COMPARING COLOR EDGE DETECTION TECHNIQUES

Hasan İhsan Turhan
Middle East Technical University

COMPARING COLOR EDGEDETECTION TECHNIQUES

Hasan Ihsan Turhan
Electrical and Electronics Engineering Department
Middle East Technical University
Ankara, Turkey
hasan.turhan@metu.edu.tr

Gozde Sahin
Electrical and Electronics Engineering Department
Middle East Technical University
Ankara, Turkey
gozde.sahin@metu.edu.tr

Aydan M. Erkmen
Electrical and Electronics Engineering Department
Middle East Technical University
Ankara, Turkey
aydan@metu.edu.tr

***Abstract*—This study is a comprehensive comparative study on color edge detection, which analyses both monochromatic and vector-valued color edge detection techniques by using their gray-level counterparts. For this analysis monochromatic Sobel, Laplacian of Gaussian (LoG) and Canny are used with vector-valued Canny and Cumani. The focus of the study is on vector-valued color edge detectors. All used color edge detectors are further corrupted with noise and their noise susceptibilities are analyzed performance wise.**

Keywords: color edge detection, monochromatic techniques, vector-valued techniques

1. INTRODUCTION

Edges, which define the boundaries of objects, are traditionally caused by a sudden change in the intensity level of the image. However, the existence of an edge cannot always be recognized by the change in intensity. There may be cases where objects have the same illumination but varying hue [1]. Therefore, the classical gray-level approaches like Sobel, Laplacian of Gaussian (LoG) and Canny may fall short in finding these special edges. From this problem arises the need for color edge detectors.

Color edge detection is not as comprehensively researched upon as the classical gray-level edge detection due to the fact that almost 90 percent of the color edges of a color image are the same as the gray-level edges [2]. However, in some applications like object tracking, object recognition and image segmentation, the difference color edge detectors make for the better becomes more important.

The importance of color edge detection can best be seen in Fig. 1.In the figure, there exists a significant difference when the colored image which have grids of different colored squares is processed with color and gray-level versions of the Canny operator. The grids in the image have the same intensity value throughout, while having different hue values for different grids. Processing with a gray-level edge detector, the algorithms cannot see the difference between hue and therefore are not able to find any edges as in Fig. 1(c), while the color version of the Canny operator can distinguish between the different colored grids, as seen in Fig. 1(b). [1]

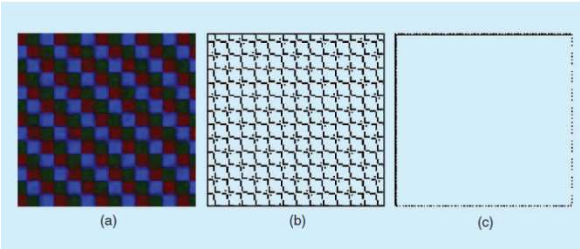


Figure 1: (a) Original color image consisting of three color grid pattern of same intensities. (b) Results for color variant of Canny operator. (c) Results for gray-level version of Canny operator. [1]

There are two main ways to apply a color edge detector. These two techniques are differentiated by their use of the color components of a pixel. If the detector is applied to each color component separately and the results are merged in the end, the technique is called a monochromatic technique. On the other hand, if the color component values are combined during the process of detection, the technique becomes a vector-valued technique. The details on the various methods used in this study can be found on Section 2.

In the literature, there does not exist a work which analyzes and compares the edge detection techniques in color images comprehensively [1, 6, 7]. Therefore, the contribution of this study to the literature is to examine the color edge detection methods, compare them with their gray-level counterparts and each other as well as analyzing their noise susceptibility extensively.

This paper consists of the following sections: Section 2 examines the color edge detection methodology in more detail. Section 3 contains the experimental results of the study. The paper ends with the conclusions in Section 4.

2. COLOR EDGE DETECTION

The color image varies from the gray-level image by having a vector of color components assigned to each pixel instead of a single value of intensity. Therefore, color edge detection deals with vector-valued image functions instead of scalar ones. The vector-valued image function can be processed using two main principles which are either dealing with the individual color component functions separately turning the problem into a scalar one (Monochromatic-based

Techniques), or using the color component vectors directly to obtain results (Vector-Valued Techniques).

2.1 Monochromatic-Based Techniques

Monochromatic-based color edge detection techniques treat each color vector component separately. Therefore, if you have three color channels of red, green and blue (RGB) in your color vector, a monochromatic method takes red, green and blue intensity values as three separate images and apply a gray-level color edge detection technique to each image. Afterwards, the three edge results are merged considering if an edge exists in one channel, it exists in the overall image. An illustration of this process can be viewed in Fig. 2.

Most color edge detection methods are monochromatic as they generally generate better results when compared to the gray-level edge detection methods. Moreover, these techniques are quite straightforward and relatively easier to apply and understand than the vector-valued techniques which are more novel [1].

2.2 Vector-Valued Techniques

Vector-valued techniques in earlier publications suggested the use of vector differences in the place of intensity differences in order to detect edges. However, as the area of vector-valued techniques expanded, newer methods focused more on the difference between the norms of the color vectors as well as other methods. Two of these techniques are the color variants of Canny operator and the Cumani operator.

2.2.1 Color Variants of the Canny Operator

The color extension of Canny Operator consists of the same stages as the classical Canny operator. The philosophy of the traditional Canny operator is formed of determining the first partial derivatives of the image function (usually smoothed by a Gaussian) according to x and y dimensions and finding the magnitude and direction of the edges from these values using non-maxima suppression.

For a vector assigned at each pixel, the color variant of Canny suggests a first partial derivative matrix is generated. This Jacobian matrix, J, consisting of the partial derivatives of each channel can be seen in (1) [1].

$$J = \begin{pmatrix} R_x & R_y \\ G_x & G_y \\ B_x & B_y \end{pmatrix} = \begin{pmatrix} C_x & C_y \end{pmatrix} \tag{1}$$

The indices x and y in equation (1) denote the first partial derivatives of the color channel images in the x and y directions.

The eigenvector of J^TJ corresponding to the largest eigenvalue is then indicates the direction where the largest change (or discontinuity) in the three channel image function occurs. This direction θ of the color edge can be defined as in (2), determined by the norms of the color vectors in x and y directions [1].

$$\tan(2\theta) = \frac{2.C_x.C_y}{\|C_x\|^2.\|C_y\|^2} \tag{2}$$

The magnitude of that edge, denoted by m, can be found with (3).

$$m^2 = \|C_x\|^2.\cos^2(\theta) + 2.C_x.C_y\sin(\theta)\cos(\theta) + \|C_y\|^2.\sin^2(\theta) \tag{3}$$

The direction and magnitude are defined for each edge and then the non-maxima suppression is used. For the non-maxima suppression a threshold is necessary to eliminate the broad edges [5].

There are several versions of the color version of Canny operator. These versions differentiate from each other in terms of where the combination of the color vectors is used and what type of combination operator is used in the combination. Canny can be said to be divided into three processing steps [5]:

I) Determining the partial derivatives

II) Calculating edge direction and magnitude

III) Implementing non-maxima suppression.

The combination can be done in either of these steps using different combination operators, which are generally various mathematical norms like L_1-norm, L_2-norm or L∞-norm. Therefore, the above described color edge operator can be said to be type II/1 if an L_1-norm is used. Kanade [3] summarized the results in the work he proposed the color approach to Canny as:

- Although 90% of the color edges are identical to gray-level edges, geometries of objects can be better acknowledged by color edges.
- II/∞ operator, which computes the magnitude and direction of edges separately for each channel and then selects the edge with the strongest magnitude using L∞-norm, proved to be the best.
- I/∞was almost as good as II/∞ and was faster than II/∞ due to the merge of channels happening sooner.
- II/∞ produced almost the same results with the above described theoretical color Canny operator.

2.2.2 Cumani Operator

Cumani operator, suggested by Cumani [4] in 1989, utilizes the second order partial derivatives to find edges in color images. In Cumani, the color image, C, is treated as a two dimensional vector field as in equation (4).

$$C(x,y) = (C_R(x,y), C_G(x,y), C_B(x,y)) \tag{4}$$

In his work, Cumani suggests a new norm [4], the squared local contrast, which is a quadratic norm of directional derivatives toward the unit vector$n = (n_1.n_2)$, and can be seen in (5).

$$S(p;n) = K.{n_1}^2 + 2.F.n_1.n_2 + H.{n_2}^2 \tag{5}$$

In (5), K, F and H values correspond to the following:

$$K = \sum_{i=1}^{3} \frac{\partial C_i}{\partial x} . \frac{\partial C_i}{\partial x} \tag{6}$$

$$F = \sum_{i=1}^{3} \frac{\partial C_i}{\partial x} . \frac{\partial C_i}{\partial y} \tag{7}$$

$$H = \sum_{i=1}^{3} \frac{\partial C_i}{\partial y} . \frac{\partial C_i}{\partial y} \tag{8}$$

Cumani states that [4], the eigen values of the matrix in (9) coincide with extreme points of the norm defined above and can be obtained if the unit vector n is the corresponding eigenvector, which enables the equations in (10,11) to be written in order to find these extreme eigen values and their corresponding eigenvectors.

$$A = \begin{pmatrix} K & F \\ F & H \end{pmatrix} \tag{9}$$

$$\lambda_{\pm} = \frac{K+H}{2} \pm \sqrt{\frac{(K+H)^2}{4} + F^2} \tag{10}$$

$$n_{\pm} = (\cos(\theta_{\pm}), \sin(\theta_{\pm})) \tag{11}$$

$$\theta_{+} = \begin{cases} \frac{\pi}{4}, \ if (K-H) = 0 \ and \ F > 0, \\ -\frac{\pi}{4}, \ \ if (K-H) = 0 \ and \ F < 0, \\ undefined, \ if \ K = F = H = 0, and \\ \frac{1}{2} \tan^{-1}\left(\frac{2.F}{K-H}\right), \quad otherwise \end{cases} \tag{12}$$

$$\theta_{-} = (\cos(\theta_{\pm}), \sin(\theta_{\pm})) \tag{13}$$

After the calculation of the extreme eigen values and eigenvectors, the minimum extremes can be omitted because they give the edges with the weakest magnitude and those are not the points of interest. Therefore, only the variables with + indices remain.

The extreme eigen values λ_{+} are defined previously as the maximum of the norm $S(p; n_{+})$ defined in (5) [1].Therefore, the maxima of the eigen values $\lambda_{+}$is searched for next, deriving the eigen value function in (10). The zeros of this derivation are sought in order to find the edges, which corresponds to looking at the zero crossings of the derivative of the norm defined in (5).

$$D_{S(p;n)} \coloneqq \nabla\lambda_{+}.n_{+} = K_x.n_1^3 + (K_y + 2.F_x).n_1^2.n_2 + (H_x + 2.F_y).n_1.n_2^2 + H_y.n_2^3 \tag{14}$$

This clearly illustrates that the equation in (14) is a form of second order directional derivative, whose zero crossings are sought to detect edges in an image just like in gray-level methods.

3. EXPERIMENTAL RESULTS

The experiments made in this study aim to analyze the chosen color edge detection techniques in such a way that it gives a broad sense of why they should be used and what are the advantages and disadvantages of using specific methods.

For the purpose of analyzing color edge detection, five different color edge detectors are used in this study. Three of these edge detectors are monochromatic-based techniques and the remaining two belong to the vector-valued edge detection principle. For monochromatic-based methods, three classical approaches for gray-level edge detection were chosen: Sobel, Laplacian of Gaussian (LoG) and Canny operators. These three operators are especially useful to recognize the difference color edge detection makes when compared to their monochromatic-based color edge detector versions.

The vector-valued techniques chosen in this study are the color variant of Canny and Cumani operator as explained in Section 2. Among these two edge detectors vector-valued Canny have the advantage of being comparable to both the gray-level version of Canny and the monochromatic. However, Cumani's performance can only be compared generally to the performance of monochromatic-based techniques or even the color variant of Canny operator.

3.1 Gray-Level and Monochromatic-based Color Edge Detectors

In this part of the experiments, the aim is to observe the difference between the performances of gray-scale Sobel, LoG and Canny edge detectors and their monochromatic-based color edge detector versions. The results can be viewed in Fig. 2.The results illustrate that the edges found by the color edge detectors are mostly the same as the edges found by the gray-level edge detectors. This founding coincides with the fact stated in Section 1 which suggests that 90 percent of the color edges are the same as the gray-level edges. However, it can also be observed that the color edge detectors are able to detect edges their gray-level counterparts cannot.

When the results for the experiments in Fig.2 are investigated more closely, it is seen that the progress that stands out the most is Sobel. For Sobel, the increase in found edges between Fig.2(c) and Fig.2 (d) looks to be more than the edges found with LoG and Canny. This is due to the fact that LoG and Canny being considered better feature detectors than Sobel itself. As LoG and Canny are already able to find more edges in a gray-level image than Sobel, the percentage of edges that are the same between gray-level and color images in those is larger than the percentage of Sobel. Thus, the difference that the color detector makes is much more significant.

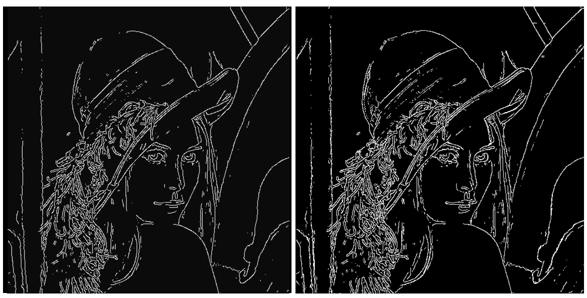
Moreover, when the gray-level LoG result in Fig.2(e) and the color LoG results in Fig.2(d) are examined, there are still differences that can easily be recognized, even though the progress in Sobel is much more obvious. These differences are the completion of some edges in the background, the more full imagery of the lips and the more complete boundaries of the hat.

Lastly, for the gray-level Canny result in Fig.2(g) and its color version in Fig.2(h), the results suggest that there is very little difference between the images for the reasons stated above. However, the progress made with the monochromatic color edge detector can be seen in Fig.2(h) where some lines are sharper than the ones the gray-level version found.

a)Original Gray Scale Image b)Original Color Image

c)Gray Scale Edge Detection with Sobel Operator d)Color Edge Detection with Sobel Operator

e)Gray Scale Edge Detection with LoG Edge Detector f)Color Edge Detection with LoG Edge Detector



g)Gray Scale Edge Detection with Canny Edge Detector h) Color Edge Detection with Canny Edge Detector

Figure 2 : Gray-Level and Monochromatic Color Edge Detection Results for the image "Lena"

3.2 Monochromatic-based and Vector-valued Color Edge Detectors

This part of the experimental results examines the results obtained with the vector-valued color edge detectors and also compares them to the monochromatic-based methods as well as with each other.The results for vector-valued color edge detectors Canny and Cumani can be viewed in Fig.3. While Fig.3(a) depicts the edges

a)Color Edge Detection with Vector-Valued Canny Operator b)Color Edge Detection with Cumani Operator

Figure 3 : Vector-Valued Color Edge Detector Results

found by the color variant of Canny described in Section 2, Fig.3(b) depicts the edges found by the complicated Cumani operator. When these two figures are compared, it is clear that the Cumani operator can produce better edge detection. Cumani is able to find the edges caused by the texture of the hair and carvings of the mirror, as well as finding the full line of edge of the wood in the background. It also fares better in the features of the hat.

As mentioned before in this section, the color variant of Canny has the advantage of being able to be compared with both the gray-level and monochromatic-based color Canny versions of this operator. When Fig.2(g), Fig.2(h) and Fig.3(a) are examined, it is observed that the performance of the vector-valued operator is not better than the monochromatic-based method. Furthermore, it also does not reach up to the level of the gray-levelCanny edge detector.

The reasons behind the vector-valued result being worse than the others may depend on the choice of threshold, the type of the norm used and the fact that making the combination of components later changes the outcome for the better as Kanade [3] suggested.

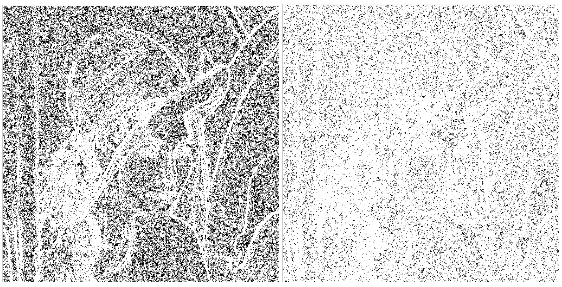
Furthermore, the performance of Cumani can be evaluated against the monochromatic-based color edge detectors. When Fig.3(b) is considered against Fig.2(d), Fig.2(f) and Fig.2(h), it is observed that apart from the problems in accuracy around the lips, Cumani operator produces as good results as the others if not better.

3.3 Noise Susceptibility

Noise susceptibility of an edge detector could play an important part in the way one chooses which operator to use. The existence of noise may cause some operators to become useless to the purposes of the application even though their abilities may be sufficient in practice. The experiments made in this study to see the responses of the utilized operators with respect to different levels of noise aims to give insight on which operator to choose in the existence of noise.

For this purpose, Gaussian white noises of different sigma values are added to the image and the edge detectors are applied. These sigma values are: 0.0005, 0.0025, 0.005 and 0.01. The output images for every edge detector and noise level were examined. Fig.4 shows the results obtained with the monochromatic Sobel edge detector. As can be viewed in Fig.4(d), the edges almost disappear with increasing amount of noise. Therefore, one might conclude, it does not serve to any purpose using Sobel in the presence of noise. In Fig.5, LoG seems to be affected by noise while still being able to keep the forms, as seen in Fig.5(d).

a)Color Sobel(sigma=0.0005) b)Color Sobel(sigma=0.005)



c)Color Sobel(sigma=0.0025) d)Color Sobel(sigma=0.01)

Figure 4 : Monochromatic Sobel Noise Performance

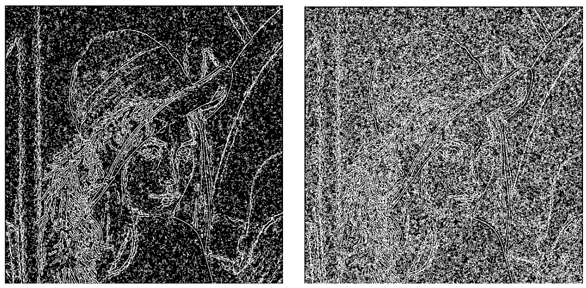
a)Color LoG(sigma=0.0005) b)Color LoG(sigma=0.005)

c)Color LoG(sigma=0.0025) d)Color LoG(sigma=0.01)

Figure 5 : Monochromatic LoG Noise Performance

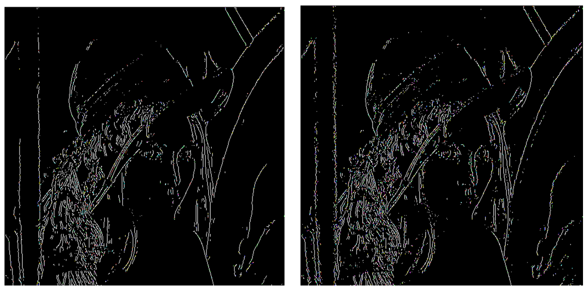
Next, the responses of monochromatic-based Canny and vector-valued Canny are examined. While Fig.6 illustrates the noise performance of monochromatic Canny, Fig.7 shows the performance of vector-valued Canny. From these we can deduct that even though vector-valued Canny fails to find as many edges as monochromatic-based version can, it is more durable to noise. When corrupted with the highest amount of noise used in this experiment, monochromatic Canny produces more fake edges than the vector valued version as seen in Fig.6(d) and Fig.7(d).

a)Mono-Canny(sigma=0.0005) b)Mono-Canny(sigma=0.005)

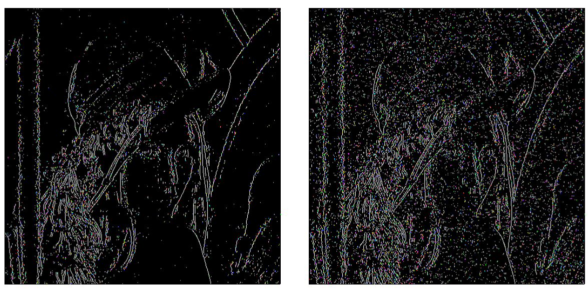


c)Mono-Canny(sigma=0.0025) d)Mono-Canny(sigma=0.01)

Figure 6 : Monochromatic Canny Noise Performance

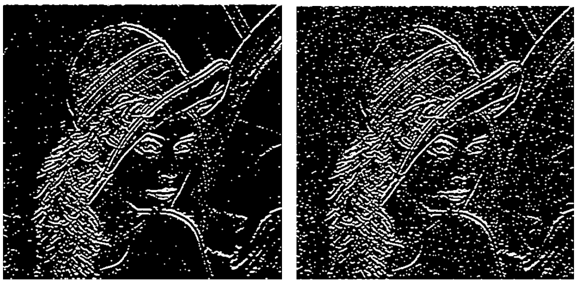


a)Vector-Canny(sigma=0.0005)b)Vector-Canny(sigma=0.005)

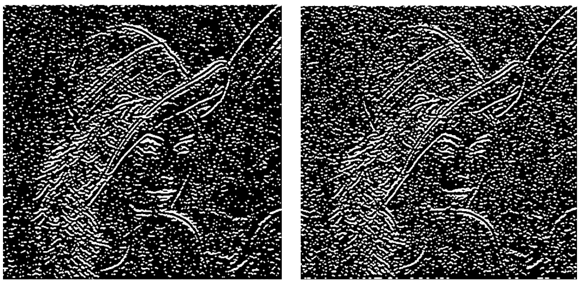


c)Vector-Canny(sigma=0.0025) d)Vector-Canny(sigma=0.01)

Figure 7 : Vector-Valued Canny Noise Performance



a)ColorCumani(sigma=0.0005)b)Color Cumani(sigma=0.005)



c)Color Cumani(sigma=0.0025)d)Color Cumani(sigma=0.01)

Figure 8 : Cumani Noise Performance

Lastly, in Fig.8, the performance of Cumani can be seen. It is similar to the performance of LoG which is to say that while keeping the main edges in the image intact; it still produces a lot of noise. The overall results show that, as the noise increases the image almost disappears for Sobel, in Fig.4 but the others fare better. With LoG, in Fig.5, it can be said that although there are many irrelevant edges found, the image continued to keep the main edges dominant, just like Cumani in Fig.8. For Canny, in Fig.6 and Fig.7, it can be said that the vector-valued method is more advantageous against noise. Moreover, the vector-valued Canny can also be considered as the least susceptible operator to noise, as even Cumanigenerated many fake edges. This fact may put the vector-valued Canny in a position where it can be preferred among these five operators regardless of the fact that it may fail to find dome of the edges in the image.

4. CONCLUSION

In this study, color edge detection techniques were analyzed and comparisons between different methods were presented.

Firstly, the differences between the monochromatic-based color edge detectors and their gray-level counterparts were examined and it was seen that the monochromatic-based color edge detectors produced better results than the gray-level versions.

Next, the vector-valued techniques were analyzed with respect to the monochromatic and gray-level edge detectors. It was observed that the vector-valued version of Canny could not generate as many edges as its monochromatic version or even its gray-level version due to reasons stated in Section 3. Moreover, it was seen that Cumani does produce better results than the vector-valued Canny while its results are almost as good as the monochromatic-based methods.

Lastly, the noise susceptibility experiments were made on the five color edge detection methods used in this study. As an outcome of this experiment, it can be suggested that the vector-valued Canny performs best against noise and therefore can be advantageous among the others for applications that need to deal with noise. This fact can make the vector-valued Canny preferable even though it cannot generate as many edges.

REFERENCES

[1] A. Koschan, M. Abidi, "Detection and Classification of Edges in Color Images: A Review of Vector-valued Techniques," *IEEE Signal Processing Magazine*, pp. 64-73, January 2005

[2] C.L. Novak, S.A. Shafer, "Color Edge Detection," *Proceedings DARPA Image Understanding Workshop*, Vol. I, pp. 35-37, Los Angeles, CA, USA, February 1987

[3] T. Kanade, "Image Understanding Research at CMU," in *Proc. Image Understanding Workshop*, vol. II, 1987, pp. 32-40

[4] A. Cumani, "Edge Detection in Multispectral Images," *Comput. Vis. Graph. Image Process.: Graphical Models Image Processing*, vol. 53, no. 1, pp. 40-51, 1991

[5] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, no. 6, pp. 679-698, 1986

[6] A. Koschan, "A Comparative Study on Color Edge Detection," *Proceedings 2nd Aisan Conference on Computer Vision ACCV'95*, vol. 3, pp. 574-578, December 1995

[7] E. Nadernejad, S. Sharifzadeh, H. Hassanpour, "Edge Detection Techniques: Evaluations and Comparisons," Applied Mathematical Sciences, vol. 2, no. 31, pp. 1507-1520, 2008