

Intelligent Edge Detector Based on Multiple Edge Maps

Mohammed Qasim, W. L. Woon, Zeyar Aung

Computing and Information Science Department, Masdar Institute of Science and Technology
Abu Dhabi, UAE

***Abstract*— An intelligent edge detection method is proposed in this paper. The method is based on fusing multiple edge detectors to enhance the edge detection capability and overcome each edge operator disadvantages. The fusion method is implemented via pattern recognition and machine learning techniques providing an intelligent edge detector. Two classifiers were adopted in the learning task, Naive Bayes Classifier, and Artificial Neural Network. The two classifiers were evaluated on artificial and real images. A simple class labelling system based on the output of all edge detectors is suggested to provide controllability between detection sensitivity and noise resistance. Principle Component Analysis was performed to reduce computational burden and improve detection accuracy. The method has been proved to achieve practical compromise between detection sensitivity, computation complexity and noise immunity.**

***Index Terms*—Machine learning, Edge detection, Naïve Bayes Classifier, Artificial Neural Networks, Principle Component Analysis**

I. Introduction

Edge detection is one of the most fundamental operations in image processing and computer vision. It is considered to be essential information in the process of image enhancement, image recognition and restoration, image registration and image compression. In computer vision, edge detection is defined as the process of capturing the significant properties of objects in the image [1].

Since edges are often characterized by discontinuities or abrupt changes in the image brightness or intensity, the principle of edge detection solely relies upon capturing those step changes of the grey level image. Therefore, the image derivatives or the image intensity gradients are computed to detect the image intensity variations and capture its underlying physical boundaries. Nevertheless, abrupt changes in image intensity may correspond to different forms of noise. Therefore, intensity gradient might amplify those high frequency components (noise) and distort the original edges of the image being detected. This problem is tackled by a smoothing stage (low pass filter) applied to the image. However, smoothing might lead to information loss (image details). This gives rise to the fact that designing a general edge detector that can perform well in all contexts is not trivial [2].

There are many commonly used edge detectors such as the Sobel [3], Roberts [4], Canny [5], Laplacian-of-Gaussian [6] (LoG) edge detectors. While most of them depend on first order or second order differentiation to compute the change in intensity, their properties differ as they inhabit different differentiation operators or smoothing stages. 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 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 described as the following:

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

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

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

On the other hand, the Canny edge detector, which is considered one of the most rigorous edge detectors with respect to noise tolerance, applies Gaussian kernel filtering on the image before intensity estimation. Furthermore, the canny edge detector depends on the first derivative in intensity computation.

Finally, the Laplacian of Gaussian (LoG) is a second order edge detector. Since the noise effect is exaggerated by first derivative, these effects are doubled with second order edge detectors. Therefore, the LoG detector performs a prior smoothing stage by first convolving the image with a Gaussian kernel smoothing filter. The edges correspond to the zero crossings of the LoG of the image. Further details of the previously mentioned edge detectors are explained in [3]-[6]. The aforementioned edge detectors are some of the edge detection methods proposed in literature. Other edge detection algorithms such as Perwitt Operator and Zero Crossing Operator can be found in in [7], [8], respectively.

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 became feasible to achieve intelligent edge detection with improved noise tolerance. For example, a novel genetic-neuro-fuzzy system is proposed for edge detection of ultrasound images in [9]. The author has utilized competitive neural networks (NN) as the edge classifier in his proposed method. An on-line genetic algorithm (OGA) has been used to optimize and regulate the system parameters. The method has proved better performance in terms of noise resistance compared to the 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) has been utilized in the edge classification and compared with the canny edge detector under different noise levels.

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 output of six edge detectors has been used for that purpose, namely Sobel, Perwitt (similar to Robert), Roberts, Canny, LoG and zero-crossing edge detectors. 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 [11] environment.

This paper is organized as follows. Section II provides a brief theoretical background of the classifiers used and PCA. In Section III, the methodology of the proposed method is discussed. Section IV presents the Data used throughout the experiments and the model parameters selected. IN Section V, the results are presented with the author comments and discussion. Finally, conclusions are drawn in Section VI.

II. THEORETICAL BACKGROUND

A. NBC

Naïve Bayes Classifier is a probabilistic classifier used to estimate the probability distribution. To simply this estimation, the NBC make the assumption of feature independence. Therefore the joint likelihood can be written as:

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

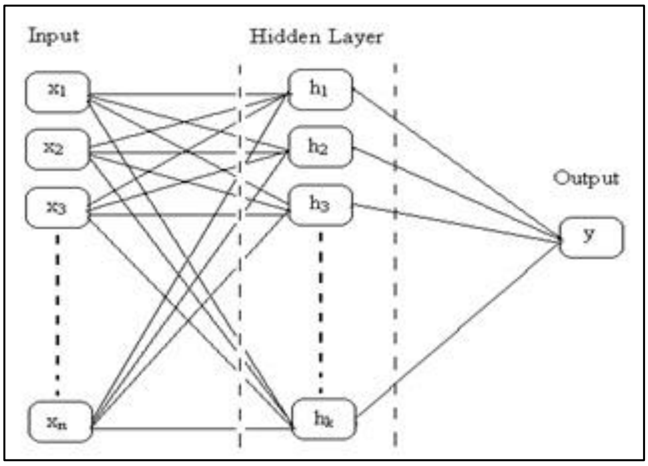

Fig. 1. Architecture of Artificial Neural Networks

The joint probability distribution could 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. ANN

Artificial Neural Networks is a classification technique inspired by the neurons in the human nervous system. The ANN is based on the perceptron theory. The advantage of ANN is that it can be adapted to learn non-linearly separable data. ANN consists of multiple perceptron layers, and therefore is often called Multiple Layer Perceptron (MLP). The structure of ANN is shown in Fig. 1.

The first layer of the ANN is called the input layer, in which the input sample point is applied. Each neuron at the input layer represents one feature. Therefore, the size of the input layer is equivalent to d (dimension). The second layer is called the hidden layer. This additional layer adds the non-linearity to the conventional single layer perceptron algorithm. The size of the hidden layer is determined by the complexity of the classification problem. That is, more number of neurons is required for higher order of non-linear separable data. 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. Therefore, number of neurons in the output layer is determined by number of classes.

The neurons are sometimes called activation functions. The activation function is chosen to be a logistic sigmoid function in the hidden and the output layers for the purpose of classification and linear function for the purpose of regression. The neurons at the input layer are always linear.

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

C. 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 [12].

The basic PCA perform the following 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:

$$z_{im} = x_{im} - \mu_d \tag{5}$$

where x_{im} is the ith observation corresponding to the feature m.

2) Find the Eigen vectors and Eigen values of (n-1) × covariance matrix:

$$\Sigma = \frac{1}{n-1} z_{im} {z_{im}}^{T} \tag{6}$$

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. Methodology of the Proposed Method:

A. Classification Problem Formulation

The methodology of the proposed method is depicted in Fig. 2. For the sake of brevity, the image is processed through the six Edge Detectors (EDs). The output of each detector is converted from a matrix format (image representation) into a vector format. Each pixel of the resulting vector is an observation or a sample point and each of the edge operators represent a feature or an attribute. The resulting vectors are combined to form the classification matrix as shown in in Fig. 2. The classification matrix has NM 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 labelling 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

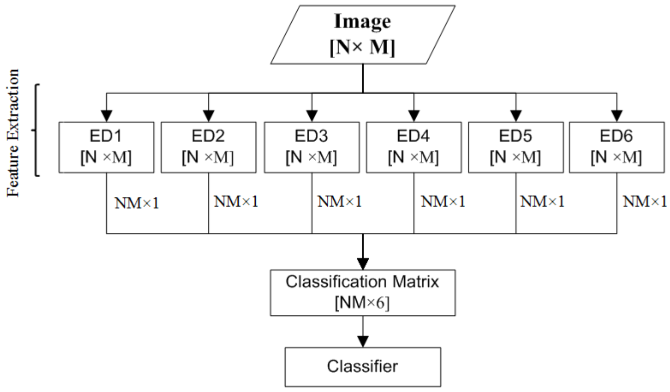

Fig. 2. Methodology of the Proposed Method

returns logical 0 if otherwise. In order to provide a reliable labelling system, a threshold limit was used to control the process of labelling. 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 labelling 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, hence providing more flexibility towards the need of the application. That is, if the user has set the threshold to 1, it means that at least one edge detector should detect the corresponding pixel as an edge which ensures the detection of weak edges that are detected by Canny and not detected by Roberts for example. This enables the fused based edge detector to sweep among the six edge detectors with respect to the desired application. From now on, the threshold is denoted by α.

C. Classifiers used in the Proposed Method

From A and B, the classification matrix with the corresponding pixel labels are 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 level 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.

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 of 3 (α=3) and threshold of 4 are shown in Fig. 3 and Fig.4. 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.

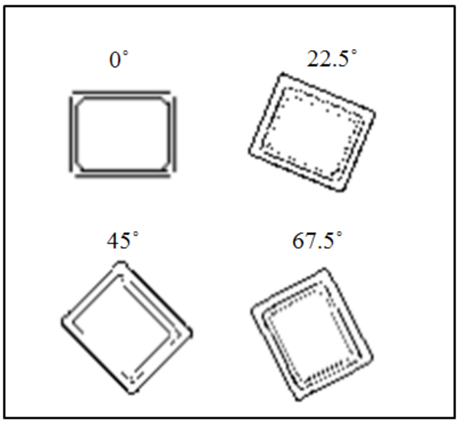

Fig. 3. Edge Targets for the Training data depicted with threshold=3

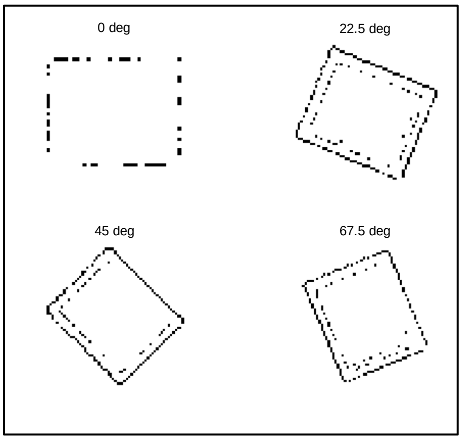

Fig. 4. Edge Targets for the Training data depicted with threshold=4

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 the later sections were produced using a threshold of 3 in some parts and a threshold of 4 in other parts. This was done to present the effect of changing the threshold on the proposed edge detector.

B. *Model Parameters*

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

1) *Threshold limit α:*

Different values of α has been used ranging from 1 to 6. It has been found that reducing α would increase the sensitivity of edge detection on the expense of noise tolerance. Therefore, it was concluded that a value of α=4 would provide a good compromise between both 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. There was no significant effect on the classifier performance beyond 10 hidden units; therefore, 10 hidden units were adopted throughout the experiments.

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 discriminant ones and yield to almost perfect classification. Therefore, the two features corresponding to the maximum variance was adopted throughout the paper.

V. Results and Discussion

A. *Examination of Classifiers Immunity to Noise*

To study the noise effect of 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 Fig 5.

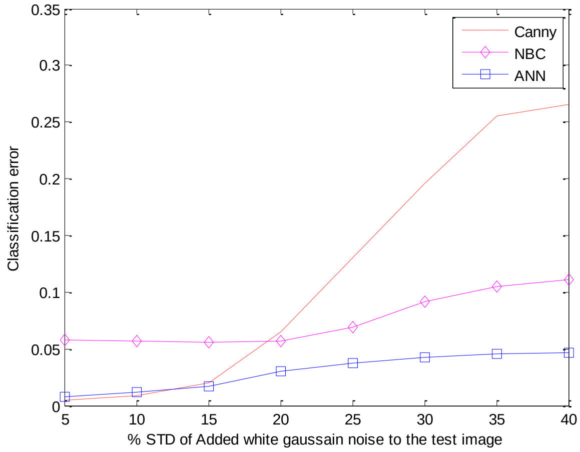

Fig. 5. Edge Targets for the Training data depicted with threshold=4

Fig. 6. Real Image employed in Testing (source: *http://pdfguides.com*)

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 will change the results. As mentioned earlier, the selection of α was based on a trade-off between sensitivity and noise immunity.

It can be noticed from Fig. 5 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, it performs reasonably satisfactory under high noise levels; however, it has the lowest accuracy (sensitivity) at lower noise levels.

B. Tests on Real images

To verify the results obtained in Fig. 5, the proposed fusion-based edge detectors (NBC and ANN) were tested under real life images at different noise levels with same threshold level (α=4). Naturally, real images differ in their complexity due to their different geometric nature. An intermediate-complexity real image was employed in the testing as shown in Fig. 6. 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 NBC, ANN, and Canny are shown in Fig. 7, Fig. 8 and Fig. 9, respectively.

The following observations can be made from the previous results:

1) It can be noticed that the canny edge detector gives the best edges at the no noise case while amplifies noise at high noise levels. This proves the results presented in Fig. 5.
2) Fig. 8 verifies the fact that ANN has the highest immunity to noise. In addition, according to Fig. 5, ANN should have close sensitivity to the canny edge detector; unfortunately, this was not exactly the case.
3) NBC classifier shows reasonable sensitivity at the noiseless case and performs much better than the Canny at high noise levels.

It can be concluded that there is always 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 strong edges. It is also worth mentioning that the results for the real image were produced based on α=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.

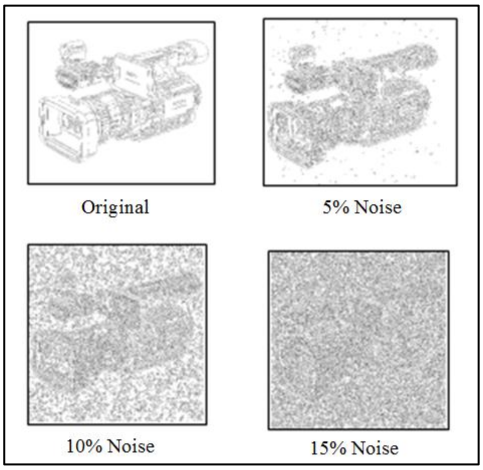

Fig. 7. Canny Fusion Edge Detector applied to the real image

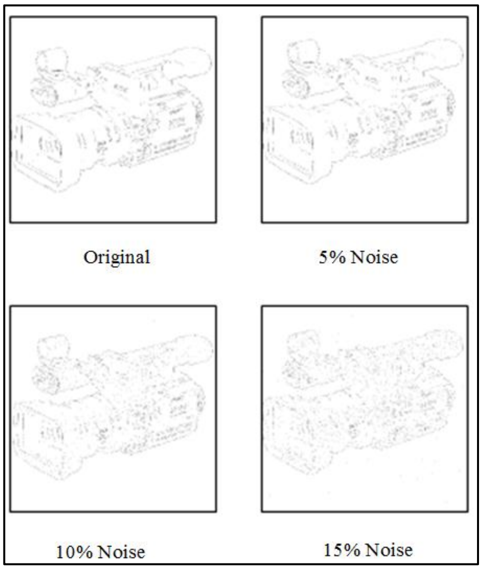

Fig. 8. ANN Fusion Edge Detector applied to the real image

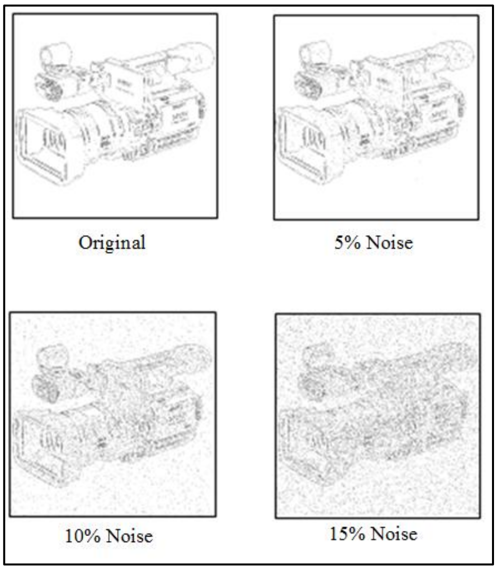

Fig. 9. NBC Fusion Edge Detector applied to the real image

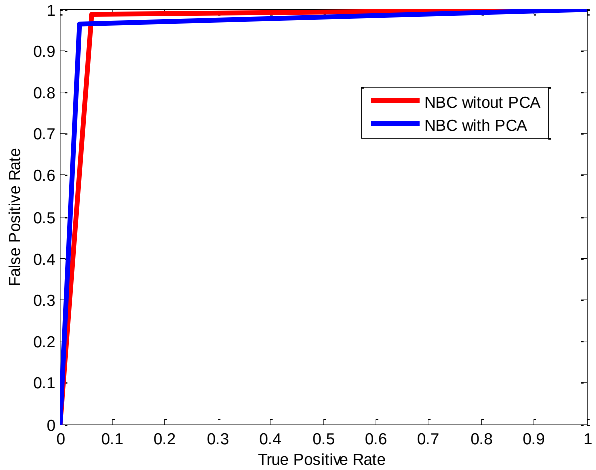

Fig. 10. ROC curve of NBC without PCA (red) and ROC curve of NBC with PCA (blue) with 5% noise added to the test image

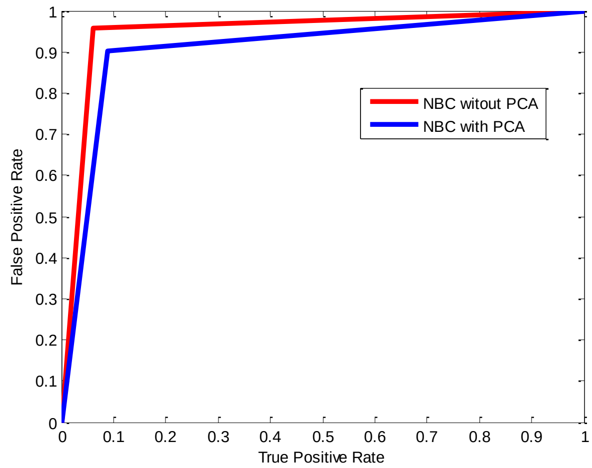

Fig. 11. ROC curve of NBC without PCA (red) and ROC curve of NBC with PCA (blue) with 20% noise added to the test image

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

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

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 the section II, the PCA preserves largest variances in the data. As a proof-of-concept, the testing 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 Fig. 10. The classification error and Area Under the Curve (AUC) are illustrated in Table. I.

As seen in Fig. 10, 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.

It should be noted that increasing the noise level can potentially affect the resulted PCA new vector space. Therefore, two dimensions might not be sufficient to produce similar results. That is, more features could be needed, e.g., three features with the largest variances instead of two might be needed in higher noise cases. 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. The comments are subjective and further investigation is needed to verify this.

Fig. 11 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 efficiency.

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 astonishing results of the NBC with PCA could be summarized due to the fact that some features are redundant and could be considered as null in the classification process.

The labelling 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 tunning the proposed edge detector to work as one of the six detectors, hence achieving more flexibility based on the application under hand.

Room for improvement is an open research. The following are some comments and recommendation for future improvement.

1) Based on the noise level, higher weights could be assigned to noise resistive edge operators such as Roberts and lower weights are assigned to less noise resistant edge detectors such as canny. This would improve the results of NBC and ANN.
2) To improve the sensitivity of ANN at low noise levels. Lower value of threshold could be used. However, this would decrease its noise resistance at higher noise levels. Therefore, it is recommended to train the NN with real images, so its generalization rate improves.
3) PCA is very powerful tool in dimensionality reduction as long as the direction of maximum variance is useful in classification. In the edge detection problem, direction of maximum variance is useful for classification, but it can be highly distorted under high level of noise.
4) It was concluded that the PCA best set of features alter as the level of noise increases. It is expected that the PCA will perform better if higher weights were given to the edge operators with higher noise immunity.
5) Experiments on ANN with PCA process data could be made to verify the workability of PCA with different classifiers and improve noise resistance.

ACKNOWLEDGMENT

I would like to thank Dr. Woon and Dr. Zeyar for their valuable help and guidance throughout the paper.

REFERENCES

[1] D. Ziou and S. Tabbone. “Edge detection techniques” - an overview. *International Journal of Pattern Recognition and Image Analysis*, vol no., pp. 537–559, 1998.

[2] W. Xiao, X. Hui. *An Improved Edge Detection Method for Image Corrupted By Gaussian Noise.* Computer and Computing Technologies in Agriculture II, Volume 2, 2009.

[3] I. E. Sobel. *Camera Models and Machine Perception*. PhD thesis, Electrical Engineering Department, Stanford University, Stanford, CA.,1970.

[4] L. G. Roberts. Machine perception of three-dimensional solids. In J. T.Tippett, editor, *Optical ond Electro-optical Information Processing*. MIT Press, Cambridge, MA, 1965.

[5] J. Canny. A computational approach to edge detection. *IEEE Transaction Pattern Analysis Machine Intelligence,* 8(6):679–698, 1986.

[6] D. Marr and E. Hildreth. Theory of edge detection. *Proc. R. Soc. Lond.*, 207:187–217, 1980.

[7] J.M.S. Prewitt and M.L. Mendelsohn, “The Analysis of Cell Images,” Ann. New York Academy Science, vol.128, pp. 1035-1053, 1966.

[8] Parker, James R., Algorithms for Image Processing and Computer Vision, New York, John Wiley & Sons, Inc., 1997, pp. 23-29.

[9] A. Rafiee, M. Hasan and M. Farzaneh. A Novel Genetic-Neuro-Fuzzy Filter for Speckle Reduction from Sonography Images. *Proceedings of the 6th WSEAS international conference on Evolutionary computing, 285-290, 2005.*

[10] W. L. Woon, P. Liatsis, K. D. Wong, Fusion Of Multiple Edge Maps For Improved Noise Resistance. (Paper Provided by Dr. Woon for the course project).

[11] Matlab v 7.11.0.584 (R2010b). Computer software. MathWorks, Inc Natick, MA, 2010

[12] Duda, Hart and Stork. *Pattern Classification*, 2nd edition, Wiley, 2001.