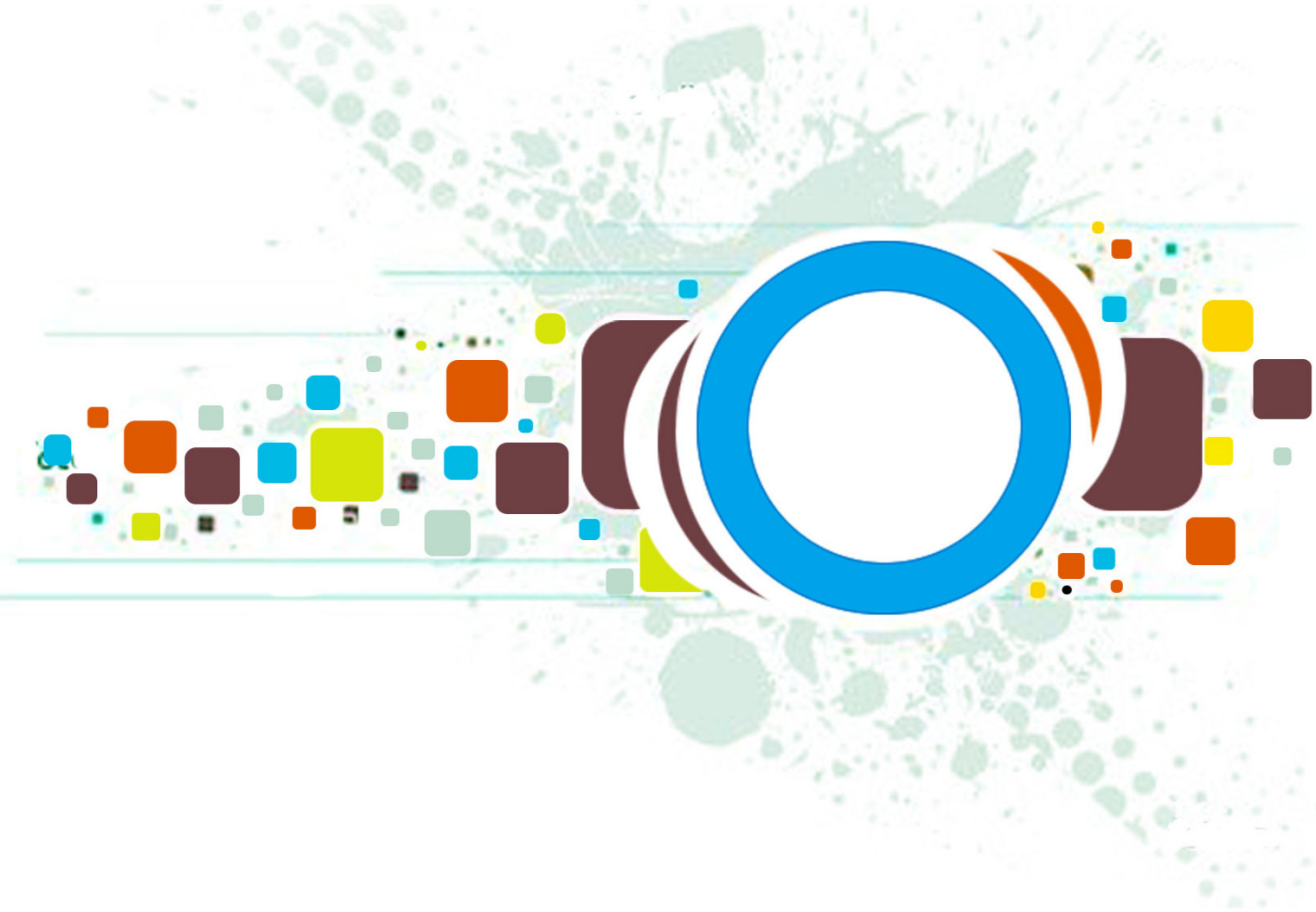
Volume 8 • Issue 6 • November / December 2014

INTERNATIONAL JOURNAL OF

IMAGE PROCESSING (IJIP)

ISSN : 1985-2304

Publication Frequency: 6 Issues Per Year



CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

VOLUME 8, ISSUE 6, 2014

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-2304

International Journal of Image Processing (IJIP) is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJIP Journal is a part of CSC Publishers

Computer Science Journals

http://www.cscjournals.org

INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

Book: Volume 8, Issue 6, November / December 2014
Publishing Date: 31-12-2014
ISSN (Online): 1985-2304

IJIP Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2014

EDITORIAL PREFACE

The International Journal of Image Processing (IJIP) is an effective medium for interchange of high quality theoretical and applied research in the Image Processing domain from theoretical research to application development. This is the *Sixth* Issue of Volume *Eight* of IJIP. The Journal is published bi-monthly, with papers being peer reviewed to high international standards. IJIP emphasizes on efficient and effective image technologies, and provides a central for a deeper understanding in the discipline by encouraging the quantitative comparison and performance evaluation of the emerging components of image processing. IJIP comprehensively cover the system, processing and application aspects of image processing. Some of the important topics are architecture of imaging and vision systems, chemical and spectral sensitization, coding and transmission, generation and display, image processing: coding analysis and recognition, photopolymers, visual inspection etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 8, 2014, IJIP appears with more focused issues. Besides normal publications, IJIP intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

IJIP gives an opportunity to scientists, researchers, engineers and vendors from different disciplines of image processing to share the ideas, identify problems, investigate relevant issues, share common interests, explore new approaches, and initiate possible collaborative research and system development. This journal is helpful for the researchers and R&D engineers, scientists all those persons who are involve in image processing in any shape.

Highly professional scholars give their efforts, valuable time, expertise and motivation to IJIP as Editorial board members. All submissions are evaluated by the International Editorial Board. The International Editorial Board ensures that significant developments in image processing from around the world are reflected in the IJIP publications.

IJIP editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJIP. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJIP provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Image Processing (IJIP)

EDITORIAL BOARD

TABLE OF CONTENTS

Volume 8, Issue 6, November / December 2014

Pages

Radial Fourier Analysis (RFA) Descriptor with Fourier-based Keypoint Orientation

S. C. F. Lin *stephen.lin@unsw.edu.au*
School of Mechanical and Manufacturing Engineering
The University of New South Wales
UNSW Sydney, NSW, 2052, Australia

C. Y. Wong *chin.wong@unsw.edu.au*
School of Mechanical and Manufacturing Engineering
The University of New South Wales
UNSW Sydney, NSW, 2052, Australia

G. Jiang *guannan.jiang@unsw.edu.au*
School of Mechanical and Manufacturing Engineering
The University of New South Wales
UNSW Sydney, NSW, 2052, Australia

M. A. Rahman md.arifur.rahman056*@unsw.edu.au*
School of Mechanical and Manufacturing Engineering
The University of New South Wales
UNSW Sydney, NSW, 2052, Australia

N. M. Kwok *nmkwok@unsw.edu.au*
School of Mechanical and Manufacturing Engineering
The University of New South Wales
UNSW Sydney, NSW, 2052, Australia

Abstract

Local keypoint detection and description have been widely employed in a large number of computer vision applications, such as image registration, object recognition and robot localisation. Since currently available local keypoint descriptors are based on the uses of statistical analysis in spatial domain, a local keypoint descriptor, namely Radial Fourier Analysis (RFA) keypoint descriptor, is developed with the use of spectral analysis in frequency domain. This descriptor converts image gradients around SIFT keypoints to frequency domain in order to extract the principle components of the gradients and derive distinctive descriptions for representing the keypoints. Additionally, a keypoint orientation estimate is also introduced to improve the rotational invariance of the descriptor rather than simply adopting SIFT keypoint orientations. The introduced orientation estimate employs the starting point normalisation of Fourier coefficients, which are frequency responses, to deduce rotating angles that ensure keypoint correspondences are aligned at the same orientation. Through experiments and comparisons, RFA descriptor demonstrates its outstanding and robust performances against various image distortions. Particularly, the descriptor has extremely reliable performances in dealing with the images, which are degraded by blurring, JPG compression and illumination changes. All these indicate that spectral analysis has strong potential for local keypoint description.

Keywords: Local Keypoint Descriptor, Keypoint Orientation, Fourier Transform, Keypoint Matching, SIFT Descriptor.

1. INTRODUCTION

Local keypoint detection and description have been frequently utilised as an integral system to accomplish diverse computer vision tasks due to its invariance capabilities against image distortions caused by many types of geometric and photometric transformations. For example, they have been widely applied to basic computer vision tasks including image retrieval [1], image registration [2] and object recognition [3]. As an extension from these basic tasks, they can be specifically designed to carry out object categorisation [4], texture classification [5] and face recognition [6]. In engineering, their applications could be related to medical diagnosis [7], robot localisation [8], land cover surveillance [9] and machine operation monitoring [10].

From the perspective of computer vision, local keypoint detection and description should be treated as independent processes as they produce different outputs and serve different purposes. Many keypoint detectors [11] have been proposed to search stable and repeatable keypoints in images, and keypoints generally contain the information of position and other supplementary measures to localise and define the properties of keypoints. One of the supplementary measures could be keypoint orientation that is commonly assigned to keypoints for rotational invariance. For example, SIFT detector [3] estimates keypoint orientation by creating a histogram of the image gradient orientations around a keypoint. Furthermore, ORB detector [12] defines keypoint orientation as the vector between a keypoint position and the intensity centroid of local image patch of the keypoint. Taylor and Drummond [13] compute keypoint orientation by simply choosing the longest vector among the vectors that are formed by intensity differences around a keypoint. All these keypoint orientation estimates are established on gradient and statistical approaches in spatial domain. It is advantageous if a different keypoint orientation estimate, such as using spectral analysis in frequency domain, can be developed to provide accurate keypoint orientations.

For keypoint description, it extracts the features concealed in the local image patches around keypoints and analyses the features to represent that keypoints. A variety of keypoint descriptors [14] have been devised to be invariant to geometric and photometric transformations such as image distortions caused by blur, rotation, scale, illumination and JPG compression changes. Many keypoint descriptors, such as SIFT [3], PCA-SIFT [15], and GLOH [14] descriptors, utilise histograms to determine the distributions of local image patches around keypoints in order to extract and analyse features, while achieving results that are invariant to image distortions. Summation of image gradients is also a robust approach in keypoint description, which is demonstrated by SURF descriptor [16]. Furthermore, BRISK descriptor [17] uses the comparisons of local image intensities around keypoints as features to represent the keypoints. Similar to the issue raised in keypoint orientation estimates, most keypoint descriptors are based on voting or statistical approaches in spatial domain. A local keypoint descriptor that extracts and analyses features in a different domain is essential as it could expand the diversity of local keypoint descriptions to increase its distinctiveness.

To address the overdevelopment of statistical analysis in spatial domain for keypoint orientation estimations and keypoint descriptions, this work relies instead on spectral analysis in frequency domain. For the keypoint orientation estimate, it utilises Fourier transform to convert the image intensities of local image patches to frequency domain from spatial domain. Afterward, starting point normalisation is performed on the intensity frequency to derive keypoint orientations in order to enhance the rotational invariance of keypoints. The proposed keypoint descriptor, which is named as Radial Fourier Analysis (RFA) descriptor, transforms the image gradients of local image patch to frequency domain, then the gradient frequency is decomposed to provide a unique keypoint description. Both of the transformed frequencies are not inverted back to spatial domain as the keypoint orientation and description are directly carried out in frequency domain.

The rest of the paper is organised as follows. In Section 2, the principle of SIFT keypoint detector is briefly revised, and a keypoint orientation estimate is developed to replace SIFT keypoint orientation. Section 3 presents the procedures to generate RFA descriptor with the reasons declared for each procedure. This is followed by Section 4, where an experiment setup to

evaluate the performances of the new descriptor is detailed. The performance results and comparisons are illustrated in Section 5. Significant observations from the results are compared and highlighted for discussions. Finally, in Section 6, all these works are concluded by suggesting possible research directions in accordance with the performances of the proposed keypoint descriptor and orientation estimate.

2. IMAGE KEYPOINT DETECTION

In this section, SIFT keypoint detector is briefly explained as SIFT keypoints are utilised to generate RFA descriptors. In order to improve the rotational invariance of SIFT keypoints, a novel estimate of keypoint orientation using Fourier normalisation is proposed to replace SIFT keypoint orientation.

2.1 SIFT Keypoint Detector

SIFT descriptor has been concluded by [18] [19] as the state-of-the-art in the research and development of local keypoint descriptor. As such, SIFT keypoints are used to localise salient points because of its trustworthy characteristics as well as providing a fair comparison base of describing power between our descriptor and SIFT descriptor. Fundamentally, SIFT keypoints are detected by utilising Difference of Gaussian (DoG) operator that was initially introduced by Crowley and Parker [20]. However, Lowe's extension of the operator [3] to detect scale invariant keypoint in a given image is more widely used and is adopted in this work.

SIFT keypoint detector involves the following three steps: (1) scale-space extrema detection; (2) keypoint localisation; and (3) orientation assignment. The first step is DoG operation, where input image is convoluted with Gaussian kernels at different scales to produce a series of Gaussian images. Adjacent images in the series are subtracted from each other, forming DoG images. A $3 \times 3 \times 3$ window is then employed to search the local maximums and minimums in the DoG images, and these extremes are labelled as potential keypoints. The second step filters potential keypoints by measuring their contrast and edge response, within which the keypoints with low contrast and high edge response are discarded. The remaining keypoints are subsequently re-localised to sub-pixel accuracy and identified as SIFT keypoints. In the final step, the dominant gradient orientation of each SIFT keypoint is estimated through binning the gradient orientations around the keypoint into a histogram and the peak of the formed histogram is taken as the gradient orientation of keypoint. The dominant gradient orientations are estimated and assigned to SIFT keypoints to enhance its rotational invariance ability. The last two steps are expanded from DoG operator for stabilising the keypoints in matching and recognition tasks.

2.2 Keypoint Orientation Estimate by Fourier Analysis

Other than histogram-based estimate of keypoint orientation, such as SIFT keypoint orientation, RFA descriptor adopts a simple and effective approach to compute keypoint orientation with Fourier analysis. This Fourier-based keypoint orientation estimate consists of three processing steps: (1) intensity sample collection; (2) Fourier transform; and (3) orientation estimate.

In the first processing step, the neighbouring image intensities around a keypoint are circularly sampled and arranged into a vector. The procedure of this intensity sample collection is presented in Algorithm 1. The intensity sample collection is preformed at the image scale where the keypoint is detected. For each sampling angle, R_o image intensities are sampled along the radial direction to represent the intensity signatures at that angle. This sampling process is performed at counter-clockwise direction till a full circular sampling is completed. The intensity vector, $z(m)$, has the image intensity samples arranged in an ascending order with respect to angular direction, and each sampling angle is represented by R_o image intensities. The intensity vector, $z(m)$, can be described as,

$$z(m) = \{I(u,v)\} \ , I(u,v) \in \mathbb{R} \qquad (1)$$

where, $m = 1 \cdots M$, $M = R_o \times 360°/\Delta\emptyset_o$ and $I(u,v)$ is the image intensity at a given sampling position (u,v). An illustration of this intensity sample collection is shown in Figure 1 with a radial sampling range of 5 pixel radius (i.e., $R_o = 5$) and an angular sampling interval of $10°$ (i.e., $\Delta\emptyset_o = 10°$) for demonstration purpose. Each black line connecting the red circles is a set of samples along the radial direction for that sampling angle. These sets of samples are then arranged into the intensity vector, $z(m)$, with the order from $0°$ to $360° - \Delta\emptyset_o$ as mentioned previously.

ALGORITHM 1: Intensity Sample Collection for Orientation Estimate by Fourier

Input gray image, I
Input keypoint position, (x_0, y_0)
Set radial sampling range, $R_o = 16$
Set angular sampling interval, $\Delta\emptyset_o = 7.5°$
Set sampling angle index, $\emptyset_i = 360°/\Delta\emptyset_o - 1$
Initialise z to a zero vector of the length $R_o \times 360°/\Delta\emptyset_o$
For $i = 0$ to $\emptyset_i$ do
  Determine sampling angle
  $\emptyset_o = \Delta\emptyset_o \times i$
  For $r = 1$ to R_o do
    Determine the sampling position
    $u = r \times \cos\emptyset_o + x_0$
    $v = r \times \sin\emptyset_o + y_0$
    Sample image intensity at $I(u,v)$
    Distribute the image intensity into z
  End for
End for
Output z

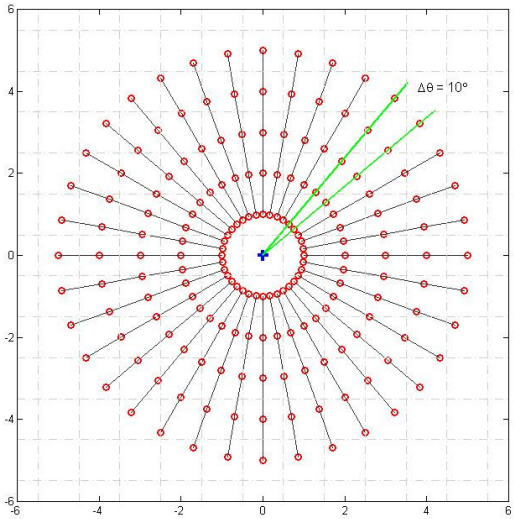


FIGURE 1: Intensity sample collection for keypoint orientation estimate. The red circles indicate the positions of samples and the blue cross sign represents the location of keypoint. Each black line connecting the red circles is a set of samples for that sampling angle.

For the second processing step, a discrete Fourier transform is performed on the obtained intensity vector, $z(m)$, that is,

$$Z(h) = \sum_{m=0}^{M-1} z(m)\, e^{\frac{-j2\pi mh}{M}}, \quad Z(h) \in \mathbb{C} \tag{2}$$

where, $h = -M/2 \cdots M/2 - 1$. Here, $Z(h)$ denotes the h^{th} Fourier coefficient (i.e., frequency response) based on the information given by $z(m)$ that is obtained by Algorithm 1. Note that, the Fourier coefficients, $Z(h)$, are shifted to the centre.

The final processing step estimates the keypoint orientation by using the phase of first positive Fourier coefficient (i.e., $Z(1)$). The phase is calculated by,

$$\rho = \tan^{-1}\left(\frac{\mathrm{Im}(Z(1))}{\mathrm{Re}(Z(1))}\right) \tag{3}$$

where, $\mathrm{Re}(\cdots)$ and $\mathrm{Im}(\cdots)$ denote the real and imaginary parts of the Fourier coefficient respectively. Since this calculated phase is in frequency domain, it must be converted back to spatial domain to properly express the phase as keypoint orientation with regard to the arrangement of image intensity samples in the intensity vector, $z(m)$. The conversion from phase, ρ, to keypoint orientation, θ, is defined as,

$$\theta = \rho \times \frac{M}{2\pi} \times \frac{\Delta\emptyset_o}{R_o} \tag{4}$$

The multiplication of $\rho \times M/2\pi$ implies that the phase is converted to spatial domain from frequency domain. After the conversion, the term $\Delta\emptyset_o/R_o$ aligns the converted phase to coincide with the angular sampling interval and radial sampling range in intensity sample collection.

The principle of this orientation estimate is based on the property of starting point of Fourier coefficients. For example, a complex signal is defined as $a(t)$ and its Fourier coefficients are defined as $b(w)$. Considering the signal is shifted by an amount of t_0 in its sequence that merely changes the starting point of the signal to $t = t_0$ from $t = 0$. The shifted signal is expressed as $a_p(t) = a(t - t_0)$, whose Fourier coefficients are

$$\begin{aligned} b_p(w) &= \sum_{t=0}^{T-1} a_p(t)\, e^{\frac{-j2\pi tw}{T}} \\ &= \sum_{t=0}^{T-1} a(t - t_0)\, e^{\frac{-j2\pi(T-t_0)w}{T}} \\ &= b(w) e^{\frac{j2\pi t_0 w}{T}} \end{aligned} \tag{5}$$

for $t = 0,1,2,\ldots,T-1$. Thus, shifting the starting point of the signal causes a linear phase shift in all Fourier coefficients that depends on the coefficient index (i.e., w) and the amount of shift (i.e., t_0).

It can be seen from Algorithm 1 that the keypoint orientation estimate is transformed to starting point normalisation as the neighbouring image pixels around a keypoint are sampled and arranged in the order corresponding to angular direction. Then, Eq. 3 determines the amount of linear phase shift that is imposed on the Fourier coefficients, $Z(h)$, obtained in Eq. 2. Lastly, Eq. 4 converts the linear phase shift to the change of starting point in the intensity vector, $z(m)$. It also reduces the computation cost by avoiding the use of inverse Fourier transform to determine the change of starting point in spatial domain.

Another issue to be considered while applying Fourier transform is the relationship between input and output of the transform. As the input of Fourier transform is a complex function, the output coefficients are difficult to predict and do not obey any symmetries since they are the sum of all the real and imaginary parts of the input. However, if the input of Fourier transform is a real function, the real and imaginary parts of the output coefficients are even and odd functions respectively. These two symmetries make the magnitude and phase of the output coefficients to become even and odd functions respectively.

For our keypoint orientation estimate, the phase calculated in Eq. 3 could either account for the first positive or negative Fourier coefficients (i.e., $Z(1)$ or $Z(-1)$) as they are the same in magnitude but opposite in direction. Nevertheless, the intensity sample collection in Algorithm 1 samples image intensities in counter-clockwise direction and analyses a non-zero area. The first positive Fourier coefficient, $Z(1)$, is therefore chosen to estimate keypoint orientation as well as to define its direction. The use of this coefficient is also suggested by Folkers and Samet [21] to normalise the Fourier coefficients that undergo the change in starting point.

3. IMAGE KEYPOINT DESCRIPTION: RFA DESCRIPTOR

In principle, keypoint descriptor aims to represent a keypoint by extracting main information within a minimum sampling window that is centred at the keypoint. The extracted information is further analysed to outline its features while maintaining the discrimination among other keypoint descriptors. The information can be extracted either in spatial domain or frequency domain with the uses of statistical or spectral analysis. For example, SIFT descriptor computes the image gradient orientations within the local image patch of a keypoint in spatial domain. Then, a statistical analysis is applied as the gradient orientations are distributed into orientation histograms that yield a 128 dimensional vector for representing the keypoint. In this work, RFA descriptor obtains the Fourier coefficients of image gradients to determine the main components of local image patches through spectral analysis in frequency domain.

The RFA descriptor is generated with the following steps: (1) gradient sample collection; (2) Fourier transform; and (3) principle frequency extraction. As previously mentioned, RFA descriptors are generated by using SIFT keypoints that contain the details of position, scale and dominant gradient orientation. In particular, the keypoint orientation estimated by Fourier analysis replaces the dominant gradient orientation in SIFT keypoints as it is more reliable and accurate.

Firstly, the local image gradients around a keypoint are circularly sampled in a manner that is similar to the intensity sample collection for Fourier keypoint orientation estimate. Here, image gradients specifically mean both horizontal and vertical gradients. The gradient sample collection process is detailed in Algorithm 2. Through the algorithm, it can be seen that the keypoint orientation is taken into account to determine the position of gradient samples to ensure that image gradients are correctly sampled with the same starting point as its corresponding keypoint. Figure 2 displays the sampling pattern of this gradient sample collection with a radial sampling range of 5 pixel radius (i.e., $R_d = 5$) and an angular sampling interval of $10°$ (i.e., $\Delta\emptyset_d = 10°$) for demonstration purpose. A slightly different sampling pattern is applied in gradient sample collection that the image gradients along angular direction are treated as the signatures at that sampling radius. It is shown by each black line that connects the red circles in Figure 2. As for the output of this step, 16 gradient vectors are obtained and they are express as,

$$s_r(n) = dx_r(n) + j\ dy_r(n) \tag{6}$$

where, sampled horizontal and vertical gradients are substituted into the real and imaginary parts of these complex vectors respectively. For Eq. 6, it has the variables $n = 1,2, \dots, N$ and $r = 1,2, \dots, R_d$, where $N = 360°/\Delta\emptyset_d$ and $R_d = 16$. The radial sampling range, R_d, indicates the number of gradient vectors and is set to this value due to the suggestion given by the SIFT descriptor performance evaluation for local keypoint matching [22].

ALGORITHM 2: Gradient Sample Collection for RFA Descriptor

Input gray image, I
Input keypoint details, (x_0, y_0, θ)
Set radial sampling range, $R_d = 16$
Set angular sampling interval, $\Delta\emptyset_d = 5°$
Set sampling angle index, $\emptyset_i = 360°/\Delta\emptyset_d - 1$
For $r = 1$ to R_d do
  Initialise s_r to a zero complex vector of the length $R_d \times 360°/\Delta\emptyset_d$
  For $i = 0$ to $\emptyset_i$ do
    Determine the sampling angle, $\emptyset_d = \Delta\emptyset_d \times i$
    Determine the sampling position
    $u = r \times \cos(\emptyset_d - \theta) + x_0$
    $v = r \times \sin(\emptyset_d - \theta) + y_0$
    Determine and sample image gradients at $I(u, v)$
    Distribute the image gradients into s_r
  End for
End for
Output s_r

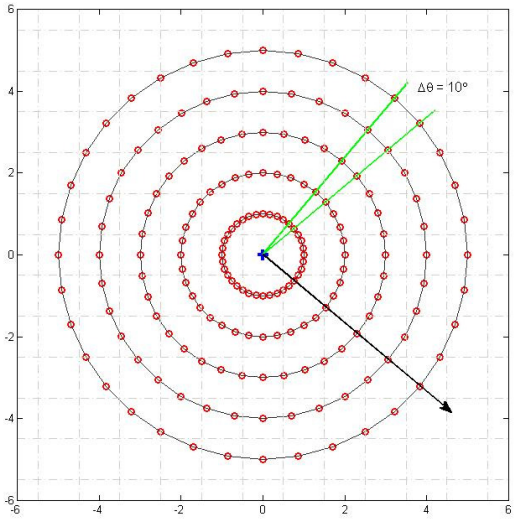


FIGURE 2: Gradient sample collection for RFA descriptor. The red circles indicate the positions of samples, the blue cross sign represents the location of keypoint and the black arrow directs keypoint orientation.

Secondly, prior to Fourier transform, unit vector normalisations are separately employed on the real and imaginary parts of each of the obtained gradient vectors, $s_r(n)$. The normalisations are used to increase gradient contrast while preserving the characteristics of the gradients. Then, a discrete Fourier transform is independently performed on each of the obtained gradient vectors to

observe the Fourier coefficients (i.e., frequency responses) at different radial distances from the keypoint. The transformations are defined as,

$$S_r(k) = \sum_{n=0}^{N-1} s_r(n)\, e^{\frac{-j2\pi nk}{N}} \tag{7}$$

for $k = -N/2 \cdots N/2 - 1$. The variable $S_r(k)$ denotes the k^{th} Fourier coefficient of the r^{th} gradient vector that is based on the information given by the $s_r(n)$ in Eq. 6. Again, these Fourier coefficients have been shifted to centre and are the output of the second step.

The last step manipulates the Fourier coefficients, $S_r(k)$, in order to extract the principle frequencies in these coefficients. It translates the coefficients into more meaningful analysis and distinctive representation of the neighbourhood of keypoints. RFA descriptor is generated by extracting four coefficients with the indices of $k = -2, -1, 1, 2$ from each set of the obtained Fourier coefficients, $S_r(k)$. The descriptor, V, is mathematically expressed as,

$$V = \{\mathrm{Re}(S_r(k)), \mathrm{Im}(S_r(k))\} \tag{8}$$

for $r = 1, 2, \ldots, R_d$ and $k = -2, -1, 1, 2$. Consequently, it results in a 128 dimensional vector since the real and imaginary parts of the extracted coefficients are treated as individual numbers. The vector is then normalised to unit length to suppress the influences induced by illumination changes. This normalised vector becomes the RFA descriptor.

It is essential to avoid or minimise the information loss associated with the generation of keypoint descriptor. For example, SIFT descriptor uses the gradient orientations weighted by their gradient magnitudes to analyse the local image gradients around keypoints. While the magnitude and orientation information of the gradients are preserved, this methodology also changes the characteristics of the gradients through the weighting operation. To prevent this situation, both horizontal and vertical gradients are taken into account for the gradient sample collection in the generation of RFA descriptor. Therefore, the information of gradients can be entirely preserved and used for Fourier analysis without altering its characteristics from the original.

During the manipulation of Fourier coefficients in the last step of RFA generation, all the centre coefficients (i.e., $S_r(0)$) have been ignored as these coefficients straddle in the line between positive and negative frequencies. Phrased differently, the coefficients contain the responses from both positive and negative frequencies that make them unstable and sensitive to geometric translation changes [23]. In addition, the same property is applied to the last Fourier coefficients (i.e., $S_r(-N/2)$).

Basically, the low frequency band of a spectrum represents the approximation of the spectrum itself, and the high frequency band stands for the detail of the spectrum. According to this principle, RFA descriptor analyses the low frequency band of local image gradients around keypoints by extracting and utilising the four coefficients as explained in the last step of RFA generation. Specifically, the extraction of the four coefficients acts like a low-pass filter convoluting with image gradients in spatial domain, but the extraction directly doing so in frequency domain. High frequency band is discarded because the details of local image gradients would be extremely dissimilar in the cases that input images are not exactly identical and the fact that images are smoothed by different Gaussian scales during the generation of RFA descriptor. Whereby, the descriptor extracts low frequency coefficients to describe the principle component in local image gradients.

Since the input of the Fourier transform in Eq. 7 is a complex function, the resulting Fourier coefficients do not obey any symmetrical properties as mentioned previously. To properly analyse low frequency band, for RFA descriptor, it is necessary to include the coefficients at both of the positive and negative frequencies. They represent for the same frequency response and contain the information at that frequency, but only in opposite direction or sign.

4. EXPERIMENT SETUP

This section describes the image database, matching techniques and evaluation metrics used to manifest the performances of RFA descriptor.

4.1 Image Database

The image database from a performance study of keypoint descriptors [14] was adopted to evaluate the performances of RFA descriptor with the keypoint orientation estimated by Fourier analysis. The image database includes a few sets of images that are individually affected by blur, illumination, JPG compression, rotation, and scale distortions. Thus, RFA descriptor's performances can be completely examined under these image distortions. Example images of this database are shown in Figure 3.

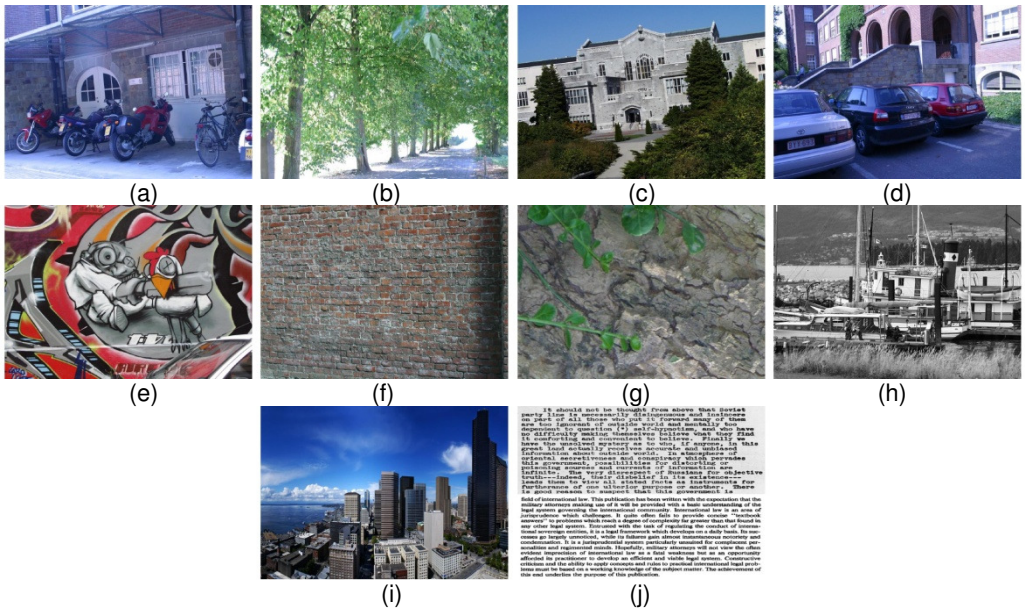


FIGURE 3: Example images of the database used for the evaluations of RFA descriptor. Blur changes: (a) Bikes and (b) Trees. JPG compressions: (c) UBC. Illumination changes: (d) Leuven. Viewpoint changes: (e) Graffiti and (f) Wall. Rotation and scale changes: (g) Bark and (h) Boat. Rotation changes: (i) Building and (j) Text.

4.2 Matching Technique

In order to obtain descriptor matches, it requires a similarity measure and a matching strategy to calculate the likeliness between descriptors and determine most likely descriptor matches among a pool of descriptors respectively. There are numerous similarity measures [24] [25] and matching strategies [26] [27] available for the matching of keypoint descriptors.

In this paper, however, the most familiar similarity measure and matching strategy are utilised to generate descriptor matches. They are Euclidean distance and threshold-based nearest neighbour (TNN) matching strategy respectively. As soon as the Euclidean distances between a descriptor and a pool of descriptors are calculated, TNN matching strategy would assess the descriptor pair that has the shortest distance between them. If the distance of that pair is below a pre-defined threshold, the pair is then recognised as a match, otherwise, no match is generated.

4.3 Evaluation Metrics

The RFA descriptor matches generated by Euclidean distance with TNN matching strategy are used to measure the performances of RFA descriptor. The matches are evaluated by recall-precision, which was initially employed in [28] to verify the performances of keypoint descriptors. It is the most popular performance metrics in the evaluation of keypoint descriptors because the major examinations on descriptor matches are their sensitivity (i.e., matching rate) and accuracy (i.e., matching accuracy). Recall represents the sensitivity of the generated descriptor matches in a given image pair that contains the same scene. It is defined as the ratio between the number of correct matches and the number of correspondences, such that,

$$recall = \frac{\#\ correct\ matches}{\#\ correspondences} \tag{9}$$

where, the correct matches and correspondences are verified by projecting keypoints from its image to the corresponding image. If the projected keypoints are equal as or close to their corresponding keypoints, then they are recognised as correct matches or correspondences. Another metrics in recall-precision is 1-precision that indicates the accuracy of generated matches and is the ratio between the number of false matches and the number of total matches. It is expressed as,

$$1 - precision = \frac{\#\ false\ matches}{\#\ correct\ \ matches + \#\ false\ matches} \tag{10}$$

5. EXPERIMENTAL RESULTS

This section presents the performance evaluations of RFA descriptor under various image distortions. Apart from the evaluations of RFA descriptor, the performances of other benchmark descriptors, such as SIFT, VLFeat SIFT and PCA-SIFT, are also provided for comparison purposes. Since PCA-SIFT requires a training process, it was trained by 26,000 image patches that are not related to the image database before it is used in the following evaluations. Twenty and 128 dimensional PCA-SIFT descriptors are illustrated in the evaluations for the completeness of the comparisons. The former is recommended by its author and the latter is to fairly compare with the other descriptors as they have dimensions of 128. Significant observations and discussions about the evaluations are also given in this section.

The descriptors were assessed while the images of Figure 3(a) and Figure 3(b) were blurred at different degrees. The performance results obtained by these blurred images demonstrate the abilities of the descriptors against blur changes, and they are presented in Figure 4. From the recall-precision graphs, it can be seen that RFA descriptor particularly outperforms the other keypoint descriptors in high precision region, between 0 and 0.5 of 1-precision. Since the smoothing effect caused by blurring occurs at high frequency band, the RFA descriptor is benefited from using Fourier transform and low-pass filter that remove the high frequency band of local image gradients around keypoints. Thus, the performances of RFA descriptor is hardly degraded by the image distortions induced by blur changes.

The quality of Figure 3(c) is degraded by JPG compression, which is based on discrete cosine transform to trade off image quality for a smaller file size. Figure 5 displays the performance results of the descriptors under this image quality degradation. As expected, RFA descriptor demonstrates a better performance than the other descriptors in this case since discrete cosine transform is a variant of Fourier transform. Both JPG compression and RFA descriptor process their input image in frequency domain and manipulate information with frequency decomposition. Whereby, the remaining information after JPG compression can be easily extracted and used with RFA descriptor.

The performance results from altering the illuminations of Figure 3(d) are depicted in Figure 6. Once more, it is observed that RFA descriptor surpasses the other descriptors in high precision region. The high recall at high precision region indicates the representations of image gradients by RFA descriptor are distinct and accurate under this type of image distortion. It is advantaged by the unit vector normalisation prior to Eq. 7 as the normalisation minimises the effect of illumination changes by locally standardising the magnitudes of horizontal and vertical gradients. Another contributor for this high performance is the linearity of Fourier transform that allows the scaling of an input signal to be factorised, then it is normalised by the unit vector normalisation in the last step of RFA descriptor generation.

Figure 7 shows the performances of the descriptors against the image distortions caused by viewpoint changes. As Figure 3(e) and Figure 3(f) were captured at different viewpoints, sophisticated geometrical transforms are imposed onto these images, increasing the possibility of generating incorrect matches. In Figure 7(a), the performance of RFA descriptor is extensively higher than the other descriptors in high precision regions. Although the other descriptors have higher recall in low precision region, it should be noted that their recall approximately starts to rise at the 1-precision of 0.7 and it can be said that the most of their matches are incorrect. With Figure 7(b), however, RFA descriptor only demonstrates an acceptable performance, which is in-between SIFT and VLFeat SIFT descriptors, as serious image deformations have been imposed in this dataset.

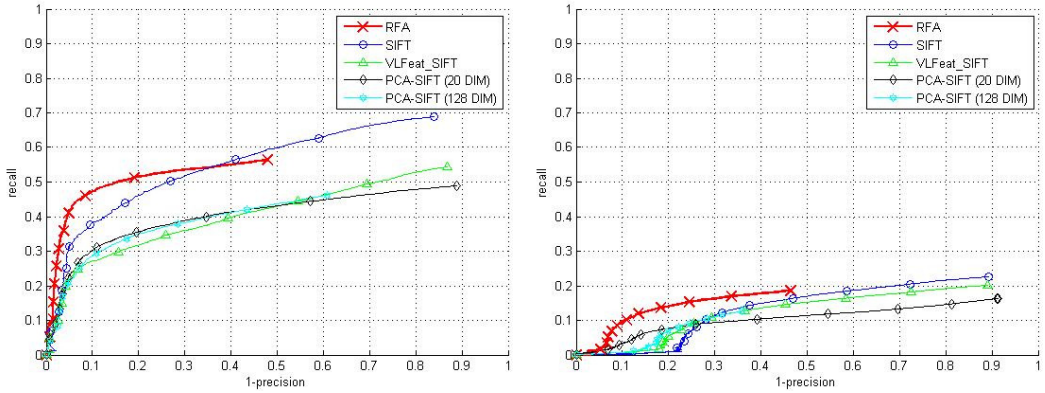


FIGURE 4: Recall-Precision evaluations for blur changes using: (a) Bikes and (b) Trees.

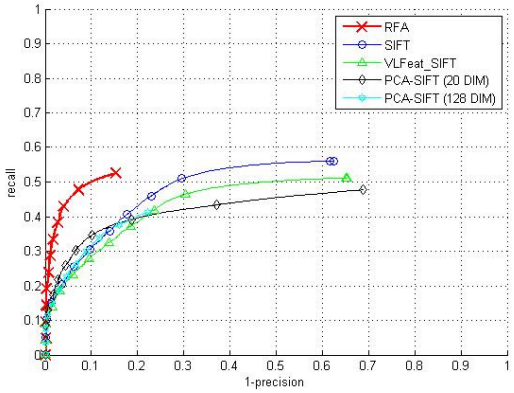


FIGURE 5: Recall-Precision evaluations for JPG compression using UBC.

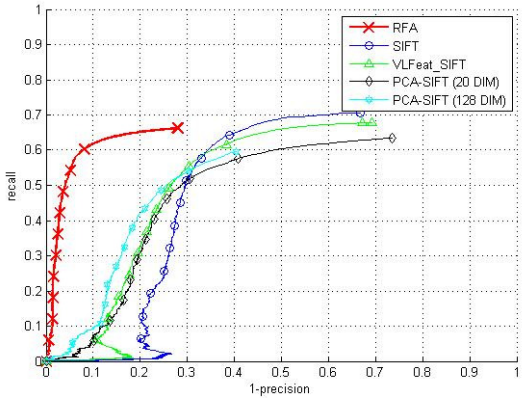


FIGURE 6: Recall-Precision evaluation for illumination changes using Leuven.

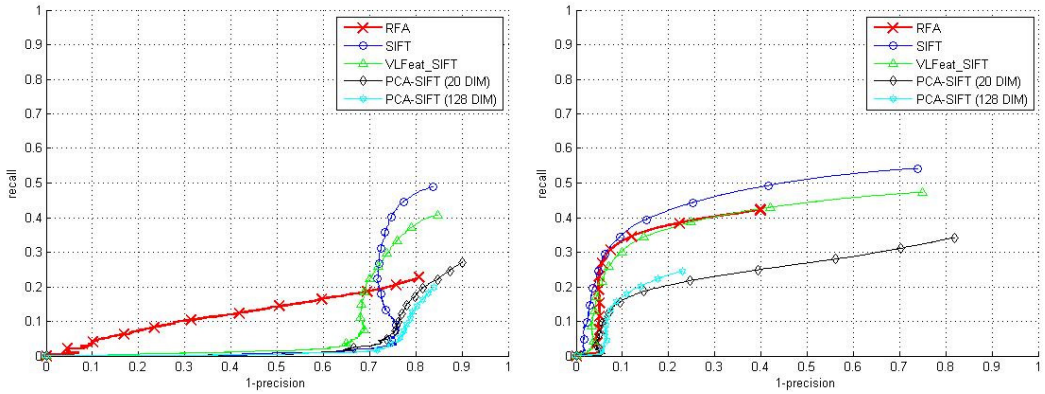


FIGURE 7: Recall-Precision evaluations for viewpoint changes using: (a) Graffiti and (b) Wall.

Figure 8(a) and Figure 8(b) present the performance results of the descriptors when the images of Figure 3(g) and Figure 3(h) simultaneously experienced the image distortions caused by rotation and scale changes. From Figure 8(a), RFA descriptor has achieved a performance that is much lower than SIFT and VLFeat SIFT descriptors. In Figure 8(b), RFA descriptor has a higher recall in high precision region when compared with the other descriptors. It is an acceptable performance, especially if accuracy is more important than sensitivity in keypoint matches.

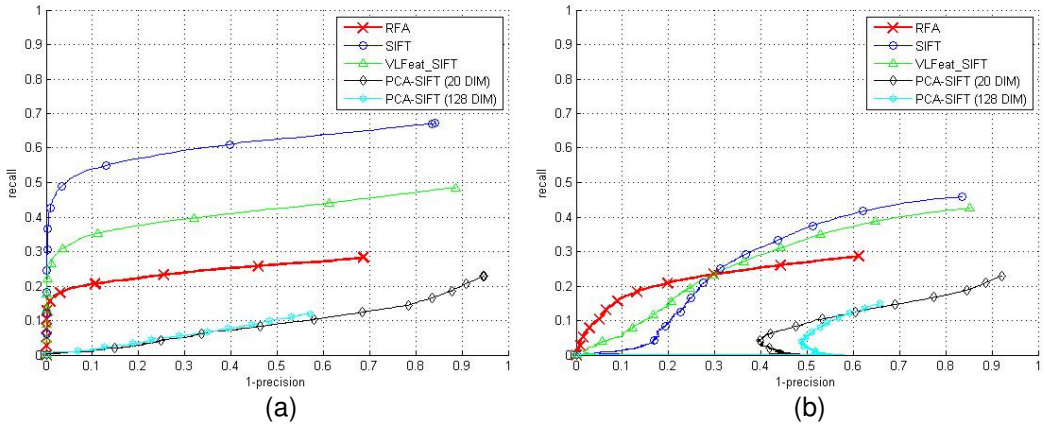


FIGURE 8: Recall-Precision evaluations for rotation and scale changes using: (a) Bark and (b) Boat.

Apart from the performance results given in Figure 7 and Figure 8, Figure 9 shows the performances of the descriptors when they were subjected under the influences of purely rotation changes. From both the recall-precision graphs, RFA descriptor demonstrates equivalent performances as VLFeat SIFT descriptor, but it outperforms SIFT and PCA-SIFT descriptors by significant margins. It is a solid evidence that our keypoint orientation estimate provides a reliable and robust reference orientation and allows the local image patches of descriptor matches to be rotated to a similar orientation. With the assistance of this keypoint orientation estimate, RFA descriptor has been able to easily deal with the image distortion induced by rotation changes.

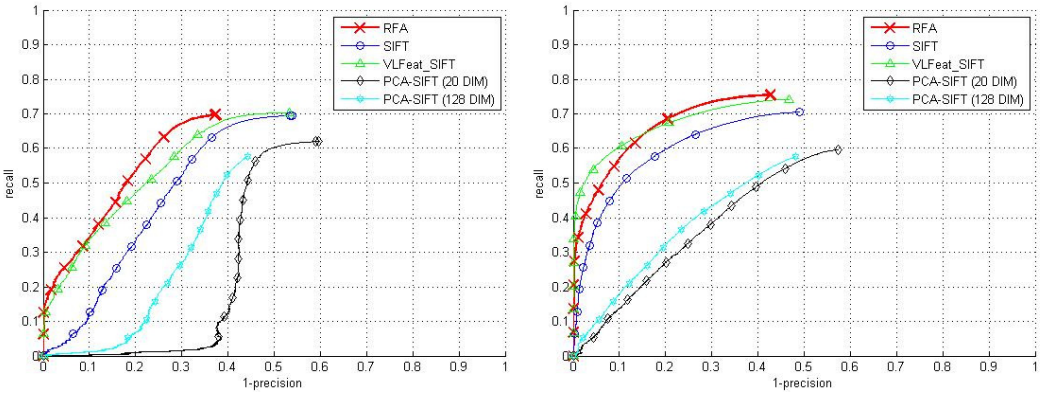


FIGURE 9: Recall-Precision evaluations for rotation changes using: (a) Building and (b) Text.

6. CONCLUSION

In this paper, a keypoint orientation estimate and a keypoint descriptor have been proposed to analyse the local image patch around keypoints in frequency domain. The keypoint orientation is estimated from the information of image intensities with the starting point normalisation of Fourier coefficients. For the keypoint descriptor, it extracts low order Fourier coefficients of image gradients to approximate overall appearance of the gradients. Meanwhile, image detail that resides in high order Fourier coefficients are ignored as excessive detail would make the descriptor superfluously distinctive.

When evaluating the performances of RFA descriptor, it was found that the descriptor outperforms benchmark descriptors, such as SIFT, PCA-SIFT and VLFeat SIFT descriptors, in images suffering from a variety of image distortions. RFA descriptor is particularly stable and reliable in dealing with the image distortions caused by blur, rotation, illumination and JPG compression changes. This is evidenced from the experimental results presented in this paper. In addition, RFA descriptor encodes image gradients into a distinctive and accurate representation as its matches always yield a recall that rapidly rises up in high precision region, indicating high sensitivity and accuracy during matching.

Through this paper, it has been proven that spectral analysis, such as Fourier transform, Fourier normalisation and low-pass filtering, can be employed to estimate keypoint orientations and to describe local keypoints. This research shows that spectral analysis serves well as an alternative or supplementary tool for statistical analysis of keypoint description and orientation estimation.

7. REFERENCES

[1] J. Li, N. Allinson, D. Tao, and X. Li, “Multitraining support vector machine for image retrieval,” *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3597–3601, 2006.

[2] J. Bauer, H. Bischof, A. Klaus, and K. Karner, “Robust and fully automated image registration using invariant features,” in *Proceedings of International Society for the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2004, pp. 1682–1777.

[3] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] G. Dorko and C. Schmid, “Selection of scale-invariant parts for object class recognition,” in *Proceedings of International Conference on Computer Vision*, 2003, pp. 634–639.

[5] S. Lazebnik, C. Schmid, and J. Ponce, “A sparse texture representation using affine-invariant regions,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. 319–324.

[6] M. Yang, L. Zhang, S. K. Shiu, and D. Zhang, “Monogenic binary coding: An efficient local feature extraction approach to face recognition,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1738–1751, 2012.

[7] Y. Fan and M. H. Meng, “3d reconstruction of the wce images by affine sift method,” in *Proceedings of World Congress on Intelligent Control and Automation*, 2011, pp. 943–947.

[8] S. Muramatsu, D. Chugo, S. Jia, and K. Takase, “Localization for indoor service robot by using local-features of image,” in *Proceedings of ICCAS-SICE*, 2009, pp. 3251–3254.

[9] A. Sedaghat, M. Mokhtarzade, and H. Ebadi, “Uniform robust scaleinvariant feature matching for optical remote sensing images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4516–4527, 2011.

[10] X. M. Mu, J. P. Liu, W. H. Gui, Z. H. Tang, C. H. Yang, and J. Q. Li, “Machine vision based flotation froth mobility analysis,” in *Proceedings of Chinese Control Conference*, 2010, pp. 3012–3017.

[11] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[12] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in *Proceedings of IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571.

[13] S. Taylor and T. Drummond, “Binary histogrammed intensity patches for efficient and robust matching,” *International Journal of Computer Vision*, vol. 94, no. 2, pp. 241–265, 2011.

[14] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[15] Y. Ke and R. Sukthankar, “Pca-sift: a more distinctive representation for local image descriptors,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 506–513.

[16] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (SURF),” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[17] S. Leutenegger, M. Chli, and R. Siegwart, “Brisk: Binary robust invariant scalable keypoints,” in *Proceedings of IEEE International Conference on Computer Vision*, 2011, pp. 2548–2555.

[18] J. Li and N. M. Allinson, “A comprehensive review of current local features for computer vision,” *Neurocomputing*, vol. 71, no. 10, pp. 1771–1787, 2008.

[19] L. Juan and O. Gwun, “A comparison of sift, pca-sift and surf,” *International Journal of Image Processing*, vol. 3, no. 4, pp. 143–152, 2009.

[20] J. L. Crowley and A. C. Parker, “A representation for shape based on peaks and ridges in the difference of low-pass transform,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 156–170, 1984.

[21] A. Folkers and H. Samet, “Content-based image retrieval using fourier descriptors on a logo database,” in *Proceedings of International Conference on Pattern Recognition*, vol. 3, 2002, pp. 521–524.

[22] S. Lin, C. Wong, T. Ren, and N. Kwok, “The impact of information volume on sift descriptor,” in *Proceedings of International Conference on Wavelet Analysis and Pattern Recognition*, 2013, pp. 287–293.

[23] A. K. Jain, *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.

[24] A. Strehl, J. Ghosh, and R. Mooney, “Impact of similarity measures on web-page clustering,” in *Proceedings of Workshop on Artificial Intelligence for Web Search*, 2000, pp. 58–64.

[25] A. Huang, “Similarity measures for text document clustering,” in *Proceedings of New Zealand Computer Science Research Student Conference*, 2008, pp. 49–56.

[26] O. Chum, J. Matas, and S. Obdrzalek, “Enhancing ransac by generalized model optimization,” in *Proceedings of Asian Conference on Computer Vision*, vol. 2, 2004, pp. 812–817.

[27] M. Cho and H. Park, “A robust keypoints matching strategy for sift: An application to face recognition,” in *Neural Information Processing*, ser. Lecture Notes in Computer Science, 2009, vol. 5863, pp. 716–723.

Comparison Between Levenberg-Marquardt And Scaled Conjugate Gradient Training Algorithms For Image Compression Using MLP

Devesh Batra *devesh.batra.in@ieee.org*
Member, IEEE

Abstract

The Internet paved way for information sharing all over the world decades ago and its popularity for distribution of data has spread like a wildfire ever since. Data in the form of images, sounds, animations and videos is gaining users' preference in comparison to plain text all across the globe. Despite unprecedented progress in the fields of data storage, computing speed and data transmission speed, the demands of available data and its size (due to the increase in both, quality and quantity) continue to overpower the supply of resources. One of the reasons for this may be how the uncompressed data is compressed in order to send it across the network. This paper compares the two most widely used training algorithms for multilayer perceptron (MLP) image compression – the Levenberg-Marquardt algorithm and the Scaled Conjugate Gradient algorithm. We test the performance of the two training algorithms by compressing the standard test image (Lena or Lenna) in terms of accuracy and speed. Based on our results, we conclude that both algorithms were comparable in terms of speed and accuracy. However, the Levenberg-Marquardt algorithm has shown slightly better performance in terms of accuracy (as found in the average training accuracy and mean squared error), whereas the Scaled Conjugate Gradient algorithm faired better in terms of speed (as found in the average training iteration) on a simple MLP structure (2 hidden layers).

Keywords: Image Compression, Artificial Neural Network, Multilayer Perceptron, Training, Levenberg-Marquardt, Scaled Conjugate Gradient, Complexity.

1. INTRODUCTION

Image Compression algorithms have received notable consideration in the past few years because of the growing multimedia content on the World Wide Web. Image Compression is a must since despite advances in computer and communication technologies, the digital images and videos are still demanding in terms of storage space and bandwidth [1].

In this paper, we present an evaluation of two popular training algorithms (Levenberg-Marquardt and Scaled Conjugate Gradient) for image compression using simple Multilayer Perceptron (MLP) classifier.

Various parameters such as the gradient, mu and validation checks are evaluated for both the algorithms to examine their performance in terms of accuracy and speed. Image Compression refers to the reduction of irrelevant and redundant image data in order to store and transfer data in an efficient manner. Image compression can be classified as lossy and lossless. Lossless image compression allows original image to be perfectly reconstructed from the image data without any loss [2]. It is generally used in medical imaging, technical drawings and other areas where the minute details of the images are required and data loss could be fatal. On the contrary, in lossy image compression, the images can be only partially reconstructed from the image data [3]. Even though some of the data is lost, this is usually advantageous because it gives improved compression rates and hence smaller sized images.

The paper is organized as follows: Some previous works on the Image Compression are presented in Section II. The theoretical background to the proposed approach is presented in Section III. The methodology of the experiment is presented in Section IV, followed by the results and discussions in Section V. Section VI presents the conclusions of the findings in this paper and finally Section VII present proposed future work in the field, followed by references in Section VIII.

2. RELATED WORK

There is a lot of research in literature that focuses on image compression using various classifiers and algorithms.

In [4] (2006), time taken for simulation has been reduced by 50% by estimating a Cumulative Distribution Function (CDF) and using it to map the image pixels.

In [5] (2013), a new approach for near–lossless compression of the medical images is proposed. Pre–processing techniques are applied to the input image to generate a visually quantized image. The visually quantized image is encoded using a low complexity block–based lossless differential pulse code modulation coder, followed by the Huffman entropy encoder. Results show the superiority of the proposed technique in terms of the bit rate and visual quality.

In [6], a comparison of Principal Component Analysis (PCA) is presented for still image compression and coding. The paper presents comparison about structures, learning algorithms and required computational efforts along with a discussion of advantages and drawbacks related to each technique. The wide comparison among eight principle component networks shows that cascade recursive least squares algorithm by Ci-chocki, Kasprzak and Skarbek exhibits the best numerical and structural properties.

[7] presents a comparison between Levenberg Marquardt (LM) and Scaled Conjugate Gradient (SCG) algorithms for Multilayer Perceptron diagnosis of Breast Cancer Tissues. The study concludes that both algorithms were comparable in terms of accuracy and speed. However, the LM algorithm showed better advantage in terms of accuracy and speed on the best MLP structure (with 10 hidden units).

[8] presents an overview of neural networks as signal processing tools for image compression model. The self-organizing feature map (SOFM) has been used in the design of codebooks for vector quantization (VQ). The resulting codebooks are shown to be less sensitive to initial conditions than the standard LBG algorithm.

3. THEORETICAL BACKGROUND

3.1 Artificial Neural Networks and MLP

ANNs can be defined in many ways. At one extreme, the answer could be that neural networks are simply a class of mathematical algorithms, since a network can be regarded essentially as a graphic notation for a large class of algorithms. Such algorithms produce solutions to a number of specific problems. At the other end, the reply may be that these are synthetic networks that emulate the biological neural networks found in living organisms [9].

Although computers outperform both biological and artificial neural systems for tasks based on precise and fast arithmetic operations, artificial neural systems represent the promising new generation of information processing networks. Neural networks can supplement the enormous processing power of the Von Neumann digital computer with the ability to make sensible decisions and to lean by ordinary experience, as we do [9].

The signal flow of neuron inputs, xi, is considered to be unidirectional as indicated by arrows, as is a neuron's output signal flow. This symbolic representation shows a set of weights and the neuron's processing unit, or node. The neuron output signal is given by the following relationship:

O = f (wtx),

where w is the weight vector defined as

W = [w1 w2 … wn] t

and x is the input vector:

X = [x1 x2 … xn] t

All vectors defined are column vectors; superscript t denotes a transposition. The function f (wtx) is referred to as an activation function. The activation functions of a neuron can be bipolar continuous or unipolar continuous as shown in figure 1 and 2 respectively.

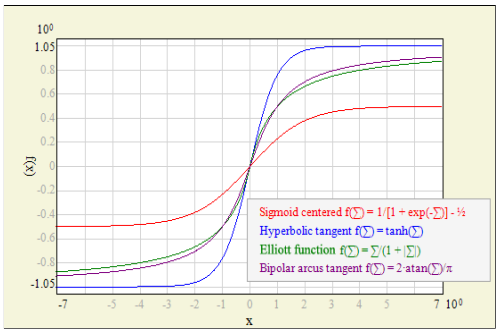


FIGURE 1: Bipolar Activation Function.

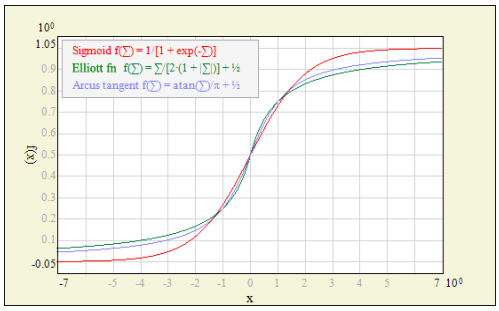


FIGURE 2: Unipolar Continuous activation function.

A feed-forward neural network is a biologically inspired classification algorithm. It consists of a (possibly large) number of simple neuron-like processing units, organized in layers. Every unit in a layer is connected with all the units in the previous layer. These connections are not all equal; each connection may have a different strength or weight. The weights on these connections encode the knowledge of a network. Often the units in a neural network are also called nodes.

Data enters at the inputs and passes through the network, layer by layer, until it arrives at the outputs. During normal operation, that is when it acts as a classifier, there is no feedback between layers. This is why they are called feed-forward neural networks [10].

Figure 3 is a 2-layered network with, from top to bottom: an output layer with 5 units, a hidden layer with 4 units, respectively. The network has 3 input units.

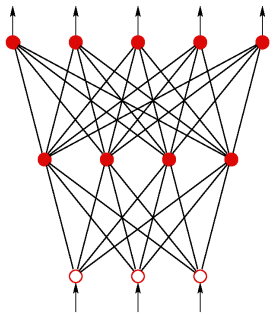


FIGURE 3: 2-Layered network.

Multi Layer perceptron (MLP) is a feed-forward neural network with one or more layers between input and output layer as shown in figure 4. This type of network is trained with the back-propagation learning algorithm. MLPs are widely used for pattern classification, recognition, prediction and approximation. Multi Layer Perceptron can solve problems, which are not linearly separable [11].

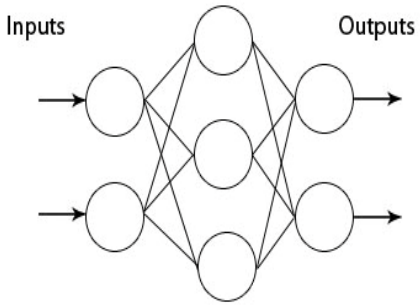


FIGURE 4: Multilayer Perceptron.

3.2 The Levenberg Marquardt Algorithm

Levenberg-Marquardt algorithm, which was independently developed by Kenneth Levenberg and Donald Marquardt, provides a numerical solution to the problem of minimizing a nonlinear function [12]. It is fast and has stable convergence. In the artificial neural network field this algorithm is suitable for small- and medium-sized problems.

Levenberg-Marquardt algorithm introduces an approximation to Hessian matrix; in order to ensure that the approximated Hessian matrix JTJ is invertible.

The approximation introduced is:

$$H = JtJ+uI$$

where, u is always positive, called combination coefficient and I is the identity matrix.
The elements on the main diagonal of the approximated Hessian matrix will be larger than zero. Therefore with this approximation, it can be sure that the matrix H is always invertible [13].
The update rule of Levenberg-Marquardt algorithm can be presented as:

$$W_{k+1} = W_k - (J_k^t J_k + uI)^{-1} J_k e_k$$

4. METHODOLOGY

The primary components of this work are training the multilayer perceptron for image compression and comparison of results obtained from the two training algorithms used. The multilayer perceptron training algorithms consist of Levenberg Marquardt and Scaled Conjugate Gradient algorithms. The results obtained are compared on the basis of various parameters such as speed (as observed in the average training iteration) and accuracy (as observed in terms of average training accuracy and mean squared error).

4.1 Image Dataset Description

We test the performance of the two training algorithms by compressing the standard test image, Lena (figure 5).

FIGURE 5: Standard Test Image: Lena.

The image properties are as follows:

Properties	Value
Pixel Dimensions	512 X 512 pixels
Print Size	5.33 X 5.33 inches
Resolution	96 X 96 DPI
Colour Space	RGB
File Size	768.1 KB
File Type	TIFF

TABLE 1: Image Properties.

4.2 Multilayer Perceptron and Structure

A multilayer feed-forward network is used. The most important characteristic of a multilayer feed-forward network is that it can learn a mapping of any complexity [9]. The network learning is based on repeated presentations of the training samples. The trained network often produces surprising results and generalizations in applications where explicit derivation of mappings and discovery of relationships is almost impossible. In the case of layered network training, the mapping error can be propagated into hidden layers so that the output error information passes backward. This mechanism of backward error transmission is used to modify the synaptic weights of internal and input layers.

Transfer function, is a process defining relationship between the input and the output. The transfer function of a neuron is chosen to have a number of properties, which either enhance or simplify the network containing the neuron. A non-linear function is necessary to gain the advantage of a multi-layer network.

4.3 Levenberg Marquardt and Scaled Conjugate Gradient Training Algorithm Parameters

The default values of various parameters used in MATLAB for Multilayer Perceptron training. The parameters and their default values used for Levenberg-Marquardt and Scaled Conjugate Gradient training algorithms are as follows:

Parameters	Value
Maximum Epochs	1000
Training Goal	0
Minimum Gradient	$1.00 \text{ X } 10^{-10}$
α	0.10
β	10

TABLE 2: Default Values of parameters used in MATLAB for MLP Training.

Detailed description of the training procedure used:

During the training procedure the input image dataset is encoded into a structure of hidden and output weight matrices. The image used for training purposes is assumed to be of dimension R X C and consists of r x c blocks. The following steps are followed during the training procedure:

1. The block matrix is converted into a matrix X of size P x N containing training vectors, where, x(n), is formed from image blocks. Mathematically, it can be expressed as follows:

   $$P= r.c \text{ and } p.N = R.C$$

2. The target data is made equal to the data, that is: D=X.
3. The network is then trained until the mean squared error, MSE, is sufficiently small. The matrices W^h and W^y are subsequently used in the image encoding and decoding steps.

IMAGE ENCODING:

The hidden-half of a neural network is used to encode images. The encoding procedure is described as follows:

$$F \rightarrow X, H = (W^h.X)$$

where X is the encoded image of F.

IMAGE DECODING:

The reconstruction of encoded image is known as decoding. It is done using the output half of the neural network. The decoding procedure is as follows:

$$Y = (W^y.H), Y \rightarrow F$$

ALGORITHM:

STEP 1: Input the image to be tested
STEP 2: The input image is divided into block of pixels
STEP 3: Each block is scanned for complexity level
STEP 4: The neurons are initialized
STEP 5: Scanned vectors are applied to each neuron on the input layer
STEP 6: Operations are performed depending upon the weights assigned and logic involved (TRANSIG)
STEP 7: They are then passed to the hidden layer
STEP 8: Repeat STEP6 (PURELIN)
STEP 9: The outputs are reassembled
STEP 10: The neural network is trained and the weights are retained.

5. RESULTS AND DISCUSSIONS

The results yielded on the comparison of Levenberg Marquardt Algorithm and Scaled Conjugate Algorithm for image compression will be discussed in this section. The conditions under which the comparison was done have already been discussed in Section IV.

The image obtained on compression of image 'Lena' (figure 5) with both the algorithms was of same quality and has been shown in figure 6.

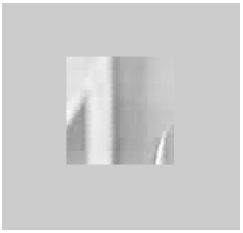


FIGURE 6: Compressed Image.

As shown in table III, Levenberg Marquardt algorithm took 53 seconds for compressing the image and running a cycle of 1000 epochs whereas Scaled Conjugate Gradient algorithm took mere 11 seconds for the same. Hence, the Levenberg Marquardt was relatively slow in processing the image in comparison to the Scaled Conjugate Gradient.

Levenberg Marquardt	Scaled Conjugate Gradient
53 seconds	11 seconds

TABLE 3: Time taken by both Algorithms.

A test was also conducted for the usage of RAM and CPU while both the algorithms were processing the image. Both the algorithms used almost equal amount of RAM and CPU while executing as shown in table IV. The CPU used for performing the experiment is Intel Core i5 – 2430M CPU@2.4 GHz (64-bit). The RAM used is 4 GB.

As indicated in the table, ideal state is a state of the system in which the execution is not being performed, i.e., Windows Task Manager is the only program running and no applications are running in the background. As for the processes and services, only basic Operating System processes (Windows 7 Professional) like Task Manager, Windows AutoUpdater and Windows Explorer.

State	Levenberg Marquardt		Scaled Conjugate Gradient	
	CPU	RAM	CPU	RAM
Ideal State	05%	43%	02%	47%
At t = 0	17%	51%	16%	52%
During Execution	58-67%	48%	55%	47%

TABLE 4: Usage of RAM AND CPU during execution of both Algorithms.

As shown in the comparison graphs of Mean Squared Error (MSE) of both Levenberg Marquardt and Scaled Conjugate Gradient (figure 7 and 8 respectively), the best results were obtained at different epochs. In both the cases it can be observed that the MSE stabilizes after certain number of epochs.

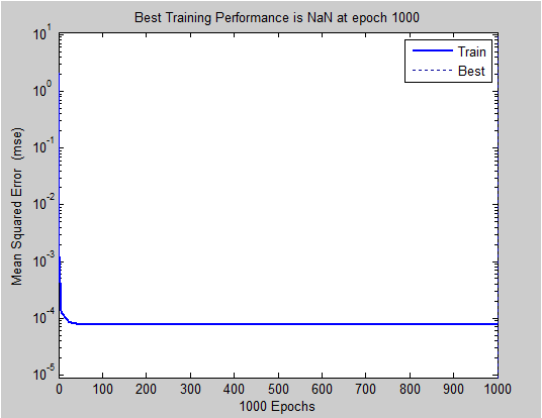


FIGURE 7: Mean Squared Error for Levenberg Marquardt.

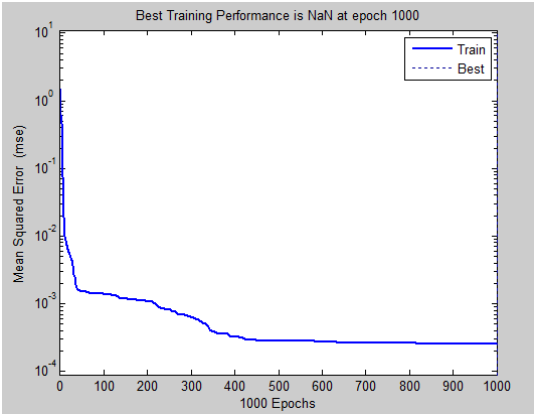


FIGURE 8: Mean Squared Error for Scaled Conjugate Gradient.

As shown in the comparison graphs of gradients of both Levenberg Marquardt and Scaled Conjugate Gradient (figure 9 and 10 respectively), it can be observed that the performance function at each epoch is different for both the cases. As the gradient becomes smaller and closer to zero, the function will be minimized. This implies that the outputs are very close to the targets and hence the network is trained.

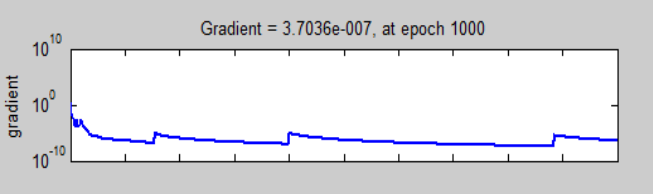


FIGURE 9: Gradient for Levenberg Marquardt.

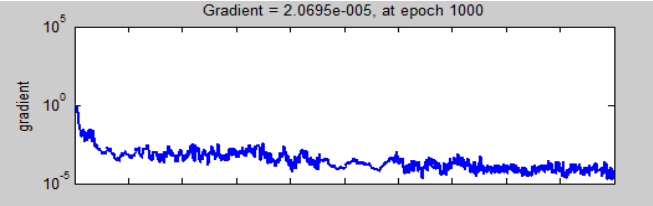


FIGURE 10: Gradient for Scaled Conjugate Gradient.

6. CONCLUSION

In this paper, we have compared the two most widely used training algorithms for multilayer perceptron (MLP) image compression - the Levenberg-Marquardt and the Scaled Conjugate Gradient algorithm. The performances of these two algorithms were tested by compressing the standard test image (Lena or Lenna) in terms of accuracy and speed. Based on our results, it was observed that both the algorithms were comparable in terms of speed and accuracy. However on the basis of Mean Squared Error (MSE) vs. epochs graph it was observed that the Levenberg-Marquardt had better accuracy as the MSE stabilized earlier in case of Levenberg-Marquardt algorithm as compared to that in the case of Scaled Conjugate Gradient algorithm. On the other hand, the Scaled Conjugate Gradient algorithm faired better in terms of speed (as found in average training iteration) on a simple MLP structure (2 hidden layers).

The paper provides results that are of utmost importance to the industry since the said comparison helps the Computer Scientists in analysing the difference between the two algorithms in minute details. Hence, they can judge, based on the comparison shown in the paper, which algorithm they want to use in transmitting images over the network. If they want the images sent over the network to be reliable, without any due consideration to time, then this paper suggests them to choose Levenberg-Marquardt algorithm over the Scaled-Conjugate algorithm. Scientists involved in complex research involving image analysis, who need the accuracy of the image to be extremely high, would generally encounter this type of a scenario. However, if they want the compression of the images to be fast, such as image sharing applications and services for general public, they can easily opt for the Scaled Conjugate algorithm.

7. FUTURE RESEARCH

Now that we have successfully compared the two most widely used training algorithms for multilayer perceptron (MLP) image, the practical implementation of these two algorithms as per the need can be done easily. Post this analysis; Levenberg-Marquardt algorithm is now ready to be used for reliable and high quality transportation of images over the networks with high bandwidths, especially in the scenarios where the focus is on transfer of more reliable images rather than the speed with which the images need to be compressed. On the other hand, the Scaled Conjugate algorithm can be used for a comparatively less accurate but faster transmission of the said images.

With this, we understand that there is a future for the application and comparison of these algorithms on animations and videos – entities that are combination of images. The tricky part in the comparison of these algorithms would be that videos and animations are composed of various other elements apart from images, such as text and sound, and similarly, their transfer

over the networks is dependent on various other parameters like “frames per second” in a video, communication technique used in the networks, etc. Thus, if due consideration is given to all the elements, a reliable comparison of the modified algorithms can be obtained.

8. REFERENCES

[1] Ebrahimi, Farzad, Matthieu Chamik, and Stefan Winkler. "JPEG vs. JPEG 2000: an objective comparison of image encoding quality." Optical Science and Technology, the SPIE 49th Annual Meeting. International Society for Optics and Photonics, 2004.

[2] What is lossless image compression. Available:http://dvd-hq.info/data_compression_1.php.

[3] What is lossy image compression. Available: http://dvd-hq.info/data_compression_2.php.

[4] Durai, S. Anna, and E. Anna Saro. "Image Compression with Back-Propagation Neural Network using Cumulative Distribution Function." International Journal of Applied Science, Engineering & Technology 3.4 (2007).

[5] Cyriac, Marykutty, and C. Chellamuthu. "A near–lossless approach for medical image compression using visual quantisation and block–based DPCM."International Journal of Biomedical Engineering and Technology 13.1 (2013): 17-29.

[6] Costa, Saverio, and Simone Fiori. "Image compression using principal component neural networks." Image and vision computing 19.9 (2001): 649-668.

[7] Mohamad, N., et al. "Comparison between Levenberg-Marquardt and scaled conjugate gradient training algorithms for breast cancer diagnosis using MLP."Signal Processing and Its Applications (CSPA), 2010 6th International Colloquium on. IEEE, 2010.

[8] Dony, Robert D., and Simon Haykin. "Neural network approaches to image compression." *Proceedings of the IEEE* 83.2 (1995): 288-303.

[9] Zurada, Jacek M. "Introduction to artificial neural systems." (1992).

[10] What is feedforward neural network. Available: http://www.fon.hum.uva.nl/praat/manual/Feedforward_neural_networks_1__What_is_a_feedforward_ne.html

[11] Multilayer Perceptron. Available: http://neuroph.sourceforge.net/tutorials/MultiLayerPerceptron.html

[12] Yu, Hao, and B. M. Wilamowski. "Levenberg-marquardt training." The Industrial Electronics Handbook 5 (2011): 1-15.

[13] Møller, Martin Fodslette. "A scaled conjugate gradient algorithm for fast supervised learning." Neural networks 6.4 (1993): 525-533.

[14] MATLAB Product Help. Available: http://www.mathworks.in/help/matlab/.

[15] Wei, Wei-Yi. "An Introduction to Image Compression." National Taiwan University, Taipei, Taiwan (2009): 1.

[16] Anderson, Dave, and George McNeill. "Artificial neural networks technology."*Kaman Sciences Corporation* 258 (1992): 13502-462.

[17] Nguyen, Derrick, and Bernard Widrow. "Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights." *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*. IEEE, 1990.

[18] Ranganathan, Ananth. "The levenberg-marquardt algorithm." *Tutoral on LM Algorithm* (2004): 1-5.

[19] Andrei, Neculai. "Scaled conjugate gradient algorithms for unconstrained optimization." *Computational Optimization and Applications* 38.3 (2007): 401-416.

[20] Steihaug, Trond. "The conjugate gradient method and trust regions in large scale optimization." *SIAM Journal on Numerical Analysis* 20.3 (1983): 626-637.

[21] Riedmiller, Martin. "Advanced supervised learning in multi-layer perceptrons—from backpropagation to adaptive learning algorithms." *Computer Standards & Interfaces* 16.3 (1994): 265-278.

Secured Reversible Data Hiding In Encrypted Images Using Hyper Chaos

T.M. Amarunnishad *amarnishad@rediffmail.com*
Professor
T K M College of Engineering
Kollam, Kerala, India

Aslam Nazeer *aslamnaz06@gmail.com*
Research Scholar
T K M College of Engineering
Kollam, Kerala, India

Abstract

Recently, more and more attention is paid to reversible data hiding (RDH) in encrypted images, since it maintains the excellent property that the original cover can be losslessly recovered after embedded data is extracted while protecting the image content's confidentiality. All previous methods embed data by reversibly vacating room from the encrypted images, which may be subject to some errors on data extraction and/or image restoration. Here a novel method is proposed by reserving room before encryption with a traditional RDH algorithm, and thus it is easy for the data hider to reversibly embed data in the encrypted image. Moreover data to be embedded is shuffled using a hyper chaotic function which is difficult to be extracted from the stego image without original key. A digital water mark is also embedded which ensures integrity of the data. The proposed method has been validated against three other available RDH schemes and it is observed that the proposed scheme outperforms these RDH schemes both in visual quality and payload. The proposed method can achieve real reversibility, that is, data extraction and image recovery are free of any error.

Keywords : Reversible Data Hiding, Encrypted Image, Self-reversible Embedding, Digital Watermarks, Hyper Chaotic System, Peak Signal To Noise Ratio.

1. INTRODUCTION

Data transfer through internet is common now a days. Since internet is a public network, confidential data have to be secured during transmission. Steganography [1] is used in such cases. It is the process of hiding secret data into a carrier in order to convey secret messages confidentially. The carrier may be audio, video or digital images. Due to availability and convenience digital images are widely used as carriers. The original image is known as cover image and the image up on which data embedded is known as stego image. Due to data embedding some distortion occurs in the stego image and these distortions are known as embedding distortions. A good embedding algorithm produces only less embedding distortions.

There are two types of data hiding, Reversible and Non-reversible. In reversible data hiding (RDH), the original image is losslessly recovered after extracting the embedded data where as in non-reversible data hiding once the image is distorted it cannot be reconstructed back. RDH is used where the image as well as the data is equally important. This technique is widely used in medical imaging, military purpose etc.

There are many RDH techniques available now a days based on lossless compression like histogram modification [2], difference expansion (DE) [3] etc. Among these, histogram based techniques are attracted much. Histogram based methods modify the histogram so that secret data can be embedded in to the modified histogram. The first histogram based method is proposed by Ni et al [4] in which data is embedded in to the image based on zero/peak pixel value. This method is simple and execution time is short. The stego image quality is also high but embedding capacity (EC) is low and the algorithm does not work if the image is having a flat histogram. Moreover it has overflow or underflow problem.

Tian et al [5] proposed another RDH method based on DE. In this method the neighbour pixel value differences are calculated and some differences are selected for DE. The payload is embedded in the difference number. It explores the redundancy in the image. Zhang et al [6] used histogram flipping method to embed data by which the encrypted image is divided in to several blocks. The three LSB's of half of pixels of each block are flipped to embed the extra bit. Zhang et al [7] proposed another method in which space to embed data are created before encryption and data is embedded into those specified areas using LSB replacement mechanism. Zhang [8] proposed a separable reversible data hiding scheme in encrypted images. In this scheme, at first the content owner encrypts the original uncompressed image using an encryption key. Then, a data-hider may compress the least significant bits of the encrypted image using a data-hiding key to create a sparse space to accommodate some additional data. Thodi et al [9] proposed another technique in which a combination of histogram shifting as well as DE technique is used. Hu et al [10] proposed another technique based on DE. Jung et al [11] proposed another technique which uses interpolation method.

The objective of the proposed method is to develop an RDH scheme with increased security. In order to increase the security a chaotic function is introduced in the proposed technique. A digital watermark is also introduced in order to identify transmission error or any explicit modification done by any third party.

The rest of this paper is organized as follows. The proposed data hiding scheme is explained in section 2, experimental and theoretical analysis is presented in section 3, and finally conclusions in section 4.

2. PROPOSED METHOD

In the existing RDH techniques "vacating room after encryption (VRAE)" [12] is used in which the original image is encrypted using a standard cipher and an encryption key. Then the data hider embeds data into encrypted image using some RDH technique and the encrypted image is send to the receiver. The receiver or third party having encryption key can extract the image from encrypted image and those who possess data hiding key can extract the embedded data in the image. It means those who have both keys can extract both image and data. This method is illustrated in figure 1(a).

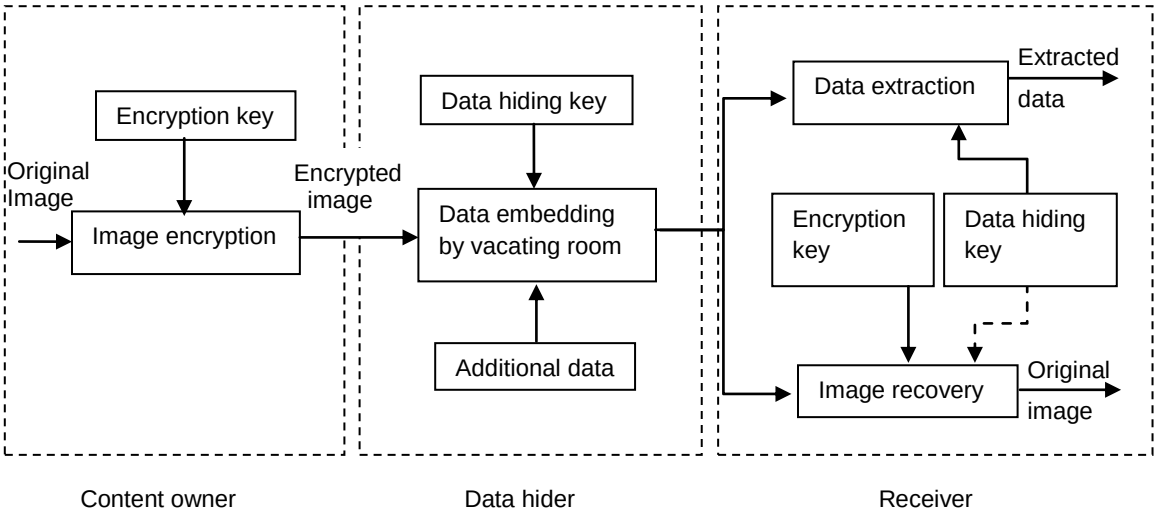
In the proposed method the order of creating space for data embedding and encryption is reversed. i.e space to embed data is reserved prior to image encryption and using some RDH technique data is embedded into these specified areas. This method is known as "reserving room before encryption (RRBE)" [7] scheme which is illustrated in figure 1(b).

In RRBE the content owner first reserves space in the original image and it is converted to encrypted form using an encryption key. The data hider needs only embedding of data on those reserved areas of the image. RRBE scheme primarily consists of four stages: generation of encrypted image, data hiding, image recovery and image extraction. The proposed RRBE technique is explained in the following subsections.

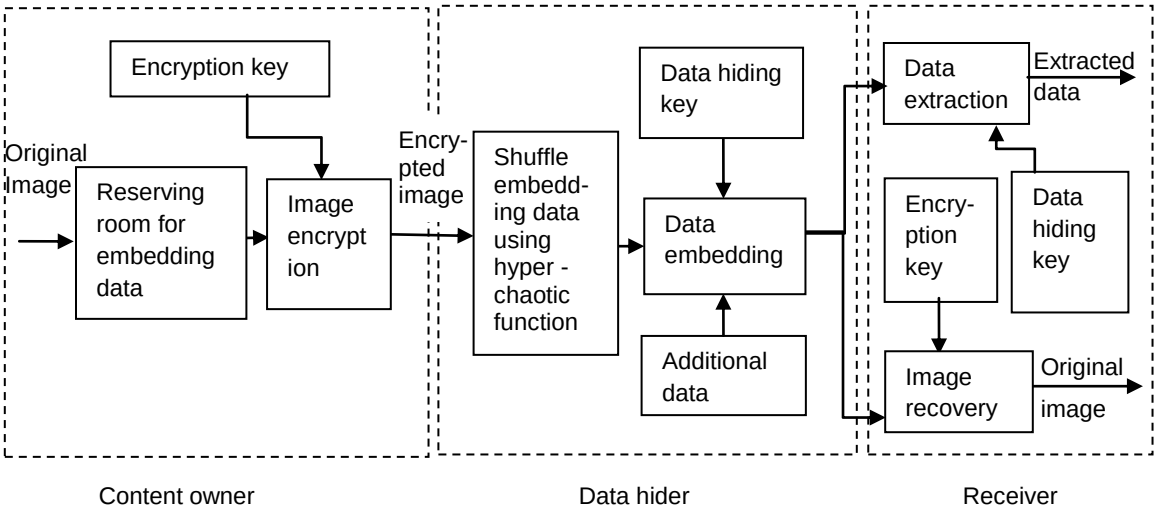
2.1 Generation of Encrypted Image

In order to produce an encrypted image using RRBE method, three steps are required: image partition, self reversible embedding and image encryption.

1) Image Partition: The goal of this step is to divide the image into two parts A and B using a smoothness function so that a smoother area B is constructed on which standard RDH algorithm [13], [14] can achieve better performance. Consider an original image I of size M x N and pixels $B_{i,j} \in [0,255]$, $1 \leq i \leq M$, $1 \leq j \leq N$. At first the size of to be embedded message is calculated and denoted as l. The original image is divided into several overlapping blocks along the rows, whose number is determined by l. Each block having m blocks, where m=$\lceil l/N \rceil$ and the number of blocks can be calculated by n= M-m+1.



(a)



(b)

FIGURE 1: Framework of (a) Vacating Room After Encryption and (b) Reserving Room Before Encryption.

Here each block is overlapped by previous or sub sequential block. A function is defined to measure the smoothness of each block

$$f = \sum_{u=2}^{m} \sum_{v=2}^{N-1} | C_{u,v} - \frac{C_{u-1,v} + C_{u+1,v} + C_{u,v-1} + C_{u,v+1}}{4} | \quad (1)$$

As the value of f increases, it means that those blocks contain complex structures. So the block with higher f is considered as A and puts in front of the image which is concatenated by rest part which is considered as B with fewer textured area, shown in figure 2.

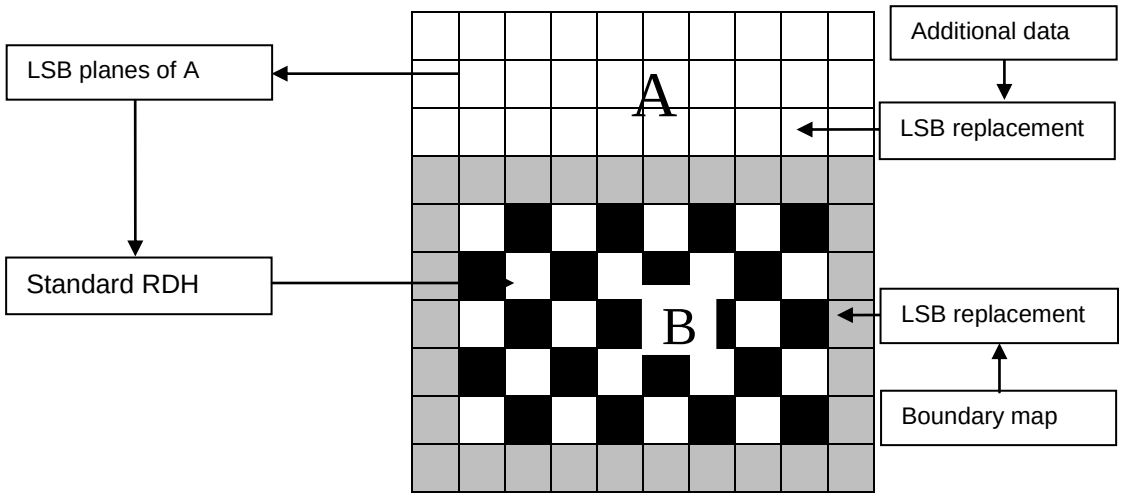


FIGURE 2: Illustration of image partition and embedding process.

2) Self-Reversible Embedding: The goal of this step is to embed the LSB planes of A into B using any RDH algorithm. Interpolation method is a commonly used RDH algorithm which is used here. Pixels in the area B are divided in to 2 sets: white and black. White pixels are those whose indices satisfy (i + j) mod 2 = 0 and black pixels are (i +j) mod 2 = 1. Then the value of each white pixel is estimated by interpolation value obtained with four black pixels surrounding it.

$$B'_{i,j} = w_1 B_{i-1,j} + w_2 B_{i+1,j} + w_3 B_{i,j-1} + w_4 B_{i,j+1} \quad (2)$$

where the weight w_i ,$1 \le i \le 4$, is determined by the same method proposed in [13]. The estimating error is calculated by $e_{i,j} = B_{i,j} - B'_{i,j}$ and then data are embedded into estimating error sequence using histogram shift. After calculating all values of white pixels the estimating error of black pixels are calculated by using modified white pixels and data are embedded into estimated error sequence of white pixels. Multilayer embedding is used if more data have to be embedded. For every single layered embedding two estimating error sequence are required for embedding messages. By using bidirectional histogram shift, the messages are embedded in the error sequence, ie. the estimated error histogram is divided into 2 parts, right and left. The highest point in each part is denoted as RM and LM and the zero point in each part is denoted as RN and LN. For ideal images RM = 0 and LN = -1. To embed messages into RM all values between RM+1 and RN-1 are shifted towards right by one step. The embedding process in the left part is similar except the shifting direction is left.

In this method the overflow/underflow problem is eliminated by embedding data only on those pixels whose values are between 1 and 254. Another problem arises when non-boundary pixels such as 1 is changed to 0 or 254 to 255.These newly created boundary pixels are known as pseudo boundary pixels. So a boundary map is maintained inorder to identify whether the boundary pixels are pseudo or natural. A binary sequence bit "0" is used to denote natural boundary pixel and "1" for pseudo boundary pixel. The marginal area is selected to embed boundary map. The parameters such as RN, LN, RM, LM, payload, start row, end row of A in original image are embedded into marginal area.

3) Image Encryption: After rearranging the self-embedded image it is encrypted by using any encryption algorithm. The encrypted image is denoted as X. After encryption a third party cannot access the image without encryption key, thus the privacy of the content is maintained.

2.2 Data Hiding in Encrypted Image

After encryption the encrypted image X is sent to the data hider. The data hider does not have any access to the real image. In the encrypted image, the region up on which data to be embedded are already identified and taken into front which is denoted as A_E .The data to embedded are shuffled randomly by using hyperchaotic function and embedded in A_E. The hyper chaotic function makes use of some keys which are known as data hiding key. One can't extract the data without data hiding key.

2.2.1 Chaotic functions

Chaos based encryption is first proposed in 1989 [15], [16] and after that many research works were appended in literature. The importance of chaotic function is "nearby" input does not generate "nearby" output. Recently, hyper chaos is widely used for encryption because it has more complex dynamical characteristics than chaos. In the proposed method hyper chaos is used to increase the security of the embedded data. The following hyper chaotic system is selected and it is used for generating random sequences

$$x_{n+1} = a_1 \times x_n + a_2 \times y_n \; ; \qquad y_{n+1} = b_1 + b_2 \times x^2{}_n + b_3 \times y_n \; ; \tag{3}$$

Here a_1 , x_n , a_2, y_n , b_1, b_2, b_3 can hold random values and which will be decided by user and that can be considered as encryption key and data hiding key. Here we set a_1=-0.95, a_2=-1.3, b_1=-0.45, b_2=2.4, b_3=1.05, x_n=0.0391 and y_n=0.019

2.2.2 Digital Watermarking

A digital watermark [17] is a kind of marker which is embedded in to the cover. It is typically used to identify ownership of the copyright. "Watermarking" is the process of hiding digital information in a carrier signal. Digital watermarks may be used to verify the authenticity or integrity of the carrier signal or to show the identity of its owners. The digital watermark should not alter carrier signal, it just marks data, but does not degrade it nor controls access to the data. A small digital watermarking is embedded into the marginal pixels of the cover image so that if any transmission error or any explicit modification done by a third party can be easily identified.

2.3 Data Extraction and Image Recovery

Data extraction is the reverse of data embedding and image decryption is the reverse of image encryption. If the receiver having both keys i.e. encryption key as well as data hiding key he can decrypt the image as well as extract the data. If he has only encryption key he can only decrypts the image, he can't extract data. If he having only data hiding key he can extract the data, but cannot decrypt the image.

3. EXPERIMENTS AND RESULTS

In this section, the simulation results and testing of performance of the proposed scheme by the key space and key sensitivity analyses are presented. All the experiments have been performed on a personal computer with a 2.4 GHz Intel Core2 i3 processor, 2G memory and 250 GB hard disk with a Windows 7 operating system. The proposed method has been tested on standard

publically available images such as Lena, Airplane, Barbara, Peppers and Boat and each image is of size 512 x 512

3.1 Key space analysis

An algorithm's key space refers to the set of all possible keys that can be used to generate a key, and is one of the most important attributes that determines the strength of a cryptosystem. In encryption algorithms the most effective attack is the Brute Force attack, where the enemy performs a complete search through all possible keys of the key space to find the right one. To prevent an adversary from using a brute-force attack to find the key used to encrypt a message, the key space is usually designed to be large enough to make such a search infeasible. On average, half the key space must be searched to find the solution. Another desirable attribute is that the key must be selected truly randomly from all possible key permutations. If keys were not randomly selected the attacker is able to determine some factor that may influence how the key was selected, so that the search space can be significantly reduced. Humans do not select passwords randomly; therefore attackers frequently try a dictionary attack before a brute force attack, as this approach can often produce the correct answer in far less time than a systematic brute force search of all possible character combinations. If a key were eight bits (one byte) long, the key space would consist of 2^8 or 256 possible keys.

Key space analysis handles maximum number of keys that can be used to find out the original key. Key length should be large to avoid brute force attack. In our proposed system we use a hyper chaotic function to determine the data hiding key. The key can be represented as (A, X1). A= (a_2, a_4, b_1, b_2, b_3) X1=(x_0, y_0). These keys are used for shuffling the pixel positions of the image. If we are using 32 bit number for generation of A and X1, then the key space of data hiding key is equal to the $2^{32} \times 2^{32} \times 2^{32} \times 2^{32} \times 2^{32} \times 2^{32} \times 2^{32} = 2^{224}$. Because of this large key space brute force attack is difficult.

3.2 Key sensitivity analysis

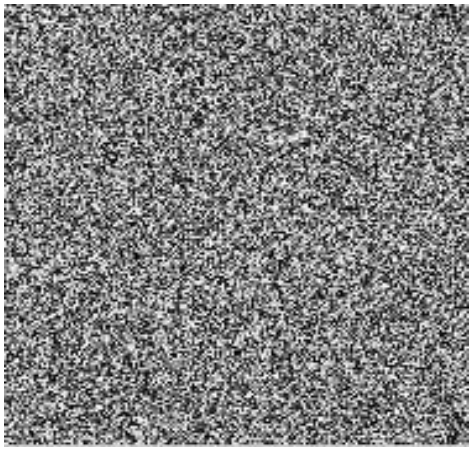
This subsection specifies sensitivity of data hiding key in proposed method. That means what changes will occur if a slight change is made in data hiding key. First, we hide some data into an image with key a_1= -0.95; a_2=-1.3; b1=-0.45; b2=2.4; b3=1.05; xn=0.0291; y_n=0.019; And then make a slight change with x_n=0.0391; y_n=0.019; So new key for data hiding is a_1= -0.95; a_2= -1.3; b_1= -0.45; b_2=2.4; b_3=1.05; x_n=0.0391; y_n=0.019. The recovery of data using incorrect key and correct key are shown in Table 1. When data is extracted using incorrect keys the extracted secret data is wrong. i. e only correct keys can extract original data. From this we can conclude that data hiding is sensitive to data hiding keys.

Figure 3 shows real time example of original image, encrypted image,decrypted image with data embedded in it and fully recovered image. The difference between original and extracted image is calculated. If the difference is zero it means that the image is losslessly recovered. Here the difference is zero which means the original image is losslessly recovered from stego image.

Data embedded using original key	Data extracted using original key	Data extracted using wrong key
"Hi…How are you"	"Hi…How are you"	"Ê¨ááGê¯cØi~"
"Who are you?"	"Who are you?"	"'ÉQwkJÛ/'"
"Hello world"	"Hello world"	"5AÇ • v/¾ • è"
"India is our country'	"India is our country"	"Ð²I¯È4Ã'Ì¹âQþ±"

TABLE 1: Key sensitivity analysis using original and wrong keys.

(a) Original Image



(b) Encrypted Image

(c) Decrypted Image

(d) Recovered image

FIGURE 3: Images after different steps of proposed RDH.

3.3 Implementation Issues

The peak signal-to-noise ratio (PSNR) is the objective criteria to find out the quality of the images after decryption. PSNR is the ratio between a signal's maximum power and the power of the signal's noise. Each picture element (pixel) may get changed when an image is modified. Logically, a higher value of PSNR is good because it means that the signal to noise ratio is higher. Signals can have a wide dynamic range, so PSNR is usually expressed in decibels, which is a logarithmic scale. PSNR values of the test images are recorded and evaluated inorder to check the quality of images and to check the efficiency of the proposed algorithm. To achieve high PSNR the following measures have to be taken.

3.3.1 Choice of LSB Plane Number

According to the proposed algorithm at first the image is divided into two parts A and B. The size of A is determined by the size of the message to be embedded and also by the number of LSB planes reversibly embedded in B. The choice of multiple LSB planes increases the size of B with an increase in embedding capacity. Table 2 shows the PSNR comparison between three different choices of LSB planes for five test images under various embedding rates measured by bits per pixel (bpp). From table 2 we can find that single LSB plane is better at a low embedding rate of

less than 0.2 bpp. For an embedding rate of 0.2 bpp and beyond, the choice of multiple LSB planes flips between 2 and 3 for a longer PSNR value.

PSNR results									
Embedding rate (bpp)		**0.005**	**0.01**	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Lena	1 LSB-plane	67.31	63.80	56.45	52.34	49.07	45.00	40.65	35.96
	2 LSB-plane	66.78	62.65	55.14	51.76	48.99	45.10	42.56	38.74
	3 LSB-plane	64.62	60.94	53.30	50.06	46.98	43.98	43.98	39.06
Airplane	1 LSB-plane	65.79	62.90	57.40	54.05	49.90	48.26	44.87	40.72
	2 LSB-plane	65.31	61.12	56.33	53.79	47.68	48.10	45.45	42.12
	3 LSB-plane	63.20	59.81	53.90	50.44	47.10	45.79	43.12	42.05
Barbara	1 LSB-plane	63.78	62.72	55.56	51.46	45.68	43.56	39.24	34.00
	2 LSB-plane	62.71	62.01	54.72	50.71	40.51	43.70	40.78	36.45
	3 LSB-plane	60.13	61.10	53.16	49.68	46.00	42.81	40.34	37.06
Peppers	1 LSB-plane	63.48	61.82	54.17	51.02	46.00	42.08	36.16	____
	2 LSB-plane	63.17	61.03	53.50	50.50	46.16	42.65	39.47	34.57
	3 LSB-plane	62.14	60.33	52.22	49.18	45.43	42.10	39.40	36.34
Boat	1 LSB-plane	66.55	64.42	56.75	52.71	49.10	45.21	41.44	36.60
	2 LSB-plane	65.91	63.34	55.75	51.02	48.40	44.98	42.26	40.87
	3 LSB-plane	64.17	61.81	53.73	50.45	46.71	43.81	41.75	39.44

Table 2: PSNR comparison for three different LSB-plane choices under various embedding rates.

5.3.2 Boundary Map

Boundary map is used to distinguish between natural and pseudo boundary pixels. Its size is a criteria to the applicability of the proposed approach. In most cases no boundary map is needed. Table 3 shows the boundary map size of the five standard images. The marginal area of the image must be large enough to record the boundary map.

From table 3 we can find that the images Lena, Airplane, Barbara and Boat are not using any boundary map for various embedding rates. But in the case of pepper image, up to an embedding rate of 0.4 bpp it can hold boundary map. Beyond that embedding rate for pepper, it does not have enough marginal pixels to hold boundary map. So embedding data in image pepper beyond 0.4 bpp is not possible in 1-LSB plane.

Boundary map size (bits)								
Embedding rate (bpp)	**0.005**	**0.01**	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Lena	0	0	0	0	0	0	0	0
Airplane	0	0	0	0	0	0	0	0
Barbara	0	0	0	0	0	0	0	0
Peppers	0	1	43	92	291	797	1741	____
Boat	0	0	0	0	0	0	0	0

Table 3: Length of boundary map under various embedding rates.

3.4 Performance Comparison with other RDH Schemes

For testing the performance of the proposed system, three other available RDH schemes from literature, Ni et. al [4], Tian [5] and Zhang [8] have also been implemented for the five test images: Lena, Boat, Barbara, Peppers and Airplane. The performance results for each test image are graphically shown in figure 4 with embedding rate (bpp) is taken on x-axis and PSNR values are taken on y-axis.

The maximum embedding rates of RDH scheme [4], [5] and [8] are 0.005, 0.4 and 0.025 respectively (figure 4). Since the implemented RDH schemes [4], [5] and [8] support data

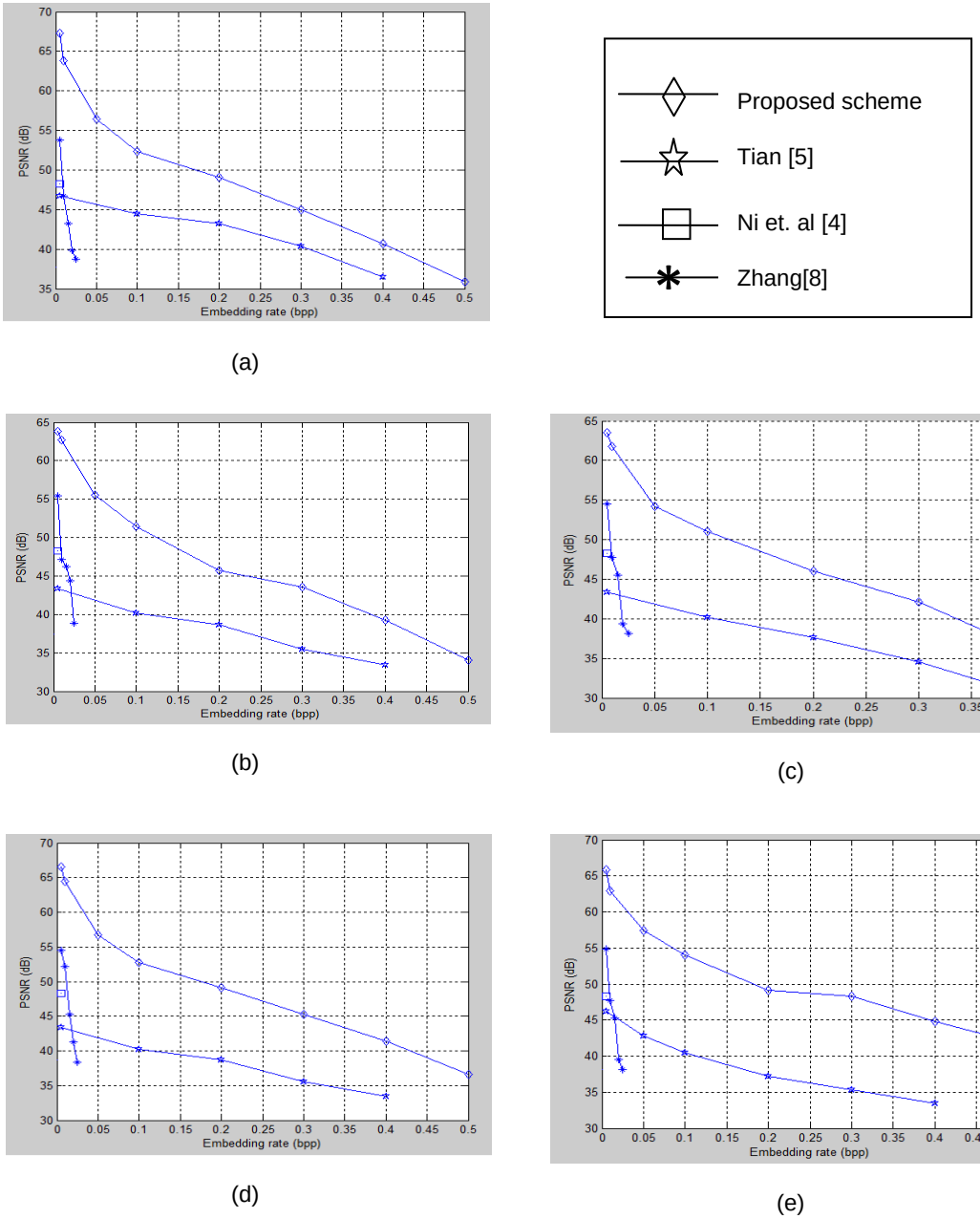


FIGURE 4: Performance of the proposed method against the three RDH schemes [4], [5], [8] for test images(a) Lena, (b) Barbara, (c) Peppers, (d) Boat, (e) Airplane.

embedding only in 1-LSB plane the proposed scheme with data embedding only in 1-LSB plane is taken for performance comparison. All the test images under proposed scheme can have a maximum embedding rate of 0.5 except peppers image. In 1-LSB plane peppers supports maximum embedding rate of 0.4 only.

From figure 4 we can also find that the proposed method shows larger PSNR values at different embedding rates. It indicates that the proposed method outperforms the other RDH schemes [4], [5] and [6] both in visual quality and payload

4. CONCLUSIONS

Reversible data hiding (RDH) is drawing attention now a days because of its ability to recover the cover without any distortion. Encryption is also used along with RDH for privacy protection. In the currently available methods RDH is implemented in encrypted images by vacating space after encryption. Moreover data hiding is done by simple LSB method in which the security of data is very less. In the proposed method space to embed data is allocated before encryption. A hyper chaotic function is also used which ensures the security of the hidden data. The hyper chaotic function shuffles the secret data based on a set of keys and these keys are difficult to track by a third person or an attacker. Hence it is prevented from brute force attack and the security of the system is increased. So all traditional RDH algorithms could achieve better performance by using the proposed system without losing secrecy. More specifically, the system is well secured. A digital water mark is also embedded into the cover image so that transmission errors and explicit modifications done by any third party can be easily identified. The proposed method has been validated by comparing the performance against three other RDH schemes [4], [5] and [8] for five test images and it is observed that the proposed method outperforms these RDH schemes both in visual quality and payload. The proposed method can achieve real reversibility, separate data extraction and great improvement on the quality of decrypted images.

In the present systems data is embedded as plain text. To increase the security some symmetric key algorithms can be used for encrypting the data to be embedded so that the encrypted data can be embedded in the image. During the first extraction at the receiver side encrypted data is retrieved as output. When the symmetric key algorithm is used the original data can be retrieved.

5. REFERENCES

[1] Johnson, Z. Duric, S. Jajodia Information Hiding. Steganography and Watermarking - Attacks and Countermeasures: Kluwer Academic Press. Norwall, 2000.

[2] J. Hwang , Kim and J. U Choi , “A reversible watermarking based on histogram shifting,” Int. Workshop on Digital Watermarking, Lecture Notes in Computer Science, Jeju Island, Korea, 2006, vol. 4283,pp. 348–361, Springer-Verlag.

[3] A. M. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” IEEE Transactions on Image Processing, Vol. 13, 2004, pp. 1147-1156.

[4] Z. Ni, Y. Shi, N. Ansari, and S. Wei, “Reversible data hiding,” IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar.2006.

[5] J. Tian, “Reversible data embedding using a difference expansion” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896,Aug. 2003.

[6] X. Zhang, “Reversible data hiding in encrypted images,” IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.

[7] W. Zhang, K. Ma, X. Zhao, N. Yu, F. Li “Reversible data hiding in encrypted images by reserving room before encryption, ”IEEE trans. Info. forensics and security ,vol. 8, no. 3, pp. 553–558, March 2013.

[8] X. Zhang, “Separable reversible data hiding in encrypted image,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 826–832, Apr. 2012.

[9] D. M Thodi and J. Rodriguez, “Expansion embedding techniques for reversible watermarking,” IEEE Trans. Image Process., vol. 16, no. 3,pp. 721–730, Mar. 2007.

[10] Y. Hu, H.K Lee, Chen and J. Li, “Difference Expansion based reversible data hiding using two embedding direction,” IEEE trans. multimedia, vol. 10, no. 8, pp. 1500–1512, 2008

[11] K. H Jung and K. Y Yoo, “Data hiding method using image interpolation,” Journ. of Compute standard and interfaces, vol. 31, pp. 465–470, 1996

[12] W. Hong, T. Chen, and H.Wu, “An improved reversible data hiding in encrypted images using side match,” IEEE Signal Process. Lett., vol.19, no. 4, pp. 199–202, Apr. 2012.

[13] L. Luo et al., “Reversible image watermarking using interpolation technique, ”IEEE Trans. Inf. Forensics Security, vol. 5, no. 1, pp. 187–193,Mar. 2010.

[14] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y.-Q. Shi, “Reversible watermarking algorithm using sorting and prediction,” IEEE Trans.Circuits Syst. Video Technol., vol. 19, no. 7, pp. 989–999, Jul.2009.

[15] Zhu, C. Zhao, X. Zhang “A novel image encryption-compression scheme using hyper-chaos and Chinese remainder theorem”, Elsevier,Image communication 670-680, 2013.

[16] Y. Wang, K. W Wong, X. F Liao, T. Xiang, G. R Chen , “A chaos based image encryption algorithm with variable control parameters “, Chaos ,solutions and Fractals 41(4) 1773-1783, 2009.

[17] I. Cox,M. Miller M, J. Bloom, J. Fridrich, T. Kalker. Digital Watermarking and Steganography Second Edition. Elsevier, 2008

Freeman Chain Code (FCC) Representation in Signature Fraud Detection Based On Nearest Neighbour and Artificial Neural Network (ANN) Classifiers

Aini Najwa Azmi *ainajwa.azmi@gmail.com*
Deparment of Computer Science,
Faculty of Computing,
Universiti Teknologi Malaysia,
Skudai, 81310, Malaysia

Dewi Nasien *dewinasien@utm.my*
Deparment of Computer Science,
Faculty of Computing,
Universiti Teknologi Malaysia,
Skudai, 81310, Malaysia

Abstract

This paper presents a signature verification system that used Freeman Chain Code (FCC) as directional feature and data representation. There are 47 features were extracted from the signature images from six global features. Before extracting the features, the raw images were undergoing pre-processing stages which were binarization, noise removal by using media filter, cropping and thinning to produce Thinned Binary Image (TBI). Euclidean distance is measured and matched between nearest neighbours to find the result. MCYT-SignatureOff-75 database was used. Based on our experiment, the lowest FRR achieved is 6.67% and lowest FAR is 12.44% with only 1.12 second computational time from nearest neighbour classifier. The results are compared with Artificial Neural Network (ANN) classifier.

Keywords: Offline Signature Verification System (SVS), Pre-processing, Thinned Binary Image (TBI), Feature Extraction, Freeman Chain Code (FCC), Nearest Neighbour, Artificial Neural Network (ANN).

1. INTRODUCTION

Biometrics authentication is subject to the identification and verification of humans by their characteristics or traits. Biometrics is widely utilise in computer science as a form of access control and identification. It is also used to recognize individuals in groups that are under surveillance. SVS is a system that identifies and verifies a handwritten signature to detect either it is genuine or forgeries. Usually, it can be done by comparing one-to-one process. Similar in personal identification, the system compares the signature information with all the images stored in the database [1]. It is very important in forensic, security and resource access control such as banking, money scam, marriage approval and user access devices. In the field of human identification, signature is one of the cheapest biometric beside DNA, fingerprint, palm print, face, vein pattern, retina and iris. These physiological traits are almost unchanged throughout of a person's life. Unlike signature, it may change with mood, environment and age. A person who does not sign in a consistent manner may have difficulty in identifying and verifying his/her signature. The database should be changed or updated in a few specified periods to make sure the authentication system is working properly. In addition, a good database must have a series of signatures from a person that are almost similar between each other for better verification. In the series, many characteristic must remain constant to determine the confidence level.

There are three groups in signature forgeries. The first one is simple forgery that the forger makes no attempt to simulate or trace a genuine signature means that the forger does not know at all how the signature looks like. The second one is random possible. The problem that we want to solve is regarding to skilled forgeries compare to simple and random forgeries [2]. Handwritten is different compare to signature. Signature must be treated as an image because people may use symbol, letter or group of letters in their signatures. This means that we cannot always identify the name of the writer when looking into the signature.

Signature Verification System (SVS) can be classified to static (offline) and dynamic (online). In an offline system, users write their signature on a paper and digitize by using a scanner or a camera. The SVS recognizes the signature by analyzing its shape or static features. In the other hand, an on-line system needs a user to write their signature in a digitizing tablet, that needs the signature in real time form. Another possibility is the acquisition by smart phone, tablet or stylus-operated PDAs. An on-line system can record dynamic features of the signature that make it difficult to forge. An on-line system is appropriate to use in real-time applications, such as financial transactions, document authenticity and office automation [1].

Generally, a SVS can be classified to four major components called as can see in Figure 1:

i. Data acquisition
ii. Pre-processing
iii. Feature extraction
iv. Verification

1.1 Data Acquisition

Data acquisition is the process of sampling signals that measure real physical conditions and converting the resulting samples into digital values that can be manipulated by a computer. A data acquisition system is converting analogue waveforms into digital values for processing. MCYT-SignatureOff-75 database will be used in the entire phases include pre-processing, feature extraction and classification. There are 75 users in this database. One user has 15 genuine signatures and 15 forgery signatures [3].

1.2 Pre-processing

Pre-processing is the process to prepare a clean signature image as an input to feature extraction. There are a lot of techniques had been done in previous works discussed in next section. As the original images contain a lot of noises and redundancy pixel, cropping, skew adjustment and thinning had been done. In an off-line handwritten Signature Authentication System (SAS), static signatures which normally scanned from a flatbed scanner contain a lot of redundant pixels that need to be removed. Besides that, size adjustments were applied to the signature images, aiming to become the authentication process more robust [4].

1.3 Feature Extraction

Feature extraction is a process that involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems occurs from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation time or a classification function which over fits the training sample and generalizes poorly to new samples. Feature extraction is a general form for methods of generating combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

1.4 Verification

Verification or classification may refer to categorization and recognization which are processes in which ideas and objects are differentiated, and understood. Classification and clustering are examples of the more general problem of pattern recognition, which is the task of some sort of output value to a given input value. Other examples are regression, which assigns a real-valued output to each input; sequence labeling, which assigns a class to each member of a sequence of

values.

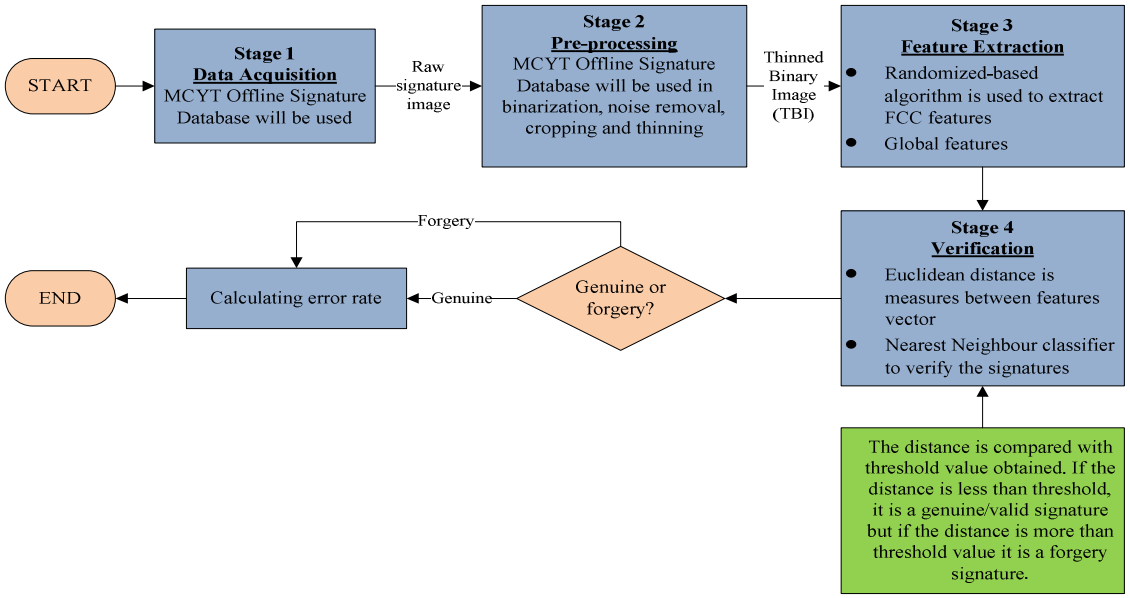


FIGURE 1: Four major components in a SVS.

2. LITERATURE REVIEW

2.1 Data Acquisition and Pre-processing

Data acquisitions for on-line and off-line systems are totally different. Signature samples will be divided to two sets which are genuine and forgery sets. In on-line system, signatures can be captured using a variety of input devices such as specially designed pens, hand gloves, special tablets, personal digital assistant (PDA) and tracking-camera [1,5]. The tablet can gather the signature position coordinates, including of X coordinates, Y coordinates, total signing duration, number of pen-ups, number of strokes and the pressure value of the pen. Some features extracted from these signatures can be used as expressing one's handwriting habit and individuality, such as pen pressure, velocity in X and Y direction [1]. All of these are called as dynamic information that we can consider as features in the next stage. In the other hand, in an off-line system, signatures are optically captured by using a scanner and the completed signatures are available as images [6]. As a scanned signature contains a lot of noise thus it must be pre-processed to produce a clean image as preparation prior feature extraction.

In an off-line system, static signatures which normally scanned from a flatbed scanner contain a lot of redundant pixels that need to be removed. Besides, size adjustments were applied to the signature images, aiming to become the authentication process more robust [4]. Some algorithm that commonly used in removing or reducing noise such as by using median filter and thinning by using morphological operation [2,7,8]. Noise removal method that based on counting filter was proposed by [9]. Besides, they also binarized and resized the image with the signature fitted to the frame or know as edges cutting [10].

Binarization means the images are converted to binary images such that signature become white and background become black. A histogram-based threshold technique was applied for binarization and the image will be stored in TIFF format [11,12] In addition, Otsu's method was used to binarize and enhance the quality of a signature image [13]. The Robert edge detection was applied on a signature image to convert it into edges and Gaussian filter was used to remove noise [2]. Color inversion, filtering and binarization were applied to the signatures images [9,15]. The true color image RGB was converted to the grayscale intensity image by eliminating the hue and saturation information while retaining the luminance and a low pass FIR was used to remove

the image high frequency components. Center of mass normalizing and smoothing by using Gaussian filter was applied in [16]. Center of mass normalizing was used to calculate image coordinates easier. Zhang-Suen Skeletonization Algorithm was applied in [9] to extract a region-based shape feature representing the general form of an object. Segmentation was done in [17, 18, 19] when the signature image was segmented to ten concentric circles instead of rectangle frame that commonly used in literature. In this case, the signature was fitted to the circular frame. By segmenting the signature image to ten concentric circles, the calculations of the feature became easy. In addition, Hamming Distance was used to stabilize the static signature image [20]. Table 1 shows summary of pre-processing methods in previous work of offline systems.

Authors	**Year**	**Pre-processing Methods Used**
Ubul *et. al* [4]	2012	Resizing
Cheng-Yaw *et. al* [8] Tomar and Singh [7] Ravi *et. al* [2]	2008 2011 2012	
Cheng-Yaw *et. al* [8]	2008 2011 2012	Noise reduction by median filter and thinning by morphological operation
Oz [9]	2005	Noise reduction by counting filter
Karouni *et. al* [16] Tomar and Singh [7]	2008 2011	Noise reduction by Gaussian filter
Ghandali & Moghaddam [10]	2008	Binarization
Porwik [11]	2007	Binarization using threshold technique
Pal *et. al* [13]	2012	Binarization using Otsu's method
Tomar and Singh [7]	2011	Robert's edge detection
Cheng-Yaw *et. al* [8] Pourshahabi *et. al* [15]	2008 2009	Colour inversion
Karouni *et. al* [16]	2008	Removing high frequency components by using low pass FIR
Karouni *et. al* [16]	2008	Center mass normalization
Cheng-Yaw *et. al* [8]	2008	Zhang-Suen Skeletonization Algorithm
Lei & Huichuan [17] Ning [19] Biswas *et. al* [18]	2009 2009 2010	Segmentation
Impedovo & Pirlo [20]	2011	Hamming Distance

TABLE 1: Summary of pre-processing methods in previous works of offline systems.

2.2 Feature Extraction

Two signatures wrote from a person are hard to compare. Some possibilities may occur such as variations in length, additions and deletions of portions of them, and changes in velocity due to pauses or hesitations of the writer [21]. Figure 2 and 3show two signatures that wrote by a person. We can see the difference between the two images but of course they have a lot of similarities that can be extracted as features that will be discussed in detail in the this section.

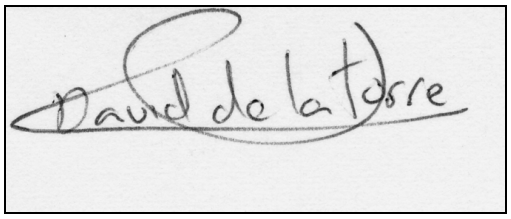


FIGURE 2: Sample signature image from person A.

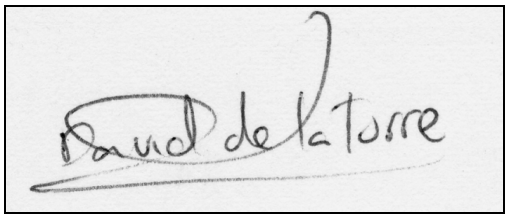


FIGURE 3: Another sample from person A.

In an off-line system, Discrete Wavelet Transform (DWT) was proposed to use in image reduction phase in order to extract the features [10, 22, 23]. DWT decomposed a signature image to sub-image based on different frequency bands. The reduced images were fused in high frequency sub image and generated a pattern matrix that will be the feature vector of a signature image. Ratios of the pixels of strokes of every section to the pixels of strokes of signature were calculated to produce the feature vector.

Different features such as direction features, gradient features, under-sampled bitmaps and modified chain-code extracted from both background and foreground components are employed for this purpose [12]. They proposed a new technique in under-sampled bitmaps that used both foreground and background pixels. A binarized signature image will be divided to 100 non-overlapping blocks and the chain code frequency for four directions was used instead of eight directions. A Robert's filter was applied on the image to obtain the gradient image after the image was binarized, normalized and resized.

Zernike moment and histogram of gradient were the two features that used by [13,14]. The Zernike moment feature obtained by solving several complex-valued polynomials. According to their passed work in [12] Robert's filter was still used to obtain the gradient image. In addition, they calculate histograms image. The intrinsic features of handwriting signature were described [24]. The features included calculation of height/width ratio, principal axis orientation, elongation, handwritten slope, amount of connected components, hole and cavity attributes and point density of different areas.

Contourlet transform was used as feature extraction in [15, 25, 26] and was an efficient tool for capturing smooth contours. Contourlet transform has five significant features which are multiresolution, localization, critical sampling, directionality and anisotropy. Contourlet transform is applied on each block of the image separately and feature vector was created for that block. By putting the 4 created feature vectors together, final feature vector was obtained. Genetic algorithm was used to find the number of optimal regions that optimally separate the spectral classes. This algorithm took into account multivariate relationships between the components of the spectral signatures [27]. The second one was an L_1-norm Support Vector Machine (SVM) that produces sparse solutions.

The pre-processed signature was considered and spatial domain features were extracted leads to global [13]. The features included height and width of signature, diagonal distance and center of gravity. Several NN-based classification models including support vector machines (SVM) with linear, polynomial, and RBF kernels and multilayer perceptron (MLP), radial basis function (RBF) network were developed for signature extraction and device identification. These models were trained and tested using spike train data gathered from the Fourier analysis of the input current waveform in the presence of multiple devices [24]. There were two approaches used in [7] for feature extraction which were energy density method and chain code method. Aspect ratio was

used as a global feature and energy density was used as local feature. Chain-code is based on the direction of the connecting pixels. Each pixel was observed for next connected pixel and their direction changes mapped by giving a numerical value to every 8-direction chain code. Statistical features such as aspect ratio, slant angle, variance, skewness, kurtosis, horizontal and vertical shift, entropy, join entropy and mutual information were extracted in [28].

A system that based on wavelet feature was proposed by [29]. Wavelet features consists horizontal and vertical projection that produced when the DWT was applied to the image. A system that based on Statistical Analysis, Theory of Estimation and Mean Variance was proposed [30]. A new Data Hiding and Extraction algorithms for data protection that detection, more verification, convincing for ownership and are more efficient for recipient and more secure was proposed in [31]. After pre-processed, the signature image was projected into feature space by using Discrete Radon Transform (DRT). As DRT yields feature space of high dimensionality, Principle Components Analysis (PCA) is introduced to compress the DRT feature without losing the significant attributes [8]. After that, PCA feature was statistically discritized into binary representation. Proportion and distribution of pixels, tilt, pressure, and centroid were applied in [32]. The proportion referred to the ratio between the height and width of the image, the distribution of pixels was made through a grid which subdivided the image and it was counted how many pixels are expressed in the current sub image. The handwritten signature in [33] was represented in time domain representation. The representation was in discrete time sequence which all the elements were in complex numbers. Feature extraction was done by mean of the Hotelling's discriminant analysis [34]. The Hotelling's approach allowed removing such features which has the smallest discriminant power. Practically, discriminant analysis is useful to decide, whether selected pair "feature–method" is important for the classification process – if not, the other pair "feature–method" is tested. The Feature-Method-Selection (FMS) was proposed when they found that there were some insufficient criteria found in the literature they had done. Geometric features were used in [16] in their feature extraction. The geometric features included area, center of gravity, eccentricity, kurtosis and skewness. The combination of variance into Dynamic Time Warping algorithm to calculate the intra-class distance (between real signatures) and inter-class distance (between real-forged signatures) was used in [35].

Three methods were used in [17] There were edge detection, Wavelet Transform and Hough Transform. Moment invariants were properties of connected regions in binary images that were invariant to translation, rotation and scaling. This method was applied in [9]. Levenshtein method in a signature recognition process was proposed in [36]. Proportion factor, center of gravity and Hough Transform were used [11] for their feature extraction. The features used in [37] were width, height, tri-Surface, six-fold surface, best fit, geometric parameters, Polar and Cartesian Direction Feature (MDF), K-Means, Histogram of frequencies, Discrete Cosine Transform and Wavelet Transform. Multi-dimensional modified grid information features were extracted according to the character of Uyghur signatures [4]. A combination of multiple distance-based classification techniques, namely individually optimized re-sampling, weighted Euclidean distance, fractional distance and weighted fractional distance was used in [38] . In order to process large amount of data, a hierarchical partitioning of data by utilizing two database reduction techniques which were feature selection and clustering and by finding the classifier that was appropriate for each signature model was proposed by [19,39]. Recently, pixel matching technique was used to verify the signature of the user with the signature that kept inside database [31] There was also a multiple feature extraction used in [40] that increased diversity of information produced in signature images. Table 2 below shows summary of feature extraction methods in previous works of offline systems.

Authors	Year	Feature Extraction Method Used
Pottier & Burel [24]	1994	Intrinsic features (height/width ratio, principal axis orientation, elongation, handwritten slope, amount of connected components, hole and cavity attributes and point density of different area)
Deng *et. al* [22] Ghandali & Moqhaddam[10] Fahmi [23] Angadi & Gour [29]	2003 2008 2010 2013	Discrete Wavelet Transform (DWT)
Pal *et. al [12]* Tomar & Singh [7]	2011 2011	Modified chain-code
Pal *et. al* [13] Pal *et. al* [14]	2012 2012	Zernike moment and histogram of gradient
Pourshahabi *et. al* [15] Soleymanpour *et. al* [25] Abushariah *et. al* [26]	2009 2010 2012	Contourlet transform
Pranckeviciene *et. al* [27]	2005	Genetic Algorithm (GA)
Akram *et. al* [28]	2012	Statistical features (aspect ratio, slant angle, variance, skewness, kurtosis, horizontal and vertical shift, entropy, join entropy and mutual information)
Bandyopadhyay *et. al* [30]	2008	Statistical Analysis, Theory of Estimation and Mean Variance
Cheng-Yaw *et. al* [8]	2008	Principle Components Analysis (PCA)
De Medeiros Napoles *et. al* [32]	2011	Global features (proportion and distribution of pixels, tilt, pressure, and centroid)
Doroz & Porwik [34]	2011	Hotelling's discriminant analysis
Karouni *et. al* [16]	2011	Feature-Method-Selection (FMS)
Palys *et. al* [37]	2013	Cartesian Direction Feature (MDF),
Oz	2005	Moment invariants
Melin [36]	2012	Levenshtein method
Ubul *et. al* [4]	2012	Multi-dimensional modified grid information features
Bhattacharya *et. al* [31]	2013	Pixel matching technique

TABLE 2: Summary of feature extraction methods in previous works of offline systems

2.3 Classifications

Neural network was one of the famous classifier that employed in previous works [20]. A modular neural network (MNN) with fuzzy integration for the problem of signature recognition was proposed by [36]. Two separate sequential neural networks were designed by [9] one for signature recognition and another for verification to detect forgery. Also a time delay neural network was used by [41]. A neural network of radial basis function optimized by Differential Evolution Algorithm with features that best discriminates between a genuine signature of a simulated forgery was proposed by [32]. A multi layer feed forward network employing a back

propagation was introduced by [7,16,28]. Several NN-based classification models including multilayer perceptron (MLP) [26,42] radial basis function (RBF) network, and support vector machines (SVM) with linear, polynomial, and RBF kernels were developed for signature extraction and device identification [29]. The proposed 2-layer perceptron neural networks were compared with other classifiers as pseudo-inverse, k-Nearest Neighbours (k-NN) and K-Means [24]. K-Nearest Neighbours (k-NN) [34,43] was developed by inserting HMM based features [18] Euclidian Distance was used by [4, 10, 15, 21, 38 ,44] in their verification stage. Support Vector Machine (SVM) [13, 14, 25, 27, 45, 46] and Nearest Neighbour (NN) were used as classifier in [12]. The Support Vector Machine (SVM), with biometric watermarking to precisely extract the signature code from the host [8]. They abbreviated the proposed method as Support Vector Machine-Biometric Watermarking) SVM-BW. The performance of SVM-BW was validated against simulated frequency and geometric attacks. An autoregressive Hidden Markov Model (AR-HMM) was employed in [47]. The correlation technique was used to match between genuine signatures [2]. A proposed Hidden Markov Model (HMM) by [48] was evaluated the robustness of their system against changes with time using long term and large scale signature database. Adaptation of the Levenshtein method in a signature recognition process is proposed by [37]. A suitable classifier was determined to a different cluster of hierarchical partitioning [38]. Neural Networks, Support Vector Machines, Bayesian classifiers or Decision Trees were known to perform well in a signature recognition system. Clustering techniques making no assumptions about the structure of the data are also widely used in this area. Global classifier was used in [49] to work on assumption that there exist forgery features that differentiate between genuine and forgery signatures. Table 3 shows summary of classification methods in previous works of offline systems.

Authors	**Year**	**Classifications Methods Used**
Pottier and Burel [24] Oz [9] Tomar & Singh [7] Karouni *et. al* [16] Jayadevan *et. al* [41] De Medeiros Napoles *et. al* [32] Akram *et. al* [28]	1994 2005 2011 2011 2012 2012 2012	Neural Network (NN)
Melin [36]	2012	Modular Neural Network (MNN)
Heinen & Osorio [42] Abushariah *et. al* [26]	2006 2012	Multi-Layer Perceptron (MLP-NN)
Srinivasan *et. al* [50]	2006	Radial Basis Function (RBF) Network
Pranckeviciene *et. al* [27] Srinivasan *et. al* [50] Cheng-Yaw *et. al* [8] Soleymanpour *et. al* [25] Batista *et. al* [46] Ribeiro *et. al* [45] Pal *et. al* [12] Pal *et. al* [13]	2005 2006 2008 2010 2010 2011 2011 2012	Support Vector Machine (SVM) Support Vector Machine (SVM)
Pottier and Burel [24] Shashi Kumar *et. al* [42] Doroz & Porwik [34] Pal *et. al* [12]	1994 2010 2011 2011	k-Nearest Neighbours (k-NN)
Ghandali & Moqhaddam [10] Pourshahabi *et. al [15]*	2008 2009	Euclidian Distance

Mendaza-Ormaza *et. al* [44] Munich & Perona [21] Ubul *et. al* [4] Moolla *et. al* [38]	2011 2011 2012 2012	
Paulik & Mohankrishnan [47]	1993	Hidden Markov Model (HMM)
Palys *et. al* [37]	2013	Levenshtein method
Moolla *et. al* [38]	2012	Bayesian Network
Putz-Leszczynska & Pacut [49]	2013	Global classifier

TABLE 3: Summary of classifications used in previous works of offline systems.

3. RESEARCH METHODOLOGY

3.1 The Research Framework

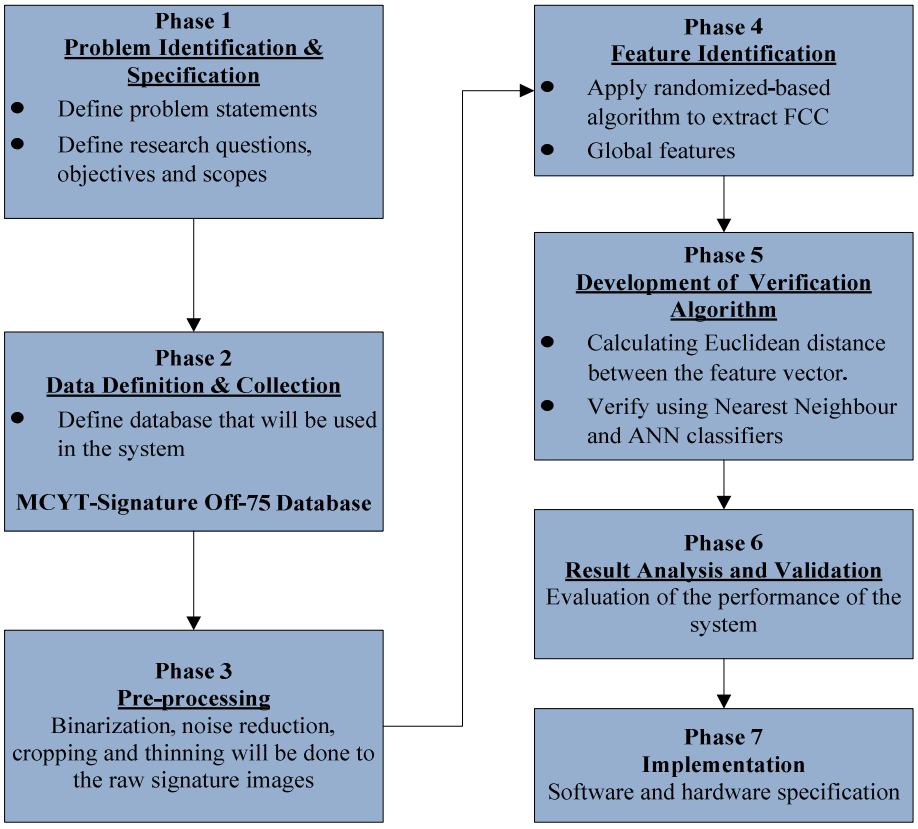


Figure 4: The research framework.

Research framework is describing the whole activity in this research. The research framework is shown in Figure 4. It shows the processes involved in each phase briefly. The research framework has seven phases, which are:

i. Problem Identification and Specification.
ii. Data Definition and Collection.
iii. Pre-Processing and Freeman Chain Code (FCC) Extraction.
iv. Development of Feature Extraction
v. Development of Verification Algorithm.
vi. Result Analysis and Validation.
vii. Implementation

3.2 The Problem of Identification and Specification

In Phase 1, the procedure started with literature review on issues related signature authentication systems. The trend of the methods for each phase can be seen that can give a clear idea on what will be done in this research. In pre- processing, several techniques will be used to prepare the signature images prior feature extraction phase. In feature extraction and selection, a randomized-based algorithm [51] will be applied to generate the chain code and followed by feature selection to be generated. Next, is the problem and description of ANN and HMM as classifiers to classify signature images. Finally the problems in verification process are to find the lowest error rate for the system.

Next, once the literature review of signature authentication system is studied, the problem statement of the research is conducted so that the objective can be achieved to answer the problem statements. The process of problem identification of the research is done by referring to the previous literatures in published papers and journals. In order to constraint the work, scope must be defined according to the decided objectives.

After the problem statements are identified, objectives and scopes of the research must be clearly defined to avoid duplication of work that has been previously made by researchers. Finally, issues on implementation specification namely software and tools for the experimentation, and hardware required for the purpose is specified. After the problem identification and specification is completed, the next step is data definition and collection

3.3 Data Definition and Collections

Data definition is a process of defining the type of data used, deciding the sources of database, checking the validity of the database, and categorizing the database for testing and validation. In the other hand, data collection means having the input data extracted or built from the original source and gathered into a compilation of huge numbers of input. Therefore, data collection is gathering relevant information in order to develop, testing, validating and analysing the algorithms.

In this research, an off-line signature database which is known as MCYT-SignatureOff-75 has been used. This database contains 75 users and originally from Universidad Autónoma de Madrid, Spain. In the case of the MCYT_Signature subcorpus, 15 client signatures and 15 highly skilled forgeries with natural dynamics are obtained for each individual [3] that equals to total of 30 signatures per user. The MCYT-SignatureOff-75 database will be used in the entire phases of this research.

3.4 Data Acquisition and Pre-processing

In pre-processing phase, binarization was done to every signature images. After that, median filter was applied to remove image noises. Next, cropping is done to the original signature images. Cropping refers to the removal of the outer parts of an image to improve framing, accentuate subject matter. This is important in aspect ratio calculation in feature extraction phase.

Cropping also must be done before skew adjustment. Finally, thinning algorithm is used to remove redundancies by eliminating specific foreground pixels. In converting raw binary image to Thinned Binary Image (TBI), thinning function in Image Processing Toolbox of MATLAB software is used respectively [51]. The proposed thinning algorithm for MCYT database is parallel thinning algorithm. Manipulation from the raw binary image to TBI is using 'bwmorph' function in the Image Processing Toolbox in MATLAB. The result of TBI will be copied automatically in a directory that is determined earlier into text format (.txt file) and for easier usage. Figure 3.4 is showing the sample signature image after every technique in feature extraction phase. Figure 3.5 shows the flow of output image of every technique used in pre-processing phase.
The signature images received are all in equal size which is 850x360 pixels. In this phase, resizing is not required. Since the database is not big, the size is acceptable in the pre-processing, feature extraction and verification phase.

Chain code representation gives the code of the boundary of signature image, the codes that is representing the direction of where is the location of the next pixel and correspond to the neighbourhood in the image. The FCC algorithm of a signature image must use binary image as input. Binary image is a image with only two gray values for each pixel, such as 0 is for background and 1 is for foreground. . An 8-direction FCC is used for descriptions of object borders in image field because of simplicity of the data representation and fast computational time. This research uses 8-neighbourhood in FCC generation of signature images as shown in Figure 5.

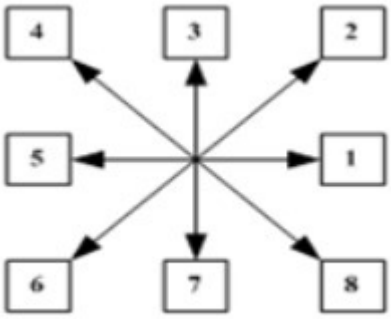


FIGURE 5: 8-Neighbourhood FCC Direction [51].

Freeman Chain Code (FCC) will be generated using randomized-based algorithm that generated the shortest computation time [51]. The randomized-based algorithm is an algorithm that makes random route choices. The advantage of randomized algorithm is the one that comes with a very high probability of correct computed results. Randomized-based algorithm is one of Heuristic technique. Heuristic technique is an optimization problem that is focused on space and time needed. Time is referred to the computational time of the program running while space is referred to the amount of computer memory needed during the program execution. A certain amount of time and space are needed to solve the computation of complexity theory [51].

The desire output is the FCC generation which is in forms of FCC length and computation time needed during generating the FCC. The first node to start the chain code tracing is randomly select. There are two possibilities, either by "node method" or "end node method". The former is to find any node based on its position at each "corner" whereas left upper, right upper, left lower and right lower. The selection of the shortest route length from the list of FCC is by selecting the chain code string with the minimum route length. If there are many minimums of route length found, the string is chose by the first time it found in the list [51].The only difference in this research is only largest contiguous pixel block of a signature image is considered in extracting the chain code because the problem will be occurred during execution since the signature image is not always in one word.

3.5 Feature Identification and Selection

Assembly of a feature vector is the target of feature identification as input for the verification phase. Sixty-nine features contained in a single vector is a combination of 2 feature parts: global features and features from character image FCC (8 FCC directional frequencies * FCC divisions).

There were three FCC directions will be tested which are 4, 8 and 16. The global features are:

(i) Signature Width (1 feature count). The signature image is scanned from left to right and measure the distance between two points in horizontal projection.

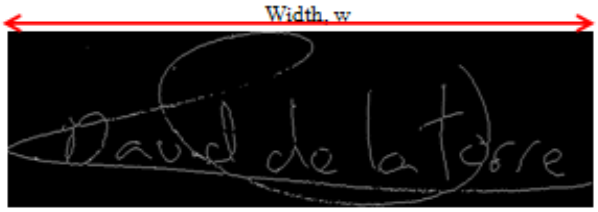


FIGURE 6: The value of width can be obtained by scanning from left to right of the image.

(ii) Signature Height (1 feature count). The signature image is scanned from top to bottom and measure the distance between two points in vertical projection.

FIGURE 7: The value of height can be obtained by scanning from top to bottom of the image.

(iii) Aspect Ratio (1 feature count). It is the ratio of signature width to height. The calculation is shown in equation 1. Figure 3.5 shows a signature image with dimension.

$$Aspect\ Ratio = \frac{\text{Signature Width,w}}{\text{Signature Height,h}} \qquad (1)$$

(iv) Diagonal Distance (2 features count): The distance is measured from left to right diagonal distance of a cropped signature image which is top right to the bottom low and top left to the bottom right. Blue line is the diagonal line, ld. Refer Figure 8.

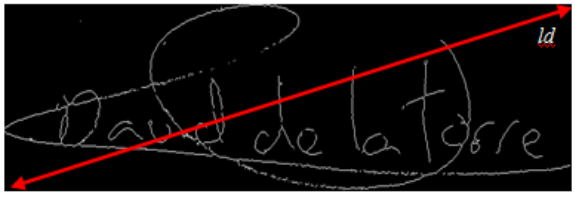


FIGURE 8: After the image was cropped, the values of left and right diagonal can be obtained.

(v) Centres of mass of all foreground pixels in an image (2 features count): is calculated for signature image by adding all x and y locations of foreground pixels and dividing it by number of pixel counted.

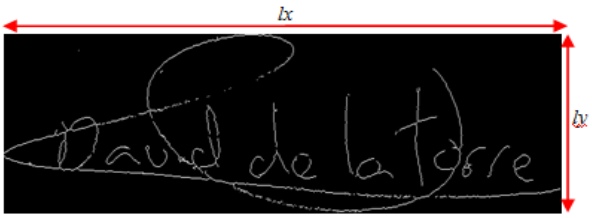


FIGURE 9: A signature image after pre-processing process.

Based on Figure 9, the equation to find the centres of mass for x and y locations:

$$x_{centre\ of\ mass} = \sum_{x=0}^{x=lx} x\, f(x) \quad (2)$$

$$y_{centre\ of\ mass} = \sum_{y=0}^{y=ly} y\, f(y) \quad (3)$$

(vi) Counting pixel value total shift per horizontal/vertical line. There are calculated by slicing the image horizontally into four parts and by summing shifts from black to white or white to black image. For vertical shifts, image is to be sliced vertically. This information is another unique property of signature because the chances of two signatures having exactly same shift numbers are very low.

$$Feature\ count = 4\ \text{lines}\ (2\ \text{directions}) = 8 \quad (4)$$

Feature vector formed from the MCYT signature database will be compiled to verification phase. Total of 32 features from FCC and 15 features from global features will be combined as one feature vector and verification input later.

3.6 Development of Verification Algorithm

Verification is the process of testing whether a claimed signature is the same writer (class) or not as the set of signatures trained in the system for that class. In our case, we have trained 12 genuine signatures and test 3 genuine and all 15 forgery signatures. Verification involved loading the template MATLAB file enrolled in the system and comparing its stored parameters. Nearest Neighbour (NN) classifier performs matching score calculation based on Euclidean distance [52]. Euclidean distance is one of the most favourite method for measuring the distance between vectors. The function has the following prototype:

results = nn_classification PhD(train, train_ids, test, test_ids, n, dist, match_kind);
Here train and test denote the training and testing feature vector matrices which are train_ids and test_ids. They stand for corresponding label vectors n represents the number of feature used in the calculations of Euclidean distance, dist ('euc') denoted the matching distance to use and match_kind ('all' (default)) represented a string identifier controlling the matching procedure. When a similarity matrix needs to be constructed, where all feature vectors from the training-feature-vector-matrix train are matched against all feature vectors in the test-feature-vector-matrix test. The similarity matrix generated by this option is stored in the results structure under results.matchdist. The structure can then be stored, where performance graphs and metrics are computed from the entire similarity matrix. The corresponding verification error rates are computed by pooling FRR data from the valid similarity matrix and FAR data from the forgery similarity matrix.

In identification process, the lowest distance between feature vector of input image and stored feature vectors is computed by using Euclidean distance and its related signature class is specified. In verification process for each signature class, a reference point is considered, if the distance between feature vector of input image and this reference point is less than a specific threshold, input image is a valid signature otherwise it is a forgery signature. A threshold value can be considered as a vector containing mean of corresponding elements of feature vectors of each class

3.7 Result Analysis and Validation

The performance quality is measured by FAR (False Acceptance Rate) and FRR (False Rejection Rate). FAR is the rate of accepting forgery signature as genuine signature wrongly. FRR is the rate of rejecting genuine signature as forgery one wrongly. FAR and FRR are related to each other inversely. By setting and changing a threshold, when FAR is increasing, FRR is decreasing. Equation 5 and 6 show the formulas of FAR and FRR.

$$FAR = \frac{Number\ of\ Falsely\ Accepted\ Images}{Total\ number\ of\ person\ in\ the\ database} \quad (5)$$

$$FRR = \frac{Number\ of\ Falsely\ Rejected\ Images}{Total\ number\ of\ person\ in\ the\ database} \quad (6)$$

In pre-processing stage, it provided "input base directory", that is sample image path for noise removal and thinning. "File extension" is used to search for other image extension. Current database sample uses file.bmp so this parameter can be ignored. The script searches recursively (all files and subdirectory) starting from its current location.

In feature extraction stage, it provided "input base directory", that is the signature image path. Output feature directory is set for output location. It is usually the same as pre-processing part in order to ensure all features to be located in same place.. Freeman Chain Code (FCC) extraction is done in this part.

In verification stage, it provided "input feature directory" which is one directory level up from intended location. The intended location name is put at "sample user number". This configuration is due to database having 75 folders with numbers labeling the sample. It also provided "output directory" to save trained parameters with its input/output data and result. The training is based on specific folder, not recursively as in previous two phases. Since there are 15 samples for both valid and forgery classes the data can be divided based on "data division" value. There are 12 valid signatures will be trained while the rest 3 valid signatures and all 15 forgery signatures will be tested.

3.8 Implementations

This section explains the equipment required in the implementation of the proposed framework. The requirements to develop the integrated system are classified into two parts: hardware and software. The hardware specification used in this research was a Asus Model s400c laptop with Intel ® Core ™ i5-3317U CPU @ 1.7 GHz, 4 Core(s) processor, 8GB RAM memory, 500 GB hard disk and Windows 8 64-bit operating system. MATLAB R2008a (version 7.6.0) is used as platform to write the code for pre-processing, FCC and feature extraction for nearest neighbour.

4. EXPERIMENTAL RESULT AND DISCUSSIONS

In this section, the result and some comparisons from previous work are highlighted. Table 4 and Table 5 show the results from both classifiers. Result from Nearest Neighbour classifier shows better result and faster computational time compare to result from ANN. In the other hand, Table 6 below shows comparison between our work and previous works.

Chain Code Division	4	8	16
FRR (%)	6.67	6.67	10.67
FAR (%)	12.44	12.44	12.00
Computational Time (s)	1.12	3.03	3.42

TABLE 4: Results from Nearest Neighbour based on Euclidean Distance.

Chain Code Division	4	8	16
FRR (%)	16.67	16.00	15.10
FAR (%)	20.22	21.60	20.90
Computational Time (s)	7.80	11.35	12.00

TABLE 5: Results from ANN.

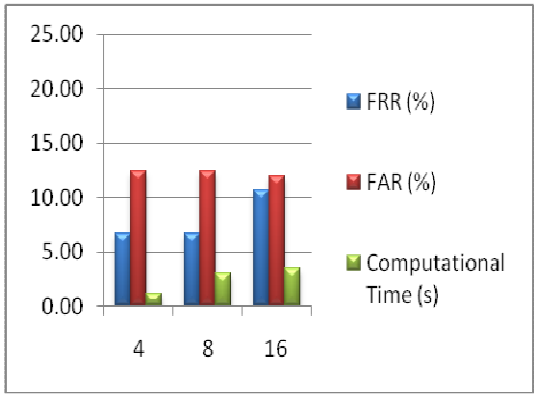


FIGURE 10: Trend chart results computed from Nearest Neighbour.

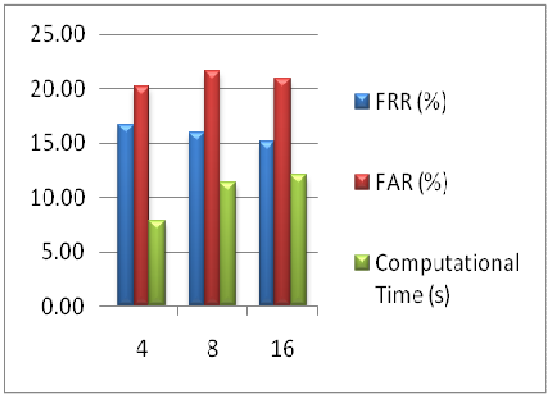


FIGURE 11: Trend chart results computed from ANN.

Authors	Pre-processing	Feature Extraction	Classification	Performance	Dataset
Our proposed system	Binarization, noise removal, cropping and thinning	FCC and global features	Euclidean Distance and Nearest Neighbour	FAR: 12.44% FRR:6.67%	MCYT
Pourshahabi	Colour inversion	Contourlet Transform	Euclidean Distance	FAR: 22.72% FRR:23.18%	Self-collected
Ghandali & Moghaddam (2008)	Binarization	Discrete Wavelet Transform (DWT)	Euclidean Distance	FAR: 8.5% FRR:11.1%	Self-collected
Moolla *et. al*, 2012	Binarization and thinning	Modified direction feature edge detection,	Modular neural network (MNN)	TSR: 89.2%	GPDS

		wavelet transform and Hough transform			
Ubul *et. al*, 2012	Noise reduction, binarization and normalization	Multi-dimensional modified grid information features	Euclidean Distance, K-Nearest Neighbor (KNN) and Bayesian Euclidean Distance	Accuracy: 86.45%	Self-collected

TABLE 6: Comparison between our work and previous works.

The database we used has 75 signers. As mentioned before, each signer has 15 valid signatures and 15 forgery signatures and total of signatures is 2250 images. In verification stage, the signatures will be divided to training and testing phases. Twelve images from each signer will be trained, three valid signatures and all fifteen forgery signatures will be tested. There are 180 valid signature from 75 signers will be trained and the rest are tested to yield the desire results.

Based on our systems, the lowest FRR achieved is 6.67% and lowest FAR is 12.44% with only 1.1178 second computational time from nearest neighbour classifier. Our results are satisfactory yet better than some previous works. Comparing with work from Moolla *et. al* [38], they are also used Euclidean distance with min FRR and FAR for skilled forgeries is 15.32% each. They investigated the use of a combination of multiple distance-based classification techniques, namely individually optimized re-sampling, weighted Euclidean distance, fractional distance and weighted fractional distance. In feature extraction stage, they are used directional features and modified directional features. Furthermore, Ghandali and Moghaddam [10] work was identifying and verifying handwritten signatures that were based on image registration Discrete Wavelete Transform (DWT) and image fusion. The classifier is based on Euclidean distance to obtain FAR 8.5% and FRR 11.1%. While in [4], they obtained 86.45% in Euclidean Distance (ED) classifier with extracted features that based on grid feature. Lastly, the work from [15] reported that they applied a special type of Contourlet transform on signature image and also related Contourlet coefficients are computed and feature vector is created. Euclidean distance is used as classifier. The results obtained for English signatures are FAR 22.72% and FRR 23.18%.

The algorithms and methods used in this system especially in feature extraction stage are simple and utilized less memory. There is no involving of complicated mathematic formula and easy to understand. This is the reason why the computational time became very short. A short computational time is important in this time for any security system when people are all in rush. Furthermore, in the context of number of features, our features are only 47. Compared to some previous works like in [4], the number of features was 120 which will need a lot of time and memory in classification stage. An optimum number of features is really important to make sure the system is working in an optimum efficiency. Based on our experiment, larger chain code division which leads to bigger number of features do not improved the error rate yet the computational time become longer that we need to avoid.

5. CONCLUSIONS AND FUTURE WORKS

This paper presents a signature verification system that used (FCC) as directional feature and data representation. There are 47 features were extracted from the signature images from six global features. Before extracting the features, the raw images were undergoing pre-processing stage which were binarization, noise removal by using media filter, cropping and thinning to produced Thinned Binary Image (TBI). Euclidean distance is measured and matched between nearest neighbours to find the result. MCYT-SignatureOff-75 database was used. Based on our

systems, the lowest FRR achieved is 6.67% and lowest FAR is 12.44% with only 1.1178 second computational time.

In the future, we are planning to calculate the value of Equal Error Rate (EER) from the system. Besides, we are going to seek and explore new methods or algorithms for feature extraction to improve the results. Furthermore, we are planning to build a hybrid classifier to test the effect on error rate and accuracy of the system.

Acknowledgement This work is supported by the Universiti Teknologi Malaysia and Ministry of Education Malaysia with Research Grant (FRGS – 4F264). The authors would like to thanks Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM) for the support in R & D, and Soft Computing Research Group (SCRG) for the inspiration in making this study success. The authors would also like to thank the anonymous reviewers who have contributed enormously to this work.

6. REFERENCES

[1] Al-Mayyan, Waheeda, Own, Hala S., & Zedan, Hussein. (2011). Rough set approach to online signature identification. *Digital Signal Processing, 21*(3), 477-485. doi: http://dx.doi.org/10.1016/j.dsp.2011.01.007.

[2] Ravi, J., Hosamani, S., & Raja, K. B. (2012, 26-28 July 2012). *Off-line Signature Identification Based on DWT and Spatial Domain Features.* Paper presented at the Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on Computing Communication & Networking Technologies.

[3] Ortega-Garcia, J., et al. (2003).: MCYT baseline corpus: A bimodal biometric database. IEEE Proceedings Vision, Image and Signal Processing 150(6).

[4] Ubul, K., Adler, A., Abliz, G., Yasheng, M., & Hamdulla, A. (2012, 2-5 July 2012). *Off-line Uyghur signature recognition based on modified grid information features.* Paper presented at the Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on Information Science, Signal Processing and their Applications.

[5] Zhang, Zhaoxiang, Wang, Kaiyue, & Wang, Yunhong. (2011). A Survey of On-line Signature Verification. In Z. Sun, J. Lai, X. Chen & T. Tan (Eds.), *Biometric Recognition* (Vol. 7098, pp. 141-149): Springer Berlin Heidelberg.

[6] Zhan, Enqi, Guo, Jinxu, Zheng, Jianbin, Ma, Chan, & Wang, Linjuan. (2009, 15-16 May 2009). *On-line Handwritten Signature Verification Based on Two Levels Back Propagation Neural Network*. Paper presented at the Intelligent Ubiquitous Computing and Education, 2009 International Symposium on Intelligent Ubiquitous Computing and Education.

[7] Tomar, Minal, & Singh, Pratibha. (2011). An Intelligent Network for Offline Signature Verification Using Chain Code. In N. Meghanathan, B. Kaushik & D. Nagamalai (Eds.), *Advanced Computing* (Vol. 133, pp. 10-22): Springer Berlin Heidelberg.

[8] Cheng-Yaw, Low, Beng-Jin Teoh, A., & Connie, Tee. (2008, 3-5 June 2008). *Support Vector Machines (SVM)-based biometric watermarking for offline handwritten signature.* Paper presented at the Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on Industrial Electronics and Applications.

[9] Oz, Cemil. (2005). Signature Recognition and Verification with Artificial Neural Network Using Moment Invariant Method. In J. Wang, X.-F. Liao & Z. Yi (Eds.), *Advances in Neural Networks – ISNN 2005* (Vol. 3497, pp. 195-202): Springer Berlin Heidelberg.

[10] Ghandali, S., & Moghaddam, M. E. (2008, 16-19 Dec. 2008). *A Method for Off-line Persian Signature Identification and Verification Using DWT and Image Fusion.* Paper presented at

the Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on Signal Processing and Information Technology.

[11] Porwik, P. (2007, 28-30 June 2007). *The Compact Three Stages Method of the Signature Recognition.* Paper presented at the Computer Information Systems and Industrial Management Applications, 2007. CISIM '07. 6th International Conference on Computer Information Systems and Industrial Management Applications.

[12] Pal, S., Alireza, A., Pal, U., & Blumenstein, M. (2011, 6-8 Dec. 2011). *Off-line Signature Identification Using Background and Foreground Information.* Paper presented at the Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on Digital Image Computing Techniques and Applications.

[13] Pal, S., Alireza, A., Pal, U., & Blumenstein, M. (2012, 4-7 Dec. 2012). *Multi-script off-line signature identification.* Paper presented at the Hybrid Intelligent Systems (HIS), 2012 12th International Conference on Hybrid Intelligent Systems.

[14] Pal, S., Pal, U., & Blumenstein, M. (2012, 10-15 June 2012). *Off-line English and Chinese signature identification using foreground and background features.* Paper presented at the Neural Networks (IJCNN), The 2012 International Joint Conference on Neural Networks.

[15] Pourshahabi, M. R., Sigari, M. H., & Pourreza, H. R. (2009, 4-7 Dec. 2009). *Offline Handwritten Signature Identification and Verification Using Contourlet Transform.* Paper presented at the Soft Computing and Pattern Recognition, 2009. SOCPAR '09. International Conference of Soft Computing and Pattern Recognition.

[16] Karouni, Ali, Daya, Bassam, & Bahlak, Samia. (2011). Offline signature recognition using neural networks approach. *Procedia Computer Science, 3*(0), 155-161. doi: http://dx.doi.org/10.1016/j.procs.2010.12.027.

[17] Lei, Liu, & Huichuan, Duan. (2009, 14-16 Aug. 2009). *The Research of handwritten signatures.* Paper presented at the IT in Medicine & Education, 2009. ITIME '09. IEEE International Symposium on IT in Medicine & Education.

[18] Biswas, Samit, Bhattacharyya, Debnath, Kim, Tai-hoon, & Bandyopadhyay, SamirKumar. (2010). Extraction of Features from Signature Image and Signature Verification Using Clustering Techniques. In T.-h. Kim, A. Stoica & R.-S. Chang (Eds.), *Security-Enriched Urban Computing and Smart Grid* (Vol. 78, pp. 493-503): Springer Berlin Heidelberg.

[19] Ning, Wang. (2009, 24-26 Nov. 2009). *Signature Identification Based on Pixel Distribution Probability and Mean Similarity Measure with Concentric Circle Segmentation.* Paper presented at the Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on Computer Sciences and Convergence Information Technology.

[20] Impedovo, Donato, & Pirlo, Giuseppe. (2011). Stability Analysis of Static Signatures for Automatic Signature Verification. In G. Maino & G. Foresti (Eds.), *Image Analysis and Processing – ICIAP 2011* (Vol. 6979, pp. 241-247): Springer Berlin Heidelberg.

[21] Munich, M. E., & Perona, P. (2003). Visual identification by signature tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 25*(2), 200-217. doi: 10.1109/TPAMI.2003.1177152.

[22] Deng, P. S., Li-Jing, Jaw, Jau-Hwang, Wang, & Cheng-Tan, Tung. (2003, 14-16 Oct. 2003). *Trace copy forgery detection for handwritten signature verification.* Paper presented at the Security Technology, 2003. Proceedings. IEEE 37th Annual 2003 International Carnahan Conference on Security Technology.

[23] Fahmy, Maged M. M. (2010). Online handwritten signature verification system based on DWT features extraction and neural network classification. *Ain Shams Engineering Journal, 1*(1), 59-70. doi: http://dx.doi.org/10.1016/j.asej.2010.09.007.

[24] Pottier, I., & Burel, G. (1994, 27 Jun-2 Jul 1994). *Identification and authentification of handwritten signatures with a connectionist approach.* Paper presented at the Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on Neural Networks.

[25] Soleymanpour, E., Rajae, B., & Pourreza, H. R. (2010, 27-28 Oct. 2010). *Offline handwritten signature identification and verification using contourlet transform and Support Vector Machine.* Paper presented at the Machine Vision and Image Processing (MVIP), 2010 6th Iranian.

[26] Abushariah, A. A. M., Gunawan, T. S., Chebil, J., & Abushariah, M. A. M. (2012, 3-5 July 2012). *Automatic person identification system using handwritten signatures.* Paper presented at the Computer and Communication Engineering (ICCCE), 2012 International Conference on Computer and Communication Engineering.

[27] Pranckeviciene, Erinija, Somorjai, Ray, Baumgartner, Richard, & Jeon, Moon-Gu. (2005). Identification of signatures in biomedical spectra using domain knowledge. *Artificial Intelligence in Medicine, 35*(3), 215-226. doi: http://dx.doi.org/10.1016/j.artmed.2004.12.002.

[28] Akram, M., Qasim, R., & Amin, M. A. (2012, 18-19 May 2012). *A comparative study of signature recognition problem using statistical features and artificial neural networks.* Paper presented at the Informatics, Electronics & Vision (ICIEV), 2012 International Conference on Informatics, Electronics & Vision.

[29] Angadi, S. A., & Gour, Smita. (2013). LVQ-Neural Network Based Signature Recognition System Using Wavelet Features. In M. S & S. S. Kumar (Eds.), *Proceedings of the Fourth International Conference on Signal and Image Processing 2012 (ICSIP 2012)* (Vol. 222, pp. 1-13): Springer India.

[30] Bandyopadhyay, S. K., Bhattacharyya, D., Das, P., & Debnath, D. (2008, 24-26 April 2008). *Handwritten Signature Authentication Using Statistical Estimation.* Paper presented at the Multimedia and Ubiquitous Engineering, 2008. MUE 2008. International Conference on Multimedia and Ubiquitous Engineering.

[31] Bhattacharya, Indrajit, Ghosh, Prabir, & Biswas, Swarup. (2013). Offline Signature Verification Using Pixel Matching Technique. *Procedia Technology, 10*(0), 970-977. doi: http://dx.doi.org/10.1016/j.protcy.2013.12.445.

[32] De Medeiros Napoles, S. H. L., & Zanchettin, C. (2012, 10-15 June 2012). *Offline handwritten signature verification through network radial basis functions optimized by Differential Evolution.* Paper presented at the Neural Networks (IJCNN), The 2012 International Joint Conference on Neural Networks.

[33] Dhar, K., & Kunz, A. (1988, 5-7 Oct 1988). *Digital technique to analyze handwritten signatures.* Paper presented at the Security Technology, 1988. Crime Countermeasures, Proceedings. Institute of Electrical and Electronics Engineers 1988 International Carnahan Conference on Security Technology.

[34] Doroz, Rafal, & Porwik, Piotr. (2011). Handwritten Signature Recognition with Adaptive Selection of Behavioral Features. In N. Chaki & A. Cortesi (Eds.), *Computer Information Systems – Analysis and Technologies* (Vol. 245, pp. 128-136): Springer Berlin Heidelberg.

[35] Doroz, Rafal, & Wrobel, Krzysztof. (2012). Dynamic Signature Recognition Based on Modified Windows Technique. In A. Cortesi, N. Chaki, K. Saeed & S. Wierzchoń (Eds.), *Computer Information Systems and Industrial Management* (Vol. 7564, pp. 158-167): Springer Berlin Heidelberg.

[36] Melin, Patricia. (2012). Signature Recognition with a Hybrid Approach Combining Modular Neural Networks and Fuzzy Logic for Response Integration *Modular Neural Networks and Type-2 Fuzzy Systems for Pattern Recognition* (Vol. 389, pp. 77-92): Springer Berlin Heidelberg.

[37] Palys, Malgorzata, Doroz, Rafal, & Porwik, Piotr. (2013). Statistical Analysis in Signature Recognition System Based on Levenshtein Distance. In R. Burduk, K. Jackowski, M. Kurzynski, M. Wozniak & A. Zolnierek (Eds.), *Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013* (Vol. 226, pp. 217-226): Springer International Publishing.

[38] Moolla, Y., Viriri, S., Nelwamondo, F. V., & Tapamo, J. R. (2012, 12-15 Aug. 2012). *Handwritten signature verification using weighted fractional distance classification.* Paper presented at the Signal Processing, Communication and Computing (ICSPCC), 2012 IEEE International Conference on Signal Processing, Communication and Computing.

[39] Potolea, Rodica, Bărbănţan, Ioana, & Lemnaru, Camelia. (2011). A Hierarchical Approach for the Offline Handwritten Signature Recognition. In J. Filipe & J. Cordeiro (Eds.), *Enterprise Information Systems* (Vol. 73, pp. 264-279): Springer Berlin Heidelberg.

[40] Rivard, Dominique, Granger, Eric, & Sabourin, Robert. (2013). Multi-feature extraction and selection in writer-independent off-line signature verification. *International Journal on Document Analysis and Recognition (IJDAR), 16*(1), 83-103. doi: 10.1007/s10032-011-0180-6.

[41] Jayadevan, R., Kolhe, S. R., Patil, P. M., & Pal, U. (2012). Automatic processing of handwritten bank cheque images: a survey. *International Journal on Document Analysis and Recognition (IJDAR), 15*(4), 267-296. doi: 10.1007/s10032-011-0170-8.

[42] Heinen, M. R., & Osorio, F. S. (2006, 0-0 0). *Handwritten Signature Authentication using Artificial Neural Networks.* Paper presented at the Neural Networks, 2006. IJCNN '06. International Joint Conference on Neural Networks.

[43] Shashi Kumar, D. R., Ravi Kumar, R., Raja, K. B., Chhotaray, R. K., & Pattanaik, Sabyasachi. (2010). Combined Off-Line Signature Verification Using Neural Networks. In V. Das & R. Vijaykumar (Eds.), *Information and Communication Technologies* (Vol. 101, pp. 580-583): Springer Berlin Heidelberg.

[44] Mendaza-Ormaza, A., Miguel-Hurtado, O., Blanco-Gonzalo, R., & Diez-Jimeno, F. (2011, 18-21 Oct. 2011). *Analysis of handwritten signature performances using mobile devices.* Paper presented at the Security Technology (ICCST), 2011 IEEE International Carnahan Conference on Security Technology.

[45] Ribeiro, Bernardete, Gonçalves, Ivo, Santos, Sérgio, & Kovacec, Alexander. (2011). Deep Learning Networks for Off-Line Handwritten Signature Recognition. In C. San Martin & S.-W. Kim (Eds.), *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* (Vol. 7042, pp. 523-532): Springer Berlin Heidelberg.

[46] Batista, Luana, Granger, Eric, & Sabourin, Robert. (2010). A Multi-Classifier System for Off-Line Signature Verification Based on Dissimilarity Representation. In N. Gayar, J. Kittler & F. Roli (Eds.), *Multiple Classifier Systems* (Vol. 5997, pp. 264-273): Springer Berlin Heidelberg.

[47] Paulik, M. J., & Mohankrishnan, N. (1993, 16-18 Aug 1993). *A 1-D, sequence decomposition based, autoregressive hidden Markov model for dynamic signature identification and verification.* Paper presented at the Circuits and Systems, 1993, Proceedings of the 36th Midwest Symposium on Circuits and Systems.

[48] Wada, N., & Hangai, S. (2007, 7-8 June 2007). *HMM Based Signature Identification System Robust to Changes of Signatures with Time.* Paper presented at the Automatic Identification Advanced Technologies, 2007 IEEE Workshop on Automatic Identification Advanced Technologies.

[49] Putz-Leszczynska, Joanna, & Pacut, Andrzej. (2013). Universal Forgery Features Idea: A Solution for User–Adjusted Threshold in Signature Verification. In N. Nguyen (Ed.), *Transactions on Computational Collective Intelligence IX* (Vol. 7770, pp. 152-172): Springer Berlin Heidelberg.

[50] Srinivasan, D., Ng, W. S., & Liew, A. C. (2006). Neural-network-based signature recognition for harmonic source identification. *Power Delivery, IEEE Transactions on, 21*(1), 398-405. doi: 10.1109/TPWRD.2005.852370.

[51] Nasien, Dewi (2012). Feature Extraction and Selection Algorithm for Chain Code Representation of Handwritten Character. Ph. D Thesis. Universiti Teknologi Malaysia, Malaysia.

[52] Štruc V., Pavešic, N.: The Complete Gabor-Fisher Classifier for Robust Face Recognition, EURASIP Advances in Signal Processing, vol. 2010, 26 pages, doi:10.1155/2010/847680, 2010.

Homomorphic Filtering of Speckle Noise From Computerized Tomography (CT) Images Using Adaptive Centre-Pixel-Weighed Exponential Filter

Martin C. Eze *martin.eze@unn.edu.ng*
Department of Electronic Engineering
Faculty of Engineering
University of Nigeria
Nsukka, 410001, Nigeria

Ogechukwu N. Iloanusi *ogechukwu.iloanusi@unn.edu.ng*
Department of Electronic Engineering
Faculty of Engineering
University of Nigeria
Nsukka, 410001, Nigeria

Uche A. Nnolim *uche.nnolim@unn.edu.ng*
Department of Electronic Engineering
Faculty of Engineering
University of Nigeria
Nsukka, 410001, Nigeria

Charles C. Osuagwu *charles.osuagwu@unn.edu.ng*
Department of Electronic Engineering
Faculty of Engineering
University of Nigeria
Nsukka, 410001, Nigeria

Abstract

Adaptive filters are needed to accurately remove noise from noisy images when the variance of noise present varies. Linear filter such as Exponential filter becomes effective in removing speckle noise when homomorphic filtering technique is used. In this paper, an Adaptive Centre-Pixel-Weighed Exponential Filter for removing speckle noise from CT images was developed. The new filter is based on varying the centre-pixel of the filter kernel based on the estimated speckle noise variance present in a noisy CT image. Ten samples of 85x73 CT images corrupted by speckle noise level ranging from 10% to 30% were considered and the new technique gave a reasonably accurate speckle noise filtering performance with an average Peak Signal to Noise Ratio (PSNR) of **70.2839dB** compared to **69.0658dB** for Wiener filter and **64.3711dB** for the Binomial filter. The simulation software used in the paper is Matrix Laboratory (Matlab).

.

Keywords: Adaptive Filter, Exponential Filter, Speckle Noise, Homomorphic Filtering, CT Image, Centre-pixel, Centre-pixel-weighed.

1. INTRODUCTION

An adaptive filter efficiently removes noise from noisy images based on the magnitude of the estimated noise variance present in a noisy image [1]. It is required when the specifications of the noise present in an image is unknown or the noise specifications cannot be satisfied by time-invariant filter [2]. Adaptive filters depend on the variation of filter centre-pixel weight or filter window sizes based on the noise variance present in a noisy image to perform effectively [3]. An adaptive centre-pixel weighed filter is a filter in which only the centre pixel of the kernel is tuned

based on the noise variance present in the noisy image. Adaptive filters are divided into adaptive linear filters and adaptive nonlinear filters. Adaptive nonlinear filters are effective in removing both additive and multiplicative noise from noisy images unlike adaptive linear filters that performs effectively for additive noise [4].

To effectively filter multiplicative noise using linear filters, homomorphic filtering technique is applied. Homomorphic filtering involves transforming multiplicative noise into additive noise and then applying a linear operation [5]. For a linear filter to be applied to images corrupted by speckle noise, the speckle noise needs to be transformed to additive noise using a logarithmic transformation [6] before the convolution operation is applied [7]. The logarithmic transformation is used to transform speckle noise. For non-speckle noises, other transform techniques such as Ascombe transform (for transforming Poisson noise to additive noise) and Ascombe-like transform (for transforming a combined additive and Poisson-like noise) exist [8],[9]. The process of transforming speckle noise to additive noise is known as logarithmic transformation. The logarithmic transformation is applied to speckle noise so that linear filtering is used to enhance the noisy image.

The removal of speckle noise from image is more complex compared to the removal of additive noise. Some researchers in the past have developed and used different filters to remove speckle noise from medical images. Medical images that has been given much attention are images captured using the CT machine because such images are very popular and also of high quality [10]. In a work done by [11], ε-Neighbourhood median filter was used to remove speckle noise from CT lung images. In another work done by [12], the effectiveness of wiener filter in removing speckle noise from CT images was investigated. In the work done by [13], Genetic-Neuro-Fussy technique was developed and used to remove speckle noise from ultrasound images. In a closely related work done by [14], a Bayesshrink Wavelet Threshold technique was used to remove speckle noise from ultrasound images. In a related work done by [15], Mathematical Morphology technique was developed and used in removing speckle noise from images.

In this paper, an Adaptive Centre-Pixel-Weighed Exponential Filter is developed. This technique is an example of an adaptive linear filter. The new technique uses the variation in the magnitude of filter centre pixel with the noise variance present in the image to achieve high performance.

2. METHODOLOGY

The filter technique developed in this work is an Adaptive Centre-Pixel-Weighed Exponential Filter. The developed technique uses the estimated speckle noise variance present in an image to vary the centre pixel of the filter for effective filtering. The variable weight technique is developed in this work because it is simpler than variable window technique and causes less blurring.

2.1 Proposed Model

In general, noiseless image, Y(m,n) corrupted by a multiplicative noise, $\mathcal{N}(m,n)$ is given as shown in (1) where X(m,n) is the noisy image [16].

$$X(m,n) = Y(m,n)\mathcal{N}(m,n) \quad (1)$$

The expression in (1) is modified by adding a correcting factor (β) to both sides as given in (2).

$$X(m,n) + \beta = Y(m,n)\mathcal{N}(m,n) + \beta \quad (2)$$

The parameter, β in (2) is a real number such that $\beta \leq 10^{-7}$. Transforming the modified noisy image in (2) into image corrupted by additive noise using a logarithmic transformation according to [16], yields (3).

$$log_e(X(m,n)+\beta) = log_e(Y(m,n)N(m,n)+\beta) \quad (3)$$

The correcting factor (β) is added to the (1) before logarithmic transformation because the pixels with zero values give -∞ after logarithm transformation and therefore, zero pixels need modification to ensure real value after transformation [17]. The value of β is made very negligible to avoid excessive change in pixel values. The expression in (3) is enhanced by linear filter, h (m, n) to obtain the filtered image, $\bar{Y}(m,n)$ using a convolution operation as given in (4).

$$\bar{Y}(m,n) = Exp\left(\sum_{j=-R_f}^{R_f}\sum_{k=-C_f}^{C_f} h(j,k)\log_e(X(m-j,n-k)+\beta)\right) \quad (4)$$

Where $R_f = \frac{M_f-1}{2}$ *and* $C_f = \frac{N_f-1}{2}$ and M_f and N_f are the row and column sizes of the filter respectively. After filtering, the mean square error (MSE) in removing noise from noisy image using a linear filter is given in (5).

$$MSE = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}[Y(m,n)-\bar{Y}(m,n)]^2 \quad (5)$$

The parameters, M and N in (5) are the image row and column length respectively. The parameter, h(m,n) in (4) that gives the minimum value of MSE (5) is the proposed Adaptive Centre-Pixel-Weighed Exponential Filter. The general expression for the proposed filter is given in (6) where Z is the tuning factor of the filter.

$$h(m,n) = \frac{2Z}{\pi}e^{-2(m^2+n^2)} \quad (6)$$

The value of Z that minimizes the expression in (5) as established in this work is given in (7) for 3×3 filter kernel.

$$Z = \begin{cases} 1.5 & : |m|+|n| \neq 0 \\ Z_1 & : |m|+|n| = 0 \end{cases} \quad (7)$$

Substituting (7) into (6), (6) is rewritten as given in (8). The expression in (8) is the proposed Adaptive Centre-Pixel-Weighed Exponential Filter developed in this paper.

$$\boldsymbol{h_{opt}(m,n)} = \begin{cases} \frac{3}{\pi}e^{-2(m^2+n^2)} & : |\boldsymbol{m}|+|\boldsymbol{n}| \neq \boldsymbol{0} \\ \frac{2Z_1}{\pi} & : |\boldsymbol{m}|+|\boldsymbol{n}| = \boldsymbol{0} \end{cases} \quad (8)$$

The variation of the value of the centre - pixel method was applied in this paper because it is simpler and retains the symmetric property of the filter. The filtering technique developed in this work is effective for speckle noise variance ranging from 0% to 30%.

The variable, Z_1 in (8) is dependent on the estimated noise variance present in the noisy image and is given by (9) where δ^2 is the estimated speckle noise.

$$Z_1 = \frac{1}{39.79(\delta^2)^2 - 0.5564\delta^2 + 1.587} \quad (9)$$

The parameter, δ^2 in (9) is the estimated speckle noise variance in a noisy image being filtered. The denominator in the (9) is obtained by applying the Least Squares Method to speckle noise variance estimated using speckle noise variance estimation technique such as the technique developed in [18]. The speckle noise estimation technique developed in this paper is as given in (10). The technique in (9) is the Shifted Mean of Averaging Absolute Derivation technique.

$$\delta^2 = 0.05804(12.3673 - \bar{M})(\bar{M} - 0.1275) \qquad (10)$$

The parameter $\bar{M}$ in (10) is the mean of the speckle noise present in the noisy image and is given in (11).

$$\bar{M} = \frac{1}{MN}\sum_{k=1}^{M}\sum_{l=1}^{N} abs(log_e W(m,n)) \qquad (11)$$

The parameter $log_e W(m,n)$ in (11) is the natural logarithm of the speckle noise present in the image and it is given by (12).

$$log_e W(m,n) = log_e((X(m,n) + \beta) - \frac{1}{M_f N_f}\sum_{j=-R_f}^{R_f}\sum_{k=-C_f}^{C_f} \log_e(X(m,n) + \beta) \qquad (12)$$

2.2 Developed Filter's Response to Noise Variance

The response of the developed filter to change in the noise variance is determined by the parameter, Z1. The actual and estimated values of Z1 as obtained using the Shifted Mean of Averaging Absolute Derivation technique in (10) is given in table 1.

Actual noise variance	Actual values of Z_1	Estimated noise variance	Estimated values of Z_1
0.100	0.5183	0.1199	0.4779
0.150	0.4169	0.1635	0.3907
0.200	0.3260	0.2042	0.3192
0.250	0.2541	0.2509	0.2530
0.300	0.2000	0.3079	0.1928

TABLE 1: The actual and the estimated values of Z1.

Based on table 1, it is observed that the values of the Z1 decrease as the values of noise variance present increases. This shows that the filter responds strongly when the noise variance is low and lightly when the noise variance is high. This property of the developed filter can also be seen in figure1. From the figure, it is seen that Z1 falls as the noise variance increases.

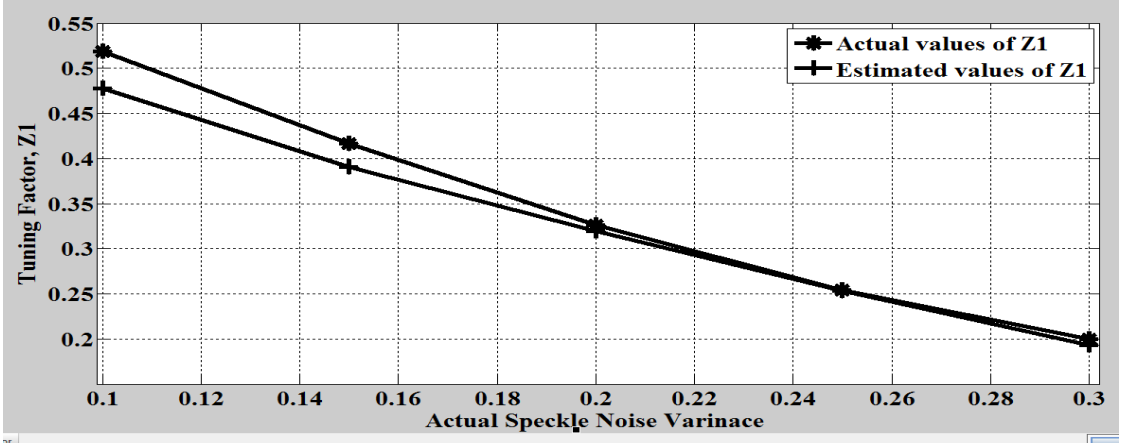


FIGURE 1: Plot of Z1 against actual speckle noise variance.

The figure also shows that the noise estimation technique used in this paper has good performance. This is shown by the closeness of the actual and estimated values of Z1 which implies minimal tuning error.

2.3 Performance Metrics

The effectiveness of the proposed Adaptive Centre-Pixel-Weighed Exponential Filter is estimated using Peak Signal-to-Noise Ratio **(PSNR)** and mean square error **(MSE).** The **PSNR** of a noisy image is the ratio of the maximum power of the signal to the maximum power of the noise distorting the image [19]. For a normalized image, the PSNR can be rewritten as shown in (13) to ensure that PSNR is always positive [20].

$$PSNR = 63 + 10log_{10}\left(\frac{1}{MSE}\right) \quad (13)$$

The PSNR is measured in decibel.

On the other hand, **MSE** is the average of the squared intensity differences between the filtered image pixels and reference image pixels and is given in (14) [21].

$$MSE = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}[\mathrm{Y}(m,n) - \bar{\mathrm{Y}}(m,n)]^2 \quad (14)$$

MSE assumes that the reduction in perceptual quality of an image is directly related to the visibility of the error signal [19].

The performances of the noise variance estimation techniques are quantified using estimation error (ξ) is shown in (15) [22].

$$\xi = \frac{abs(\sigma_n - \delta_n)}{\sigma_n} \times 100 \quad (15)$$

The parameter, σ_n is the actual standard deviation of speckle noise present in the noisy image while δ_n is the estimated standard deviation of speckle noise present in the same image.

2.4. Steps in applying the proposed Algorithm

The steps needed in applying the algorithm developed in this work are as follows:

Step 1: Normalize the noisy image

Step 2: Compute the natural logarithm of the normalized noisy image in step 1.
Step 2: Estimate the speckle noise variance present in the noisy image in step 2.
Step 3: Use the estimated noise variance to compute the value of Z1 and filter kernel.
Step 4: Compute the convolution operation between the natural logarithm of the normalized noisy image obtained in step 2 and the computed filter kernel from step 3.
Step 5: Compute the inverse natural logarithm of the output from step 4 to obtain the filtered image.

3. RESULTS AND DISCUSSION

To effectively analyze the performance of the new technique, the performance of the developed filter and the speckle noise variance estimation technique developed is discussed.

3.1. Discussion of the performance of the developed filter

The performance of the developed model in effectively removing speckle noise from noisy images was compared with other filtering techniques using ten (10) samples of 85x73 CT lung images corrupted by speckle noise of variance ranging from 10% to 30%. The PSNR and MSE values calculated in this paper using (13) and (14) respectively for the three filtering techniques are shown in tables 2 and 3.

Filter	**Noise Variance (%)**					**Average PSNR(dB)**
	10	15	20	25	30	
	PSNR(dB)					
3X3 Binomial filter	68.7851	66.0713	64.0965	62.3410	60.5618	**64.3711**
3X3 Wiener filter	72.9188	70.2405	68.6635	67.3939	66.1122	**69.0658**
3X3 Proposed filter	73.0741	71.3014	70.3109	69.0721	67.6608	**70.2839**

TABLE 2: The PSNR values for the three filtering techniques.

Table 2 shows how the PSNR value for the three filters vary with the speckle noise variance present in the noisy images. From table 2, it is observed that the proposed filter gives the highest average PSNR (**70.2839dB**) compared to Wiener filter (**69.0658dB**) and Binomial filter (**64.3711dB**).

The trend in table 2 is clearly shown in figure 2. From figure 2, it is observed that for all noise levels considered, the proposed filter gives the largest values of PSNR. Figure 2 shows that the difference between the PSNR of the proposed adaptive filter and the PSNRs of wiener and binomial filters widens as the noise variance increases. However, as noise variance decreases, the differences between PSNR of the proposed adaptive filter and the PSNRs of wiener and binomial filters tend to diminish. This is because the proposed filter is the most stable, though, all the filters used give strong filtering at low noise variance and light filtering at high noise variance [23].

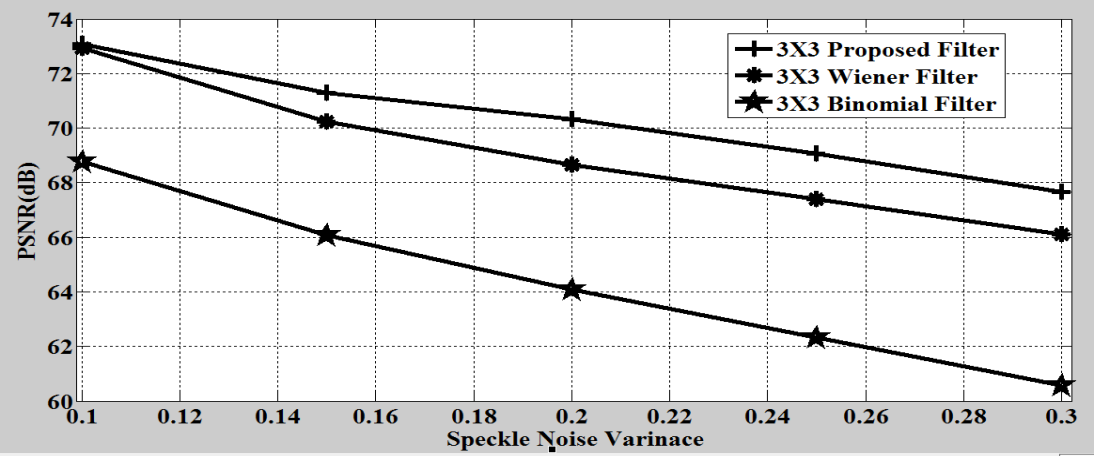


FIGURE 2: Plot PSNR against Speckle Noise Variance for the three filtering techniques.

This shows that the proposed filter is more stable, gives the best result and therefore is the best filter for the preprocessing among the three filters considered.

Filter	Noise Variance (%)					Average MSE
	10	15	20	25	30	
	PSNR(dB)					
3X3 Binomial filter	0.3106	0.5502	0.8574	1.2819	1.8697	**0.97394**
3X3 Wiener filter	0.1493	0.2489	0.3620	0.4768	0.6002	**0.36744**
3X3 Proposed filter	0.1339	0.2025	0.2691	0.3522	0.4539	**0.28232**

TABLE 3: The PSNR values for the three filtering techniques.

Considering table 3, it is also observed that the proposed filter gives the lowest value of MSE (**0.28232**) compared to Wiener filter (**0.36744**) and Binomial filter (**0.97394**). The trend in table 3 is clearly shown in figure3. Figure 3 shows that the difference between the MSE of the proposed adaptive filter and the MSEs of wiener and binomial filters widens as the noise variance increases. However, as noise variance decreases, the differences between MSE of the proposed adaptive filter and the MSEs of wiener and binomial filters tend to diminish. This is because the proposed filter is the most stable, though, all the filters used give strong filtering at low noise variance and light filtering at high noise variance [23].

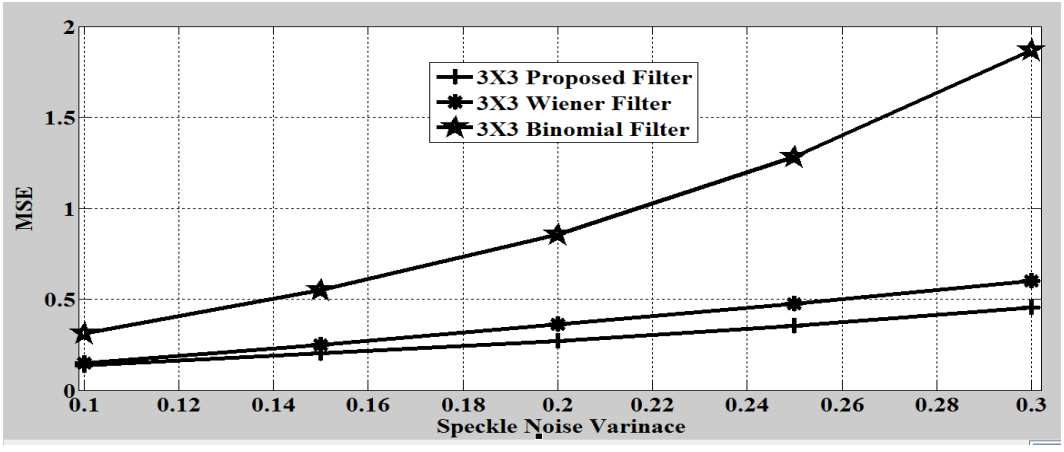


FIGURE 3: Plot MSE against Speckle Noise Variance for the three filtering techniques.

From figure 3, it is observed that for all noise levels considered, the proposed filter gives the lowest values of MSE and is more stable compared to wiener and binomial filters.

3.2. Results of noise estimation technique

The performance of the speckle noise variance estimation technique developed in this paper was compared with Median of Median Absolute Derivative Technique developed in [18] using ten (10) samples of 85x73 CT lung images corrupted by speckle noise of variance ranging from 10% to 30%. The average estimated speckle noise variances obtained in the analysis using the two (2) techniques are shown in table 4. The average speckle noise estimation errors calculated from table 1 using (15) for the two (2) techniques are as shown in table 5.

Noise Estimation Techniques	**Actual Noise Variance**				
	0.10	0.15	0.20	0.25	0.30
	Estimated noise variance				
Developed Estimation Technique	0.1199	0.1635	0.2042	0.2509	0.3079
Median of Median Absolute Derivative Technique [18]	0.0828	0.1182	0.1471	0.1892	0.2160

TABLE 4: Average estimated speckle noise variance for the two noise estimation techniques.

The estimated speckle noise variances using the two (2) techniques are shown in table 4. From this table, it is seen that the noise variance estimated using the developed estimation technique is closer to the actual noise variance present compared to median of median absolute derivative technique for all the noise levels considered. Looking at figure 4, it is observed that the developed estimation technique has the higher graph showing it has higher accuracy. The estimated variance of the developed estimation technique becomes closer to the actual noise variance as the actual noise variance present increases as shown in figure 4. However, at very low noise level, the estimated variance from each technique becomes closer. This is because the developed estimation technique is more stable, though, both estimation technique used give strong filtering at low noise variance and light filtering at high noise variance[23].

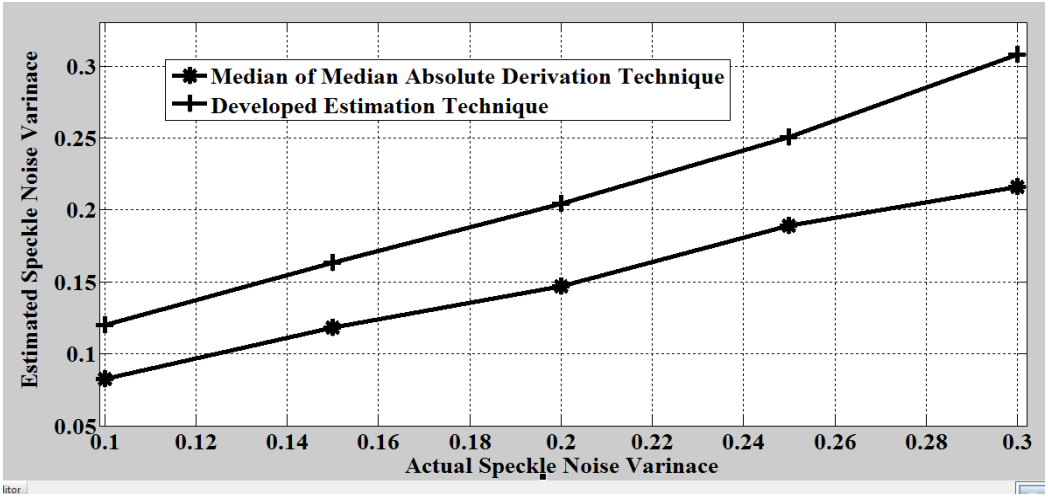


FIGURE 4: Plot of average estimated speckle noise variance against actual noise variance for the two noise estimation techniques.

Based on the plot in figure 4, it is concluded that the developed estimation technique has better average speckle noise variance estimation accuracy for all the noise variance range considered. Also, it can be observed that the estimated noise variance for the developed estimation technique becomes more accurate as the actual noise variance present increases.

Noise Estimation Technique	Actual Noise Variance					Average Noise Estimation Error (%)
	0.10	0.15	0.20	0.25	0.30	
	Noise Estimation Error (%)					
Developed Estimation Technique	9.5080	4.3999	1.0322	0.1818	1.3048	3.2853
Median of Median Absolute Derivative Technique [18]	9.0000	11.2457	14.2416	13.0005	15.1570	12.5289

TABLE 5: Average speckle noise estimation error for the four noise estimation techniques.

The difference in performances among the two estimation techniques considered is clearly seen in table 5. From table 5, it is observed that the developed estimation technique has lower estimation error except at 10% noise variance showing that it has better estimation accuracy.

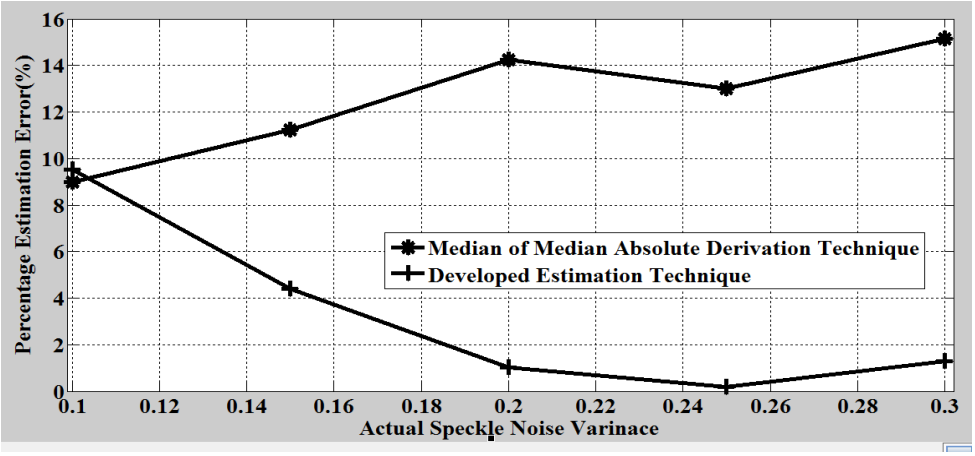


FIGURE 5: Plot of average speckle noise estimation error against actual noise variance for the four noise estimation techniques.

Looking at figure 5, it is observed that the developed estimation technique has the lower graph showing it has higher accuracy. The estimation error of the developed estimation technique decreases as the variance of noise present increases as shown in figure 5 and it falls sharper than the other technique. However, at very low noise level, the estimation error of the two techniques becomes approximately the same. This shows that the developed estimation technique is more stable and has higher accuracy.

4. CT LUNG IMAGES USED IN THE PAPER

The CT images used in the paper are shown in figure6. The images are of sizes 85X73. Figure 7 shows the images when each is corrupted by 30% speckle noise.

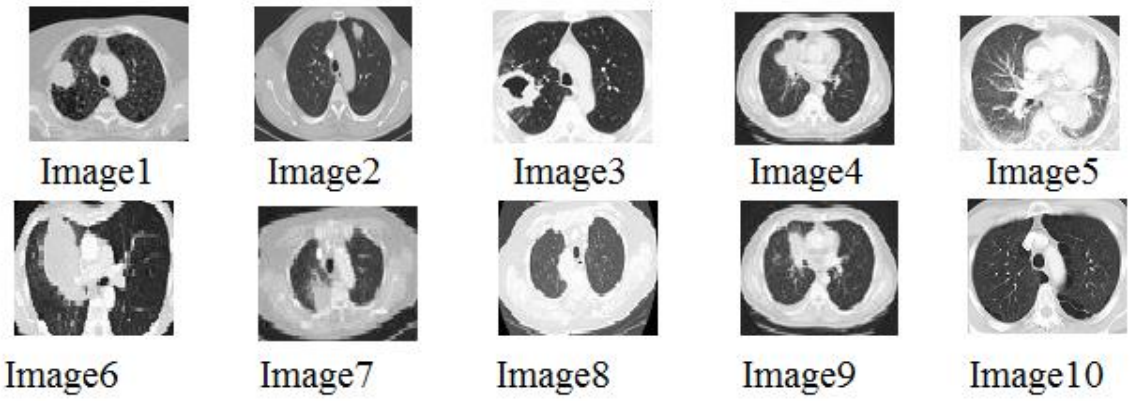


FIGURE 6: Noiseless images used in the paper.
Source: http://radiopaedia.org

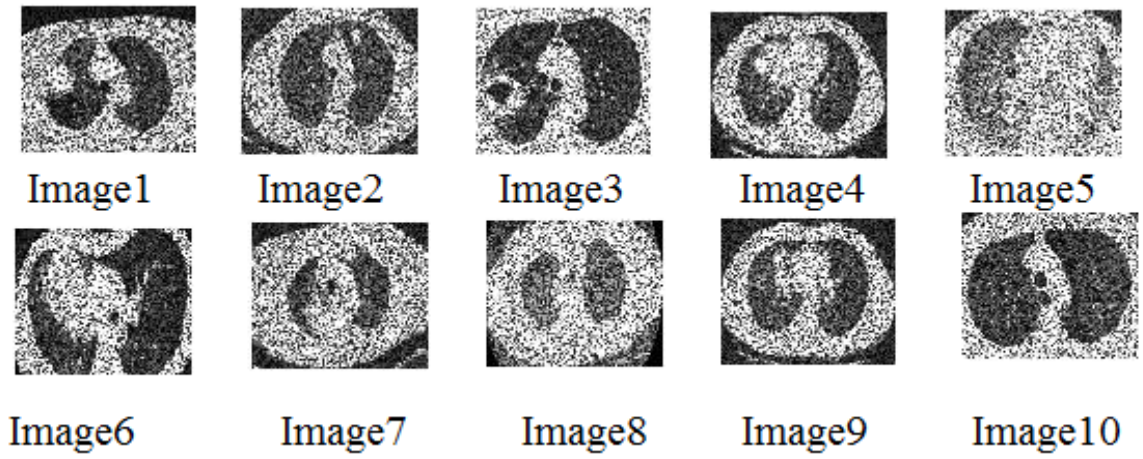


FIGURE 7: Unfiltered images corrupted by 30% noise.

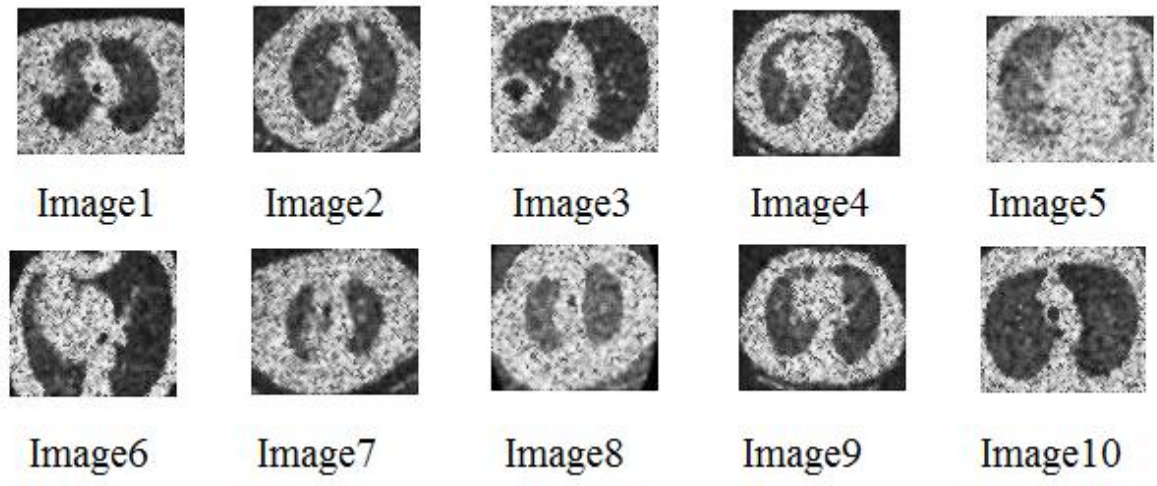


FIGURE 8: 30% noisy Images filtered using Wiener Filter.

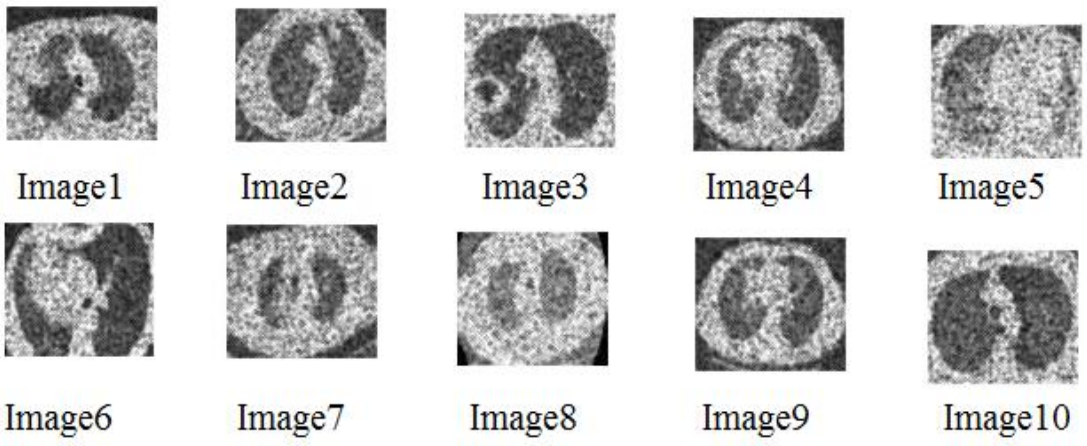


FIGURE 9: 30% noisy Images filtered using proposed Filter.

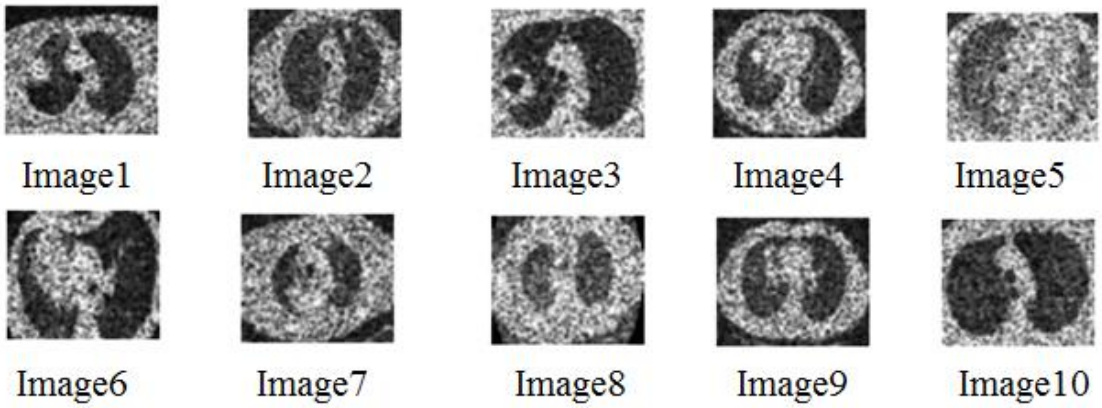


FIGURE 10: 30% noisy Images filtered using Binomial Filter.

Figures 8, 9 and 10 show the noisy images in figure7 filtered using Wiener filter, proposed filter and Binomial filter respectively. Based on visual observation, it is seen that images filtered using proposed filter has best visual details compared to others. This shows that the proposed filter gives the best filtering performance among the three filters used.

5. CONCLUSION

In this paper, an Adaptive Centre-Pixel-Weighed Exponential Filter was developed. The developed technique filters speckle in a noisy CT image with an average PSNR as high as **70.2839dB** for the noise variance ranging from 10% to 30%. The new technique out performs other techniques used to verify its performance, including the *Wiener filter* which gave an average MSE of **69.0658dB** and Binomial filter which gave an average MSE of **64.3711dB** for the same noise range. Since the filter developed in this paper is linear, it blurs the edges of the filtered images.

In further research, an attention will be given to an Adaptive alpha-trimmed Centre-Pixel-Weighed Exponential Filter. This filter is proposed combines the properties of linear and nonlinear filters which ensure effective filtering without edge blurring.

6. ACKNOWLEDGEMENT

We thank immensely Conquest Medical Imaging, Trans Ekulu, Enugu for their invaluable technical contribution during this research.

7. REFERENCES

[1] A. B. Sankar and D. Kumar, “Performance Study of Various Adaptive Filter Algorithms for Noise Cancellation in Respiratory Signals Therefore , they can detect shape variations in the ensemble and thus they can obtain a better signal estimation . This paper focuses on (i) Model Resp,” Sgnal Processing:An International Journal, vol. 4, no. 5, pp. 267–278.

[2] V. Anand, S. Shah, and S. Kumar, “Intelligent Adaptive Filtering For Noise Cancellation,” International Journal of Advanced Research in Electrical,Electronics and Instrumentation Engineering, vol. 2, no. 5, pp. 2029–2039, 2013.

[3] M. Juneja and R. Mohana, “An Improved Adaptive Median Filtering Method for Impulse Noise Detection,” International Journal of Recent Trends in Engineering, vol. 1, no. 1, p. 274, 2009.

[4] A. Manseur, D. Berkani, and A. Mekhmoukh, “Adaptive filtering using Higher Order Statistics (HOS),” International Journal of Computer Science Issues, vol. 9, no. 2, p. 477, 2012.

[5] S. Mann, “Comparametric Equations with Practical Applications in Quantigraphic Image Processing,” IEEE transactions on image processing, vol. 9, no. 8, pp. 1389–1406, 2000.

[6] G. S. S, U. S. Nagar, and A. Safir, “REMOVAL OF SPECKLE NOISE FROM EYE IMAGES,” International Journal of Advanced Engineering Technology, vol. 2, no. 1, p. 3, 2011.

[7] S. Solbø and T. Eltoft, “λ-WMAP: a statistical speckle filter operating in the wavelet domain,” International Journal of Remote Sensing, vol. 25, no. 5, pp. 1019–1036, Mar. 2004.

[8] S. Abramov, V. Zabrodina, V. Lukin, B. Vozel, K. Chehdi, and J. Astola, “Methods for Blind Estimation of the Variance of Mixed Noise and Their Performance Analysis,” INTECH Numeric Analysis, 2011.

[9] S. Abramov, V. Abramova, V. Lukin, N. Ponomarenko, B. Vozel, K. Chehdi, K. Egiazarian, and J. Astola, “Methods for Blind Estimation of Speckle Variance in SAR Images : Simulation Results and Verification for Real-Life Data,” INTECH Computational and numerical Simulations, 2010.

[10] B. S. Kramer, C. D. Berg, D. R. Aberle, and P. C. Prorok, “Lung cancer screening with low-dose helical CT: results from the National Lung Screening Trial (NLST).,” Journal of medical screening, vol. 18, no. 3, p. 110, Jan. 2011.

[11] G. Ilango, “- Neighbourhood Median Filters to Remove Speckle Noise from CT – Images,” International Journals of Applied Information Systems, vol. 4, no. 10, pp. 40–46, 2012.

[12] A. Ozcan, A. Bilenca, A. E. Desjardins, B. E. Bouma, and J. Tearney, “Speckle Reduction in Optical Coherence Tomography Images using Digital Filters,” Journal of the Optical Society of America., vol. 24, no. 7, pp. 1901–1910, 2009.

[13] A. Rafiee, M. H. Moradi, and M. R. Farzaneh, “Novel genetic-neuro-fuzzy filter for speckle reduction from sonography images.,” Journal of digital imaging, vol. 17, no. 4, pp. 292–300, Dec. 2004.

[14] K. Karthikeyan, “Speckle Noise Reduction of Medical Ultrasound Images using Bayesshrink Wavelet Threshold,” vol. 22, no. 9, pp. 8–14, 2011.

[15] A. Singhal and M. Singh, “Speckle Noise Removal and Edge Detection Using Mathematical Morphology,” International Journal of Soft Computing and Engineering, vol. 1, no. 5, pp. 146–149, 2011.

[16] G. Padmavathi, P. Subashini, M. M. Kumar, and S. K. Thakur, “Comparison of Filters used for Underwater Image Pre-Processing,” International Journal of Computer Science and Network Security, vol. 10, no. 1, p. 2, 2010.

[17] M. C. Eze and C. C. Osuagwu, “Evaluation of the Performances of Homomorphic and Non-homomorphic Speckle Noise Filtering Techniques,” International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS), vol. 10, no. 1, pp. 50–55, 2014.

[18] A. Cameron, D. Lui, A. Boroomand, J. Glaister, A. Wong, and K. Bizheva, “Stochastic speckle noise compensation in optical coherence tomography using non-stationary spline-based speckle noise modelling.,” Biomedical optics express, vol. 4, no. 9, pp. 1769–85, Jan. 2013.

[19] A. K. E. Isaac, “T he Performance of Filters for Reduction of Speckle Noise in SAR Images of the Coast of the Gulf of Guinea,” International Journal of Information Technology, Modeling and Computing, vol. 1, no. 4, pp. 43–52, 2013.

[20] L. M. Davala, “Directional Linear Minimum Mean Square- Error Estimation in Color Demosaicking,” vol. 2, no. 4, pp. 171–183, 2012.

[21] A. Stella and B. Trivedi, “Implementation of Order Statistic Filters on Digital Image and OCT Image : A Comparative Study,” International Journal of Modern Engineering Research, vol. 2, no. 5, pp. 3143–3145, 2012.

[22] K. Kutty and S. Ojha, “A Generic Transfer Function based Technique for Estimating Noise from Images,” International Journal of Computer Applications, vol. 51, no. 10, pp. 26–32, 2012.

[23] K. Balakrishnan, K. Sunil, A. V Sreedhanya, and K. P. Soman, “Effect Of Pre-Processing On Historical Sanskrit Text Documents,” International Journal of Engineering Research and Applications, vol. 2, no. August, pp. 1529–1534, 2012.

Perceptual Weights Based On Local Energy For Image Quality Assessment

Sudhakar Nagalla *sudhakar.nagalla@becbapatla.ac.in*
Department of Computer Science and Engineering
Bapatla Engineering College
Bapatla, 522102, India

Ramesh Babu Inampudi *rinampudi@hotmail.com*
Department of Computer Science and Engineering
Acharya Nagarjuna University
Guntur, 522510, India

Abstract

This paper proposes an image quality metric that can effectively measure the quality of an image that correlates well with human judgment on the appearance of the image. The present work adds a new dimension to the structural approach based full-reference image quality assessment for gray scale images. The proposed method assigns more weight to the distortions present in the visual regions of interest of the reference (original) image than to the distortions present in the other regions of the image, referred to as perceptual weights. The perceptual features and their weights are computed based on the local energy modeling of the original image. The proposed model is validated using the image database provided by LIVE (Laboratory for Image & Video Engineering, The University of Texas at Austin) based on the evaluation metrics as suggested in the video quality experts group (VQEG) Phase I FR-TV test.

Keywords: Image Quality, HVS, Full-reference Quality Assessment, Perceptual Weights.

1. INTRODUCTION

Any image processing system should be aware of the impacts of processing on the visual quality of the resulting image. Numerous algorithms for image quality assessment (IQA) have been investigated and developed over the last several decades. The objective image quality measurement seeks to measure the quality of images algorithmically. Objective image quality metrics can be classified as full-reference in which the algorithm has access to the original (considered to be distortion free) image, no-reference in which the algorithm has access only to the distorted image and reduced-reference in which the algorithm has partial information regarding the original image. A comprehensive review of research and challenges in image quality assessment is presented in [1].

In [2], a number of simple statistical image quality metrics based on numerical errors are compared for gray scale image compression. These metrics include average difference, maximum difference, absolute error, mean square error (MSE), peak MSE, Laplacian MSE, histogram and Hosaka plot. It is observed that although some numerical measures correlate well with the human response for a specific compression technique, they are not found to be reliable for evaluation across various methods of compression. The most widely adopted statistical feature is the Mean Squared Error (MSE). However, MSE and its variants may not correlate well with subjective quality measures because human perception of image distortions and artifacts is unaccounted for. A detailed discussion on MSE is presented by Girod [3].

Most HVS based quality assessment metrics share an error-sensitivity based paradigm [4], which aims to quantify the strength of the errors between the reference and the distorted signals in a

perceptually meaningful way. A well-known method, Visible Differences Predictor (VDP) [5], Lubin's algorithm [6], Teo and Heeger's metric [7], a perceptual image quality metric named information mean square error (IMSE) proposed by David Tompa et al. [8], a measure of perceptual image quality of Westen et al. [9], a comprehensive distortion metric for digital color images presented by Stefan Winkler [10], an image quality metric using contrast signal-to-noise ratio (CSNR) by Susu Yao et al. [11], image quality metric named visual information fidelity (VIF) introduced by Sheikh and Bovik [12] belong to this category. The rest of the paper is organized as follows. Section 2 explains structural similarity measure, Section 3 presents local energy model for detecting image features, Section 4 explains weighting of structural similarity indices and formulation of Perceptual Structural Similarity index. Section 5 presents the results followed by conclusions.

2. STRUCTURAL SIMILARITY MEASURE

One distinct feature that makes natural image signals different from a "typical" image randomly picked from the image space is that they are highly structured and the signal samples exhibit strong dependencies amongst themselves. These dependencies carry important in-formation about the structures of objects in the visual scene. An image quality metric that ignores such dependencies may fail to provide effective predictions of image quality. Structural similarity based methods [13, 14] of image quality assessment claim to account for such dependencies in assessing the image quality. In [14] a more generalized and stable version of the universal quality index was proposed named as Structural SIMilarity quality measure (*SSIM*).

Let x and y be two discrete non-negative signals where $x = \{x_i \mid i = 1, 2...N\}$ and $y = \{y_i \mid i = 1, 2...N\}$ are aligned with each other (e.g. two image patches extracted from the same spatial location of original image and distorted image being compared). Let $\mu_x, \mu_y, \sigma_x, \sigma y, \sigma_{xy}$ represent mean intensity of signal x, mean intensity of signal y, standard deviation of x, standard deviation of y, and covariance between x and y respectively. The Structural Similarity measure between the image patches is defined in (1), where C_1 and C_2 are small constants introduced to avoid instability when the denominator is close to zero.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{1}$$

Let X and Y be the two images being compared. A local moving window approach is followed, to compute $SSIM(X, Y)$. The window moves pixel-by-pixel from the top left corner to the bottom right corner of the image. In each step, the local statistics and $SSIM(x_j, y_j)$ index are calculated using (1) within the local window j. The $SSIM$ index between X and Y is defined in (2) where N_S is the number of local windows in the image, and $W_j(x_j, y_j)$ is the weight given to the j-th window of the image. If all the local regions in the image are equally weighted, then $W_j(x_j, y_j) = 1$. This results in the mean $SSIM(MSSIM)$ measure employed in [14].

$$SSIM(X, Y) = \frac{\sum_{j=1}^{N_s} W_j(x_j, y_j).SSIM(x_j, y_j)}{\sum_{j=1}^{N_s} W_j(x_j, y_j)} \tag{2}$$

It may be noted that the $MSSIM$ algorithm gives equal importance to distortions for all local regions of the image. Wang et al. [14] suggest that the performance of $SSIM$ can be improvised by weighting the local $SSIM$ indices. They also suggest that the prior knowledge about the

importance of different regions in the image if available can be converted into a weighting function. A variety of such approaches can be found in [15]-[20].

Studies of visual attention and eye movements [6, 21, 22] have shown that humans attend to few areas in the image. Even though unlimited viewing time is provided, subjects will continue to focus on few areas rather than scan the whole image. These areas are often highly correlated amongst different subjects, when viewed in the same context. In order to automatically determine the parts of an image that a human is likely to attend to, we need to understand the operation of human visual attention and eye movements. In [23], many algorithms for defining Visual regions of interest were evaluated in comparison with eye fixations. The present work adopts the local energy model to identify feature rich local regions which are normally attended to by humans and to define a weighting function that is proportional to feature richness of the region. The weighting function is used in (2) to define the Perceptual Structural SIMilarity index $PSSIM^e$ proposed in this paper.

3. LOCAL ENERGY MODELING FOR FEATURE DETECTION

The local energy model of feature detection postulates that features are perceived at points of maximum phase congruency in an image. Venkatesh and Owens [24] show that points of maximum phase congruency can be calculated equivalently by searching for peaks in the local energy function. The calculation of energy from spatial filters in quadrature pairs has been central to the models of human visual perception proposed by Heeger [25], Adelson and Bergen [26]. Local frequency and, in particular, phase information in signals are of importance in calculating local energy. To preserve phase information, linear-phase filters must be used. That is, one must use non orthogonal filters that are in symmetric/antisymmetric quadrature pairs. In this work, the approach of Morlet et al. [27] is followed with a modification in the usage of filters. Logarithmic Gabor functions [28, 29] are used instead of Gabor filters as the maximum bandwidth of a Gabor filter is limited to approximately one octave and Gabor filters are not optimal if one needs broad spectral information with maximal spatial localization.

Field [28] suggests that natural images are better coded by filters that have Gaussian transfer functions when viewed on the *logarithmic* frequency scale. Firstly, log-Gabor functions, by definition, always have no DC component, and secondly, the transfer function of the log Gabor function has an extended tail at the high frequency end. Field's studies of the statistics of natural images indicate that natural images have amplitude spectra that fall off at approximately inverse of the frequency. To encode images having such spectral characteristics one should use filters having spectra that are similar. Field suggests that log Gabor functions, having extended tails, should be able to encode natural images more efficiently than, say, ordinary Gabor functions, which would over-represent the low frequency components and under-represent the high frequency components in any encoding. Another point in support of the log Gabor function is that it is consistent with measurements on mammalian visual systems which indicate we have cell responses that are symmetric on the log frequency scale.

The local energy function is computed using log-Gabor filters in 4 scales with center frequencies of 1/3 cycles/pixel, 1/6 cycles/pixel, 1/12 cycles/pixel and 1/24 cycles/pixel and 6 orientations at 0^0(horizontal), 30^0, 60^0, 90^0(vertical), 120^0, and 150^0. The following discussion [29] is made for a specific orientation θ of the filter. If $I(x)$ denotes the image signal and M_e and M_o denote the even-symmetric (cosine) and odd-symmetric (sine) filters at a scale n, the respective responses e_n and o_n of each quadrature pair of filters can be represented by the vector,

$$\left[e_n(x), o_n(x)\right] = I(x) * M_n^e, I(x) * M_n^o \tag{3}$$

The amplitude $A_n(x)$ and phase $\phi_n(x)$ of the transform at any given scale is given by

$$A_n(x) = \sqrt{e_n(x)^2 + o_n(x)^2} \tag{4}$$

$$\phi_n(x) = atan2\left(e_n(x), o_n(x)\right) \tag{5}$$

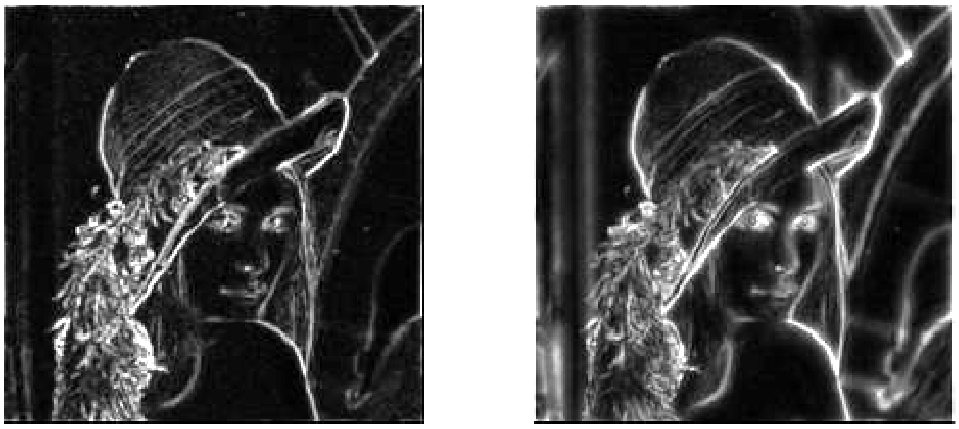
At each point x in a signal, an array of these response vectors is obtained, one vector for each scale of filter of the chosen orientation. These response vectors form the basis of localized representation of the signal, and they can be used in calculating the resultant local energy vector at point x. The design of the filter bank needs to be such that the transfer function of each filter overlaps sufficiently with its neighbors so that the sum of all the transfer functions forms a relatively uniform coverage of the spectrum. If the local energy should accurately represent the feature strength at point x, then a broad range of frequencies in the signal are to be retained. The local energy at point x of the image $E_\theta(x)$ for a given orientation θ can be calculated from $F(x)$ which can be formed by summing the even filter convolutions over all scales and $H(x)$ which can be estimated by summing the odd filter convolutions over all scales given by

$$F(x) = \sum_n e_n(x) \tag{6}$$

$$H(x) = \sum_n o_n(x) \tag{7}$$

$$E_\theta(x) = \sqrt{F(x)^2 + H(x)^2} \tag{8}$$

Figure 1(a) and Figure 1(b) show the normalized maps of the local energy function of the *Lena* image considering different ranges of frequencies. Figure 1(a) is the result of considering frequencies larger than 0.2*pixels/cycle*. Figure 1(b) is the result of considering the complete set of frequencies. One can observe that the latter makes a clear distinction among the significance of features than the former. The former shows that the majority of features are equally important while the latter shows a broad scale distinction.



(a) Energy function considering 2 scales **(b)** Energy function considering 4 scales

FIGURE 1: Energy function of *Lena* Image.

At each location in the image, the weighted local oriented energy $E_\theta(x)$ in each orientation is calculated, and the sum over all orientations $E(x)$ is computed. The following algorithm illustrates the above steps.

Let $I(x)$ be the original image;
$c \leftarrow 0.5$;
$\varepsilon \leftarrow .0001$;
$\gamma \leftarrow 10$;
$E(x) \leftarrow [\ ]$;
for each orientation θ do
    $sum\_e(x) \leftarrow [\ ]$;
    $sum\_o(x) \leftarrow [\ ]$;
    $sum\_A_n(x) \leftarrow [\ ]$;
    for each scale n do
        compute $e_n(x), o_n(x)$ as in (3);
        compute $A_n(x)$ using as in (4);
        $sum\_e(x) \leftarrow sum\_e(x) + e_n(x)$;
        $sum\_o(x) \leftarrow sum\_o(x) + o_n(x)$;
        $sum\_A_n(x) \leftarrow sum\_A_n(x) + A_n(x)$;
        if first scale then
            $A_{max}(x) \leftarrow A_n(x)$;
        else
            $A_{max}(x) \leftarrow max(A_n(x), A_{max}(x))$;
        end if
    end for
    compute $E_\theta(x)$ as in (8);
    $E(x) \leftarrow E(x) + E_\theta(x)$;
end for

Figure 2 shows the perceptual map of *Lena* image based on local energy. The perceptual importance map assigned more weights (more bright) to image features in face, hair, hat and background. It can also be observed that the perceptual weights assigned to the features are well distributed.

FIGURE 2: Perceptual map of Lena image.

4. LOCAL ENERGY WEIGHTED STRUCTURAL SIMILARITY

We assume that the width w and the height h of the original image X and the distorted image Y are exact multiples of 9. If the size does not conform to these dimensions the images are cropped on all sides so that minimum amount of details is lost. This requirement comes from the fact that $SSIM$ indices are computed in non-overlapping 9×9 regions. These regions in X and Y are referred to as x_{ij} and y_{ij} respectively.

The computation of perceptual weights of local regions in the original image begins with the log-Gabor decomposition of the image. Log-Gabor filters with 4 scales and 6 orientations are used for this purpose. The algorithm presented earlier explains the computation of local maxima $E(x)$ of local energy function at each pixel location x for the original image X. Let the matrix $E(x)$ be divided into non-overlapping blocks of size 9×9. Each resulting block in corresponds to a non-overlapping block x_{ij} of the original image X, $1\le i\le m=h/9$ and $1\le j\le n=w/9$. The local maxima values present in each such block are summed up to obtain the local maxima for the block x_{ij}. The resulting matrix is normalized and these values are proposed as the perceptual weights of the corresponding to 9×9 local regions which will be indicative of the human attention the regions call for. Let the resulting matrix be E of size $m\times n$.

The structural similarity index between corresponding blocks of X and Y of is computed using (1) to obtain the matrix $SSIM$ of size $m\times n$. The Weighted Structural SIMilarity measure $PSSIM^{e}$ between X and Y is calculated using (9). $PSSIM^{e}$ indicates the quality of distorted image on a scale of 0 to 1, where a value of 1 indicates that the images are identical.

$$PSSIM^{e}=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}[E]\,[SSIM]}{\sum_{i=1}^{m}\sum_{j=1}^{n}[E]} \tag{9}$$

5. RESULTS

The proposed models are validated using the image database provided by *LIVE (Laboratory for Image & Video Engineering, The University of Texas at Austin)* [30]. The psychometric study for the development of the database contained 779 images distorted using five different distortion types and more than 25,000 human image quality evaluations.

The distorted image database consists of twenty-nine high resolution 24-bits/pixel RGB color images (typically 768 × 512). The distortions include white Gaussian noise, Gaussian blur, simulated fast fading Rayleigh (wireless) channel, JPEG compression and JPEG2000 compression and with each type the perceptual quality covered the entire quality range. Observers are asked to provide their perception of quality on a continuous linear scale that was divided into five equal regions marked with adjectives "Bad", "Poor", "Fair", "Good", and "Excellent". About 20-29 human observers rated each image. The raw scores for each subject are converted to difference scores (between test and reference images) and then converted to Z-scores , scaled back to 1-100 range, and finally a Difference Mean Opinion Score (*DMOS*) value for each distorted image is computed.

The proposed models and other models used for comparison are validated using the *LIVE* image database based on the evaluation metrics as suggested in the video quality expert's group (VQEG) Phase I FR-TV test [31]. A nonlinear regression model is fitted to the *DMOS* values in the database, and calculated image quality metric values (*IQ)* of the distorted images for each

distortion and for each quality assessment model used in comparison. The following 4 parameter logistic function is used in the present work.

$$DMOS_p = 1/\left(1 + exp\left(-b*\left(IQ - c\right)\right)\right) + d \quad (10)$$

The nonlinear regression function is used to transform the set of *IQ* values to a set of predicted *DMOS* values, *DMOSp*, which are then compared with the actual *DMOS* values from the subjective tests. The Correlation Coefficient (*CC*), the Mean Absolute Error (*MAE*), and the Root Mean Squared Error (*RMSE*) between the subjective scores *DMOS* and predicted scores *DMOSp* are evaluated as measures of prediction accuracy. The prediction consistency is quantified using the Outlier Ratio (*OR*), which is defined as the percentage of the number of predictions outside the range of 2 times the standard deviation of errors between *DMOS* and *DMOSp*. Finally, the prediction monotonicity is measured using the Spearman Rank-Order-Correlation Coefficient (*ROCC*).

Table 1 shows the results for the proposed method to estimate the image quality index $PSSIM^e$ in comparison with the three image quality assessment models *PSNR* (Peak Signal-to-Noise Ratio) and the Structural SIMilarity quality measure *SSIM* [14]. The Correlation Coefficient (*CC*), the Mean Absolute Error (*MAE*), and the Root Mean Squared Error (*RMSE*) values for the three assessment models considered prove that the prediction accuracy of the proposed model is superior to the others. The values of the Spearman Rank-Order-Correlation Coefficient (*ROCC*) indicate that the proposed model correlates well with the human judgment. However, the values of Outlier Ratio (*OR*) are inferior marginally when compared with the other two models. This can be attributed to the fact that the human judgment is impulsive in case of images with higher levels of distortion in contrast to the computational algorithms for image quality assessment.

Figure 4 shows the scatter plots for different distortions in which each data point represents true mean opinion score (*DMOS*) versus the predicted score of one test image by the proposed method after the nonlinear mapping.

Model	CC	ROCC	MAE	RMSE	OR
White Noise					
PSNR	0.922	0.938	4.524	6.165	0.055
SSIM	0.94	0.914	4.475	5.459	0.027
$PSSIM^e$	0.967	0.958	3.344	4.073	0.034
Gaussian Blur					
PSNR	0.744	0.725	8.395	10.501	0.034
SSIM	0.947	0.940	3.992	5.027	0.034
$PSSIM^e$	0.971	0.966	3.02	3.711	0.041
Fast Fading					
PSNR	0.857	0.859	6.383	8.476	0.068
SSIM	0.956	0.945	3.806	4.799	0.055
$PSSIM^e$	0.965	0.964	3.471	4.315	0.041
JPEG Compression					
PSNR	0.842	0.828	6.636	8.622	0.062
SSIM	0.891	0.863	5.386	7.236	0.057
$PSSIM^e$	0.916	0.888	4.727	6.393	0.062
JPEG2000 Compression					
PSNR	0.859	0.851	6.454	8.269	0.059
SSIM	0.899	0.894	5.687	7.077	0.023
$PSSIM^e$	0.931	0.926	4.754	5.876	0.041

.

TABLE 1: Performance comparison of image quality assessment models.

6. CONCLUSION

In this paper, we proposed a full-reference perceptual image quality metric for gray scale images based on structural approaches unified with perceptual regions humans attend to in a natural image. The local energy model was used to indentify feature rich regions in natural images and to formulate a weighting function for distortions a given image.

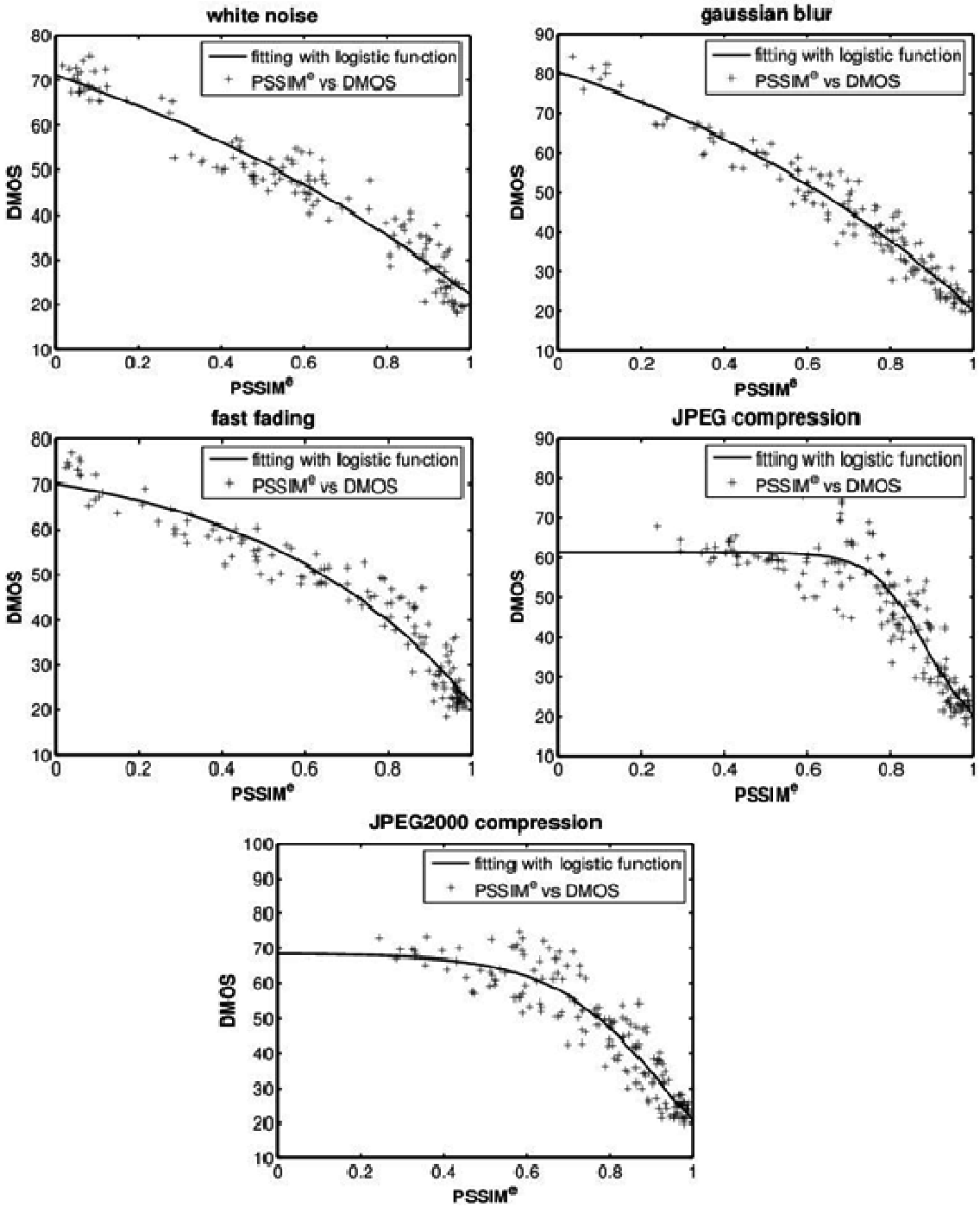


FIGURE 3: Scatter plots between *DMOS* and *PSSIM*^{e} for different image distortions.

The proposed models and other models used for comparison are validated using the metrics suggested by VQEG. The results prove that the performance of the metric is close to the human (subjective) judgment. The metrics are also found to be superior in performance in comparison with the other models of quality assessment considered. The metrics are generic and they are applicable to a wide variety of image distortions like white noise, Gaussian blur, fast fading, and different compression artifacts.

7. FUTURE RESEARCH

The present work formulated a framework for the image quality assessment is evolved in which the model of identifying perceptual regions and the process of computing the image distortions are independent. Such a frame work facilitates a modular approach so that the individual models can be modified and optimized independently. As the framework for formulating perceptual quality metric is flexible, different combinations of distortion modeling and perceptual region modeling can be explored. In the present work, the notion of perceptual regions is used in image quality assessment. It can be extended to other possible areas of image processing like face recognition and watermarking.

8. REFERENCES

[1] Damon M. Chandler, "Seven challenges in image quality assessment: Past, present, and future research," ISRN Signal Processing, Article ID 905685, [Online]. Available:http://dx.doi.org/10.1155/2013/905685, 2013.

[2] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance", IEEE Transactions on Communications, vol. 43, Dec. 1995, pp. 2959-2965.

[3] B. Girod, "What's wrong with mean-squared error," in Digital Images and Human Vision, A. B. Watson, Ed., MIT press, 1993, pp. 207-220.

[4] Z.Wang, H.R. Sheikh, and A.C. Bovik, Objective video quality assessment: The Handbook of Video Databases: Design and Applications, B.Furht and O. Marques, Eds., CRC press, 2003.

[5] S. Daly, "The visible differences predictor: an algorithm for the assessment of image fidelity," in Digital Images and Human Vision, A. B. Watson, Ed., MIT press, 1993, pp.197-206.

[6] J.Lubin, "A visual discrimination model for image system design and evaluation," in Visual Models for Target Detection and Recognition, E.Peli, Ed., World Scientific Publishers, Singapore, 1995, pp. 245-283.

[7] P. C. Teo and D. J. Heeger, "Perceptual image distortion," Proc. SPIE, vol. 2179, 1994, pp. 127-141,

[8] David Tompa, John Morton, and Ed Jernigan, "Perceptually based image comparison", International conference on image processing, ICIP, vol. 1, 2000, pp. 489-492.

[9] S. Western, K.L. Lagendijk, and J. Biemond, "Perceptual image quality based on a multiple channel hvs model," ICASSP, vol. 4, 1995, pp. 2351-2354..

[10] Stefan Winkler, "A perceptual distortion metric for digital color images," Proc. International Conference on Image Processing, ICIP98, vol. 3, Oct. 1998, pp. 399-403.

[11] Susu Yao, Weisi Lin, EePing Ong, and Zhongkang Lu, "Contrast signal -to-noise ratio for image quality assessment," Proc. IEEE International Conference on Image Processing, ICIP 2005, vol. 1, Sept. 2005, pp. 397-400.

[12] Hamid Rahim Sheikh and Alan C.Bovik, "Image information and visual quality," IEEE Transactions on Image Processing, vol. 15, Feb. 2006.

[13] Z.Wang and A.C. Bovik, “A universal image quality index,” IEEE Signal Processing Letters, vol. 9, pp. 81-84, Mar. 2002.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Si-moncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Transactions on Image Processing, vol. 13, pp. 600-612, Apr. 2004.

[15] D.Venkata Rao, N.Sudhakar, B.R.Babu, and L.Pratap Reddy, “An image quality assessment technique based on visual regions of interest weighted structural similarity,” ICGST international journal on Graphics Vision and Image Processing, vol. 6, pp. 69-75, Sept. 2006.

[16] D.Venkata Rao, N.Sudhakar, B.R.Babu, and L.Pratap Reddy, “Image quality assessment complemented with visual regions of interest,” ICCTA 2007-Proc. International Conference on Computing: Theory and Applications, IEEE Computer Society Press, vol. 2, Mar. 2007, pp. 681-687.

[17] D.Venkata Rao and L.Pratap Reddy, “Image quality assessment based on perceptual structural similarity,” in Pattern Recognition and Machine Intelligence, ser. Lecture Notes in Computer Science, Springer-Verlag, vol. 4815, Dec. 2007, pp. 87-94.

[18] D.Venkata Rao and L. Pratap Reddy, “Weighted similarity based on edge strength for image quality assessment,” International Journal of Computer Theory and Engineering, vol. 1, pp. 138-141, June. 2009.

[19] Z. Wang and Q. Li, “Information content weighting for perceptual image quality assessment,” IEEE Transactions on Image Processing, vol. 20, pp. 1185-1198, 2011.

[20] Punit Singh and Damon M. Chandler, “F-mad: a feature-based extension of the most apparent distortion algorithm for image quality assessment,” Proc. SPIE, Image Quality and System Performance X, SPIE Digital Library , vol. 8653, Feb. 2013.

[21] J.Findlay. “The visual stimulus for saccadic eye movement in human observers,” Perception, vol.9, pp. 7-21, Sept. 1980.

[22] J. Senders. “Distribution of attention in static and dynamic scenes,” Proceedings of SPIE, vol. 3016, Feb. 1997, pp. 186-194.

[23] Claudio M. Privitera and Lawrence W. Stark, “Algorithms for defining visual regions of interest: Comparison with eye fixations,” IEEE Tans. on Pattern Analysis and Machine Intelligence, vol. 22, Sept. 2000.

[24] S. Venkatesh and R. Owens, “An energy feature detection scheme,” Proc. of The International Conference on Image Processing, 1989, pp. 553-557.

[25] D. J. Heeger, “Normalization of cell responses in cat striate cortex,” Visual Neuroscience, vol.9, pp.181-197, 1992.

[26] E. H. Adelson and J. R. Bergen, “Spatiotemporal energy models for the perception of motion,” Journal of the Optical Society of America A, vol. 2, pp. 284-299, 1985.

[27] J. Morlet, G. Arens, E. Fourgeau, and D. Giard, “Wave propagation and sampling theory - part ii: Sampling theory and complex waves,” Geophysics, vol. 47, pp. 222-236, Feb. 1982.

[28] D. J. Field, “Relations between the statistics of natural images and the response properties of cortical cells,” Journal of Optical Society of America, vol. 12, pp. 2379-2394, 1987.

[29] Peter Kovesi, “Image features from phase congruency,” Videre: A Journal of Computer Vision Research, vol.1, 1999.

[30] H. R. Sheikh, A. C. Bovik, L. Cormack, and Z. Wang, LIVE image quality database, [Online]. Available: http://live.ece.utexas.edu/research/quality.

[31] Ann Marie Rohaly, Philip Corriveau, John Libert, Arthur Webster, Vittorio Baroncini, and John Beerends, VQEG, “Final report from the video quality experts group on the validation of objective models of video quality assessment,” [Online]. Available: http://www.vqeg.org/.

Color Constancy For Improving Skin Detection

A. Nadian-Ghomsheh *a_nadian@sbu.ac.ir*
Cyberspace research group,
Shahid Beheshti University, GC
Tehran, 1983963113, Iran

Abstract

Skin detection is a preliminary step in many human related recognition systems. Most skin detection systems suffer from high false detection rate, resulting from low variance between the skin and non-skin color distributions. This paper proposes the use of simple color correction algorithms with low computation complexity to obtain a corrected version of the skin color distribution, which could lead to more accurate skin detection. White patch retinex, Grey world assumption and several improved versions of these two state of the art correction algorithms were chosen and applied to an image set of 4000. The results, compared with skin detection with no color correction revealed that color correction will improve the skin detection accuracy.

Keywords: Skin Detection, Color Constancy, Gaussian Distribution, White Patch Retinex, Grey World Assumption.

1. INTRODUCTION

Human skin is widely used in image processing research related to humans, with applications in areas such as face detection [1], person tracking [2], pornography filtering [3], and steganography [4]. Recent approaches for skin detection are focused on skin color information, since it is an invariant feature against rotation and scaling [5].

Visible spectrum imaging has been the focus for skin detection in most research. This is a challenging task due to various factors such as illumination, camera characteristics, ethnicity, and added facial components. Basically, skin detection can be considered as classification of skin pixels and non-skin pixels. Three steps have to be considered for skin detection: choice of a proper color space, modeling skin and non-skin pixels with a suitable model, and classification of the modeled distributions.

Many color spaces have been considered for skin detection. Basic color spaces such as the RGB and normalized RGB (nRGB) have shown to be very attractive for skin detection since RGB is the default color space for many available image formats [6-8]. Perceptual color spaces, such as HSV, HSI, HSL, and TSL are another group of color spaces that have been used for skin detection extensively [9-11]. Transforming for RGB to such color spaces is invariant to high intensity, surface orientation and ambient light which makes these color space attractive for skin detection. However, perceptual feature of these group of color spaces, such as hue and saturation cannot be directly converted from a RGB space, and many transformation that do so are based on non-linear equations which could increase the computation time for skin detection. Orthogonal color space, such as YCbCr, YIQ, YUV and YES on the other hand are transformed into RGB space using linear transformations [12-14]. Using this group of color spaces, it is possible to separate the luminance component of a color space from its chrominance component. This is a great advantage when considering a color space for skin detection, because omitting the luminance in an image will help to achieve a skin color distribution independent of scene illumination [15].

Detecting skin pixels can be viewed as a two class problem, where, skin pixels are grouped against the non-skin pixels. In general, two approached are considered for skin detection: pixel-based skin detection and regional-based skin detection. Pixel-based skin detection uses the pixel color information for detecting skin pixels, where as regional methods use the information of an ensemble of pixels to detect the skin region [16].

Pixel-based methods use the color information from a large number of training skin pixels to obtain the skin color distribution used to detect skin pixels [15, 17]. In some methods, explicit rules were used to detect the skin region for naked image detection [18, 19]. Although this approach executes very fast, it suffers from lack of accuracy [20]. Histogram-based Gaussian mixture model was used by [21]. In this method, the processing time increases dramatically as the number of Gaussians is increased. Lee [22] used a learning-based chromatic distribution-matching scheme to determine the image's skin chroma distribution online.

Regional-based methods use the relationship between pixels to detect the human skin. In the work of [18], after using explicit rules for skin detection, a region growing step was performed to refine the detected skin region. In the method of [23], skin pixels were detected by finding the skin color distribution from certain areas around the eyes. Adaptive methods have also been used for detecting skin pixels in arbitrary images [24-26]. These methods are dependent on the information within each image, rather than the initial skin color distribution. However, if the adaptation step requires an intense iterative process, computation time increases dramatically.

In general, pixel-based skin detection suffers from low accuracy [15, 27, 28]. On the other hand, regional methods have shown to be more accurate compared to pixel-based methods [26, 29]. However, simplicity of implementation and low computational cost, makes pixel-based skin detection more widely used in applications that require skin detection.

In this paper we try to improve the skin detection accuracy using color constancy. Color constancy is integrated with the Gaussian model for skin detection. The focus of this study is to investigate the effectiveness of simple to implement color constancy algorithms for improving the accuracy of skin detection. White Patch Retinex (WPR) algorithm, the Grey World (GA) assumption and some of their variations are considered as color correction algorithms. The skin image database used in this paper was a collection of 4000 images divided into two equal sets of skin and non-skin images. The images were randomly chosen from the QOMPAC database[30].

The rest of the paper is organized as follows: in section two, color correction algorithms used in this paper are briefly explained. Section three explains the training and test phases of the skin detection procedure used in this paper. Experimental results are shown in section four and the paper is concluded in section five.

2. COLOR CORRECTION FOR SKIN DETECTION

Color is an important biological signalling mechanism, and without it, objects could no longer be reliably identified by their color [31]. There are two approaches to follow in developing color constancy algorithms. The first method is to determine the reflectance of objects. The second approach is to perform a color correction that closely mimics the performance of the visual system.

In our work, we consider the fact that different lighting situations and the camera settings will produce a skin color distribution map that is different when compared to a color corrected version. By correcting the colors in images, it is possible to obtain a skin color distribution that is more similar to the true skin color distribution. Thus, the objective of this study is not focused on finding a color correction algorithm that is closer to what human visual system expects, but to find a correction algorithm that is more suited for distinguishing between skin pixels and pixels that show the same color properties as skin. This similarity could be the result of scene illumination or the camera calibration. Figure 1a shows a scene at sunset. It is clear that the colors in this image

are very close to the color of human skin, mostly due to the illumination of the scene. Figure1b shows the same image corrected with the Multi Scale Retinex with Color Restoration (MSRCR) [32] method. The output of the color correction algorithm might not be considered as a good picture, but the output image of this algorithm clearly changes the skin-like colors, so that most of the pixels in the output image do not have colors which are close to the color of the human skin. On the other hand, in Figures 1c and 1d, it's shown that the pixels that belong to the face region, still resemble the color of the skin after applying MSRCR. This examples shows how color constancy can be used in favour of improving the accuracy of skin detection.

There are many color constancy algorithms available ranging from very simple algorithms such as WP algorithm and some more complex and sophisticated algorithms that are capable of handling the correction of poorly colored images such as method presented in [33, 34]. In this study, the desired algorithm is the one that is executed fast and has the capability of correcting images with various lighting situations up to an expectable level. For this reason, we have chosen two main approaches: the WP algorithm and some improved versions of this method. The Gray World assumption (GW) algorithm has also been considered in this paper. These methods are briefly explained in this section.

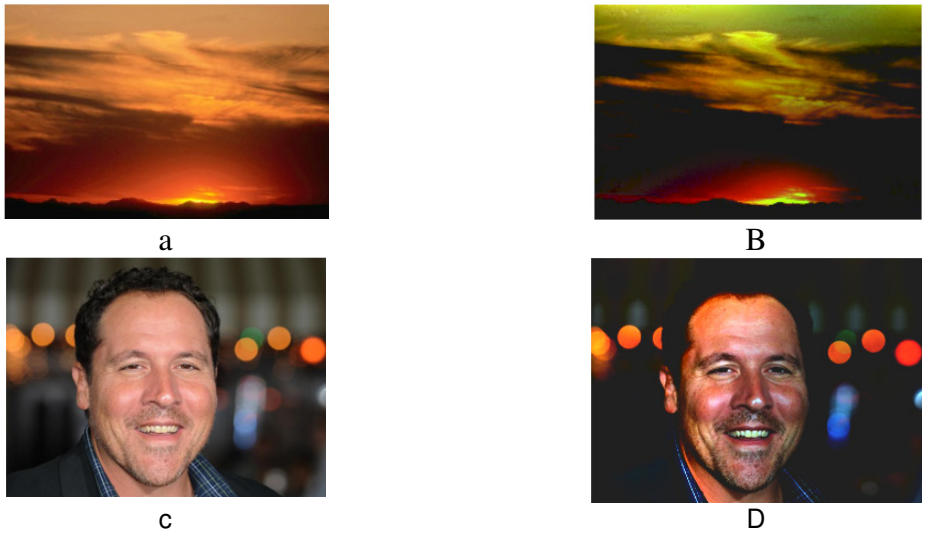


FIGURE 1: a) sun set scene and **b)** its corrected version. **c)** A skin image and its **d)** corrected version.

2.1 White Patch Retinex Method

The WP algorithm relies on having a bright patch somewhere in the image. The idea is that, if there is a white patch in the scene, then this patch reflects the maximum light possible for each band, which can be considered as the color of the illuminant.

In this model, the illuminant is considered to be uniform and the intensity at pixel position *(x,y)* for each color channel $i \in \{R,G,B\}$ representing the three color channels red, green and blue, is calculated by:

$$I_i(x, y) = G(x, y)R_i(x, y)L_i \quad (1)$$

where $G(x,y)$ is a factor that depends on the scene geometry at the corresponding object position, $R_i(x,y)$ is the reflectance for wavelength λ_i, and L_i is the irradiance at wavelength λ_i. For WP it is assumed that $R_i(x,y)=1$ for all i, and $G(x,y)=1$. Since the maximum intensity values in each channel is considered, the color of illuminant is obtained by:

$$L_{i,\max} = \max_{x,y} \{c_i(x, y)\} \quad (2)$$

where c is the color value of a pixel at position (x,y) in the image. The color corrected image can obtained by normalizing the image by $L_{i,max}$:

$$O_i(x, y) = \frac{c_i(x, y)}{L_{i,\max}} \quad (3)$$

where O is the output of the color correction algorithm for the i^{th} channel. In the Modified White Patch Retinex (MWP) algorithm, the pixels that were used to find the maximum $L_{i,max}$ were chosen at a gray level less than the maximum saturated value [31]. Thus $L_{i,max}$ in (3) would be less than the actual maximum value in the image.

An improved version of WP was presented by [35], called the Single-Scale Retinex (SSR). In this algorithm, $L_i(x,y)$ is estimated by applying a Gaussian-form linear low pass filter to an input color image $c_i(x,y)$. $o_i(x,y)$, is then obtained by subtracting the log signal of the estimated illumination from the log signal of the input color image as follows:

$$o_i = \log c_i(x, y) - \log l_i(x, y) \quad (4)$$

This algorithm is shown to have trade-off between dynamic range compression and tonal rendition. Thus, improving each factor will cause a degradation on the other. Jobson, [36] further improved their work to overcome this problem be implementing the Multi-Scale Retinex (MSR). In this method the surround function is represented as:

$$o_i = \sum_{n=1}^{N} w_n O_{SSRni}(x, y) \quad (5)$$

where, N represents the number of scales, W_n represents the weight of each scale, and O_{SSRni} are the outputs of the SSR algorithm. In the SSR and MSR algorithm, a color change in the output image may occur, because these algorithms are employed to color channels i separately, and each channel is enhanced independently. In this algorithms, the number of bins that image should be quantized into, needs to be specified.

MSRCR was introduced by [32] to overcome these problems. In this method, the output image is calculated as:

$$o_i = WF_i O_{MSRi}(x, y) \quad (6)$$

where O_{MSRi} is the output image from the MSR for channel i, and WF_i is the weight function obtained via:

$$WF_i = b \log\left[a \frac{O_{MSRi}(x, y)}{\sum_{j \in \{R,G,B\}} O_{MSRj}(x, y)} \right] \quad (7)$$

In the equation, a and b are the adjustment constants in this algorithm.

2.2 Gray World Assumption

The gray world color constancy algorithm assumes that the colors of the objects in view are

uniformly distributed over the entire color range and we have a sufficient number of objects with different colors in the scene, then the average color computed for each channel will be close to 0.5. The corrected image via the gray world assumption can be obtained by:

$$o_i(x,y) = \frac{c_i(x,y)}{fa_i} \tag{8}$$

where f is a constant related to the expected value of the geometry factor G and the and a is the space color average of an image with size $n=nx\times ny$ obtained by:

$$a_i = \frac{1}{n} \sum_{x,y} c_i(x,y) \tag{9}$$

Using this notation, a_i can be affected by the size of different objects in the scene. One way to overcome this problem is to calculate a_i for a color segmented image [31]. Let n_r be the number of different regions and let $a(R_j)=[a_R(R_j),a_G(R_j),a_B(R_j)]^T$ be the average color of region $J\epsilon\{1,\ldots,n_r\}$. Now, a_i can be calculated by looping over the unique regions:

$$a_i = \frac{1}{n_r} \sum_{j=1}^{n_r} a_i(R_j) \tag{10}$$

This approach was further improved by using image histograms instead of using the segmented image. This was performed to avoid the redundant segments generated resulted from the position of objects in the scene. The resulting a_i was calculated by:

$$a_i = \frac{1}{n_{nz}} \sum_{j=1}^{n_b} c_j(j) \tag{11}$$

were n_b is the total of buckets used in the histogram, and n_{nz} is the number of nonzero buckets.

3. SKIN DETECTION

To detect the skin pixels, it is required to choose a color space that skin pixels are well distributed within that space. Much research has been devoted to finding such a color space [5]. Although, no unique color space has been defined as the best choice for skin detection, two points can be noted: chrominance spaces provide sufficient information for skin detection, and YCbCr, normalized-RGB (nRGB), have been frequently reported as good choices for detecting skin pixels. Thus, these two spaces have been considered in this study. The block diagram of the skin detection process for the training and test phase implemented in this paper are shown in figure 2.

YCbCr color space is obtained from the RGB space by:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.55 & 24.96 \\ -37.3 & -74.2 & 112 \\ 112 & -93.7 & -18.2 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{12}$$

In this space, Cb and Cr color channels represent the chrominance information and are used for skin detection. nRGB space is obtained from RGB space via:

$$r = \frac{\mathrm{R}}{\mathrm{R}+\mathrm{G}+\mathrm{B}}, \quad g = \frac{G}{R+G+B}, \qquad b = \frac{B}{R+G+B} \tag{13}$$

Since r + g + b = 1 in (13), the information of one color channel will be redundant and can be ignored. In this paper, the color channels *r* and *g* channels have been considered for skin detection.

The Gaussian distribution was chosen for skin pixel classification. To implement the method, the skin color distribution obtained from the sample skin pixels were used to estimate the Gaussian distribution representing the skin class. In the test stage, the probability of each pixels was obtained via:

$$P_{Gaussian}(X \mid \mu, \Lambda) = \frac{1}{2\pi|\Lambda|^{0.5}} \exp^{-0.5(X-\mu)^T|\Lambda|^{-1}(x-\mu)} \tag{14}$$

were X is the input pixel, μ and Λ are the mean vector and covariance matrix of the skin distribution in the respective color space. Each pixel with $P_{Gaussian} > T$ is classified as skin.

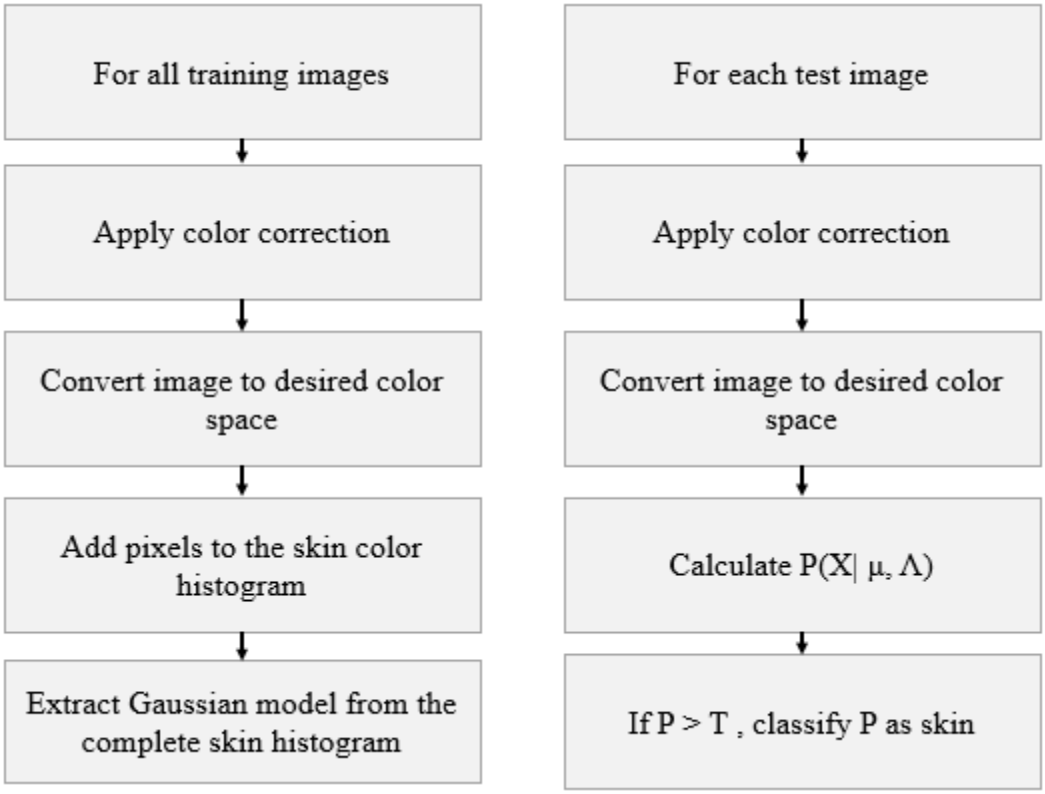


FIGURE 2: Left. Training process of skin detection. **Right.** Steps for skin detection.

4. EXPERIMENTAL RESULTS

This section reports the result of skin detection on color corrected images. The test data set used in this study was a collection of 2000 skin image and 2000 non-skin images, randomly chosen from the COMPAQ data base. In addition, 1000 test images were used to obtained skin color distribution in the training phase. Choosing two color spaces and several color constancy algorithms provide a large set of experimental scenarios. Some of the color constancy algorithm used in this paper require parameter settings which further increased the number of cases that

require to be experimented with.
For WP and the GW, no specific setting was required. Testing these algorithms on the YCbCr and nRGB spaces made four test cases. For the SSR and MSR algorithms, the number of bins that was used to build the image histogram needed to be considered. In this study histograms were build using {16, 32, 64, 128, 256} bins. Further, SSR and MSR were applied to the R channel, RGB channels, Y channel, and YCbCr channels from the RGB and YCbCr space separately, providing 40 test cases. The MWP algorithm was tested for 4 cases by choosing input entries {50, 100, 150, 200} as the threshold for finding the white patch in the image. For this algorithm, skin detection was applied using the YCbCr space. The MSRCR algorithm required three tuning parameters. By choosing three values for each entry as {a ϵ [2 5 7], b ϵ [2 5 7], scales ϵ [10 15 20]}, a total of 27 scenarios were required to be tested for this algorithm.

Figure 3 shows the result of color correction on two sample images. As this figure shows, applying color constancy algorithms influences both the skin color and the color of non-skin pixels. Although these changes might not favour the better appearance of an image, however, such changes might lead to a better skin detection result. For example, in the case of image correction with WP algorithm, the skin image has preserved its skin color, while the color of the non-skin image has totally changed. As another example, when MWP is applied to the sample images of figure 3, the colors in the non-skin image does not resemble the skin color after color correction, while, the skin image shows a color skin that looks more relevant with the human visual system after color correction. The result of skin detection for each case is shown in the following paragraphs of this section.

FIGURE 3: Applying color correction algorithms to two sample skin (left) and non-skin (right) images.

Figure 4 shows the skin color distribution of skin pixels in Cb-Cr space. As the figure indicates the skin color distribution is much different for each correction algorithm. Skin color distribution obtained from the WP, GW and MSR algorithms represent skin color distribution in a small region of the space when compared with the other three method.

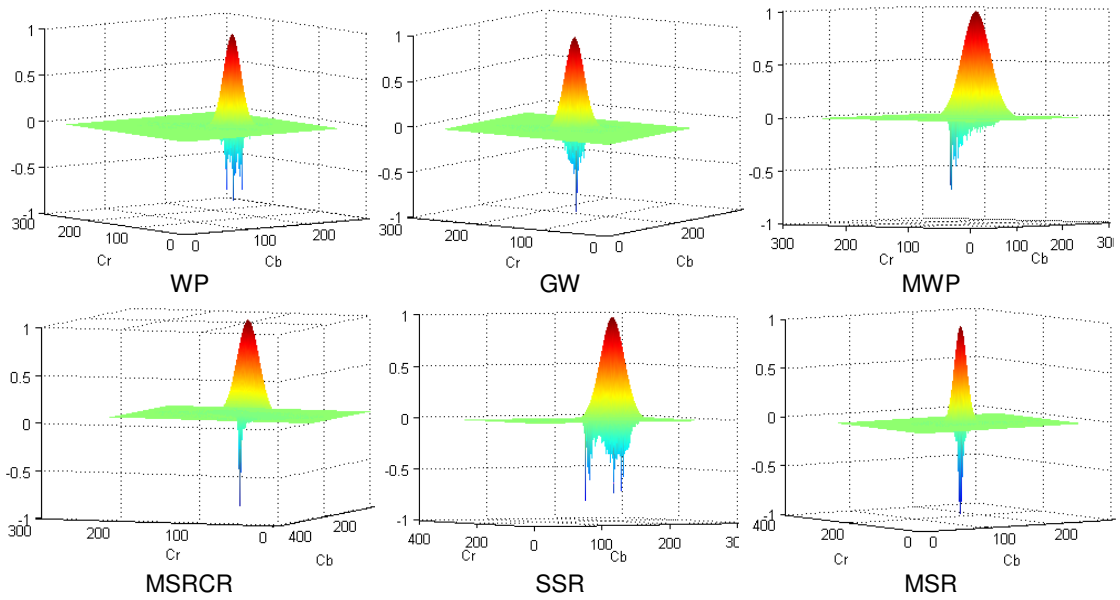


FIGURE 4: Skin color distribution after color correction in Cb-Cr chrominance space.

Although it might seem that MSR, WP, and GW algorithm should be better correction methods, but, their effect on non-skin images is unknown. To numerically evaluate the effectiveness of the color correction algorithm for improving skin detection accuracy, Recursive Operation Characteristic (ROC) curves were used based on the values of True Positive Rate (TPR) and

False Positive Rate (FPR) [30]. The accuracy of skin detection was obtained via:

$$Acc = \frac{TPR}{TPR + FPR} \tag{15}$$

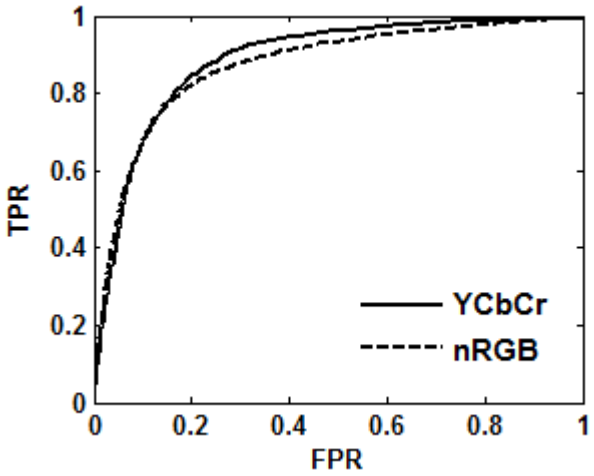


FIGURE 5: Result of skin detection before applying color constancy.

Figure 5 shows the result of skin detection before applying any color correction to the image data set. As this figure shows, the YCbCr color space yield better results when compared to that of the nRGB color space. Table 1 summarizes the FPR rates for TPR = 90% and 95%. As the table indicates, the YCbCr outperforms the nRGB space by 4.5% for TPR of 90%.

	TPR(%)	FPR(%)	Accuracy(%)
YCbCr	85	19	83
nRGB		24	80
YCbCr	90	28	81
nRGB		37	76.5

TABLE 1: Skin detection results without color correction.

Figure 6 shows the result of skin detection after applying the WP and GW color correction algorithms. As the results indicate, after applying color correction algorithms, the YCbCr algorithm outperforms the result of skin detection when using nRGB color space. Comparision between the two correction algorirhms reveal that the WP had better skin detection result. Table 2 shows the numerical results obtained from figure 6. An accuracy of 84.25% was obtained when skin detection was performed using Cb-Cr chrominance space and WP algorithm was used for color correction. The best result for skin detection in nRGB color space was 81.75%, which was obtained when WP algorithm was used for color correction.

		TPR(%)	FPR(%)	Accuracy(%)
WP	YCbCr	85	17	84
	nRGB		21.5	81.75
	YCbCr	90	21.5	84.25
	nRGB		34.5	77.75
GW	YCbCr	85	19	83.75
	nRGB		22	81.5
	YCbCr	90	26.5	81.75
	nRGB		35	77.5

TABLE 2: Skin detection results for applying WP and GW algorithms.

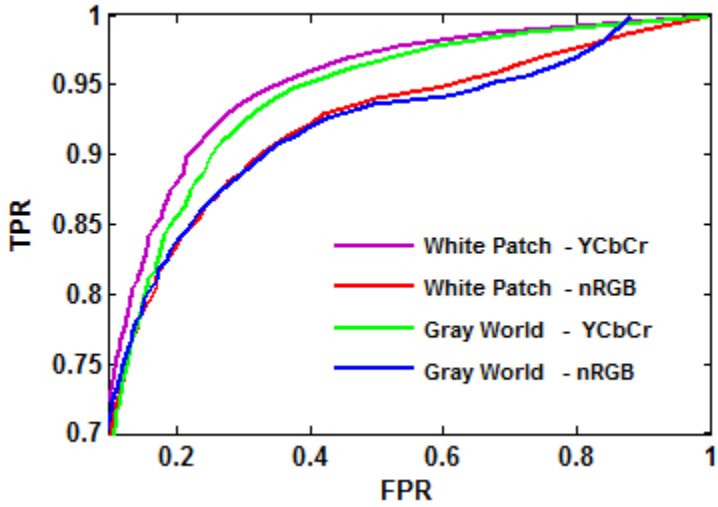
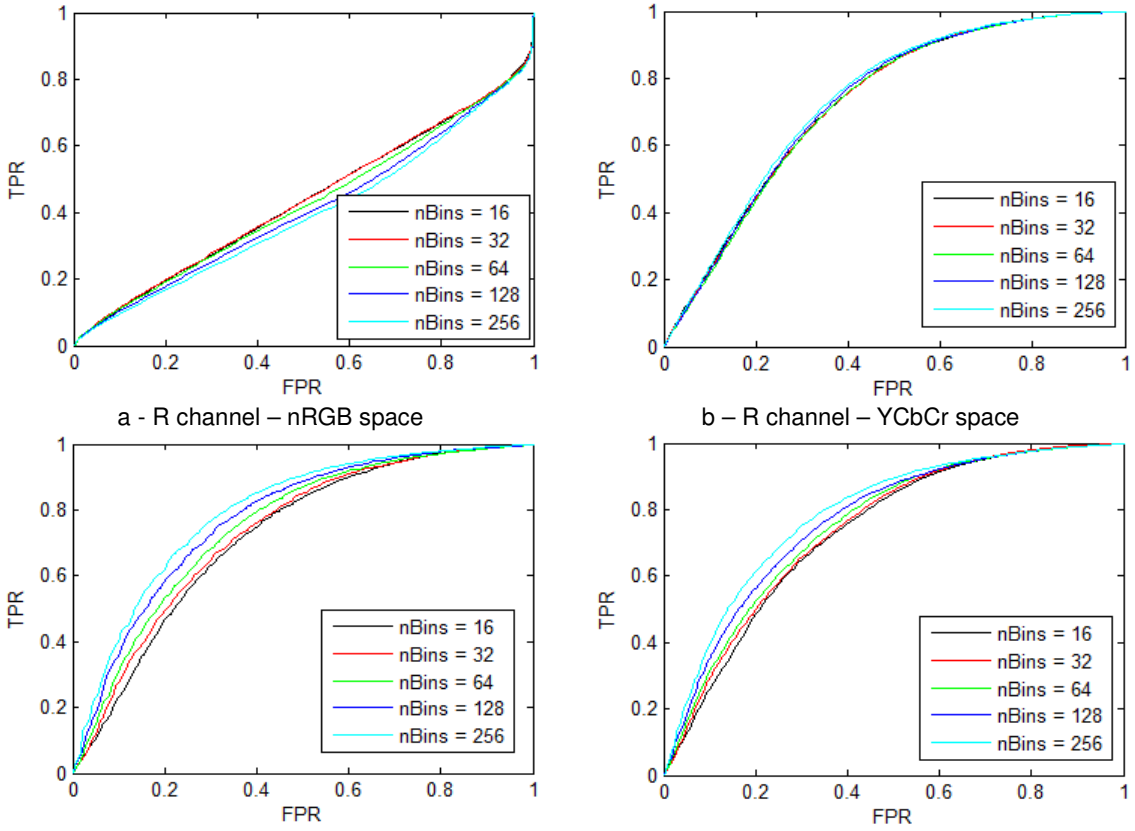


FIGURE 6: Result of skin detection after applying WP and the GW.

The skin detection ROC curves obtained after applying the SSR algorithm are shown in figure 7. Each figure is captioned as X_c-Y_s, where X_c shows which color channels were corrected and Y_s show which color space was used for skin detection. X ϵ {R, RGB, Y, YCbCr} and Y ϵ {nRGB, YCbCr}. For figure 7a through 7d, the color correction was applied to the color channels of the RGB space, where, the best ROC curves for applying SSR algorithm was obtained. The figure also indicates that correction of color in the YCbCr space has a negative effect on the skin detection accuracy. For the case where SSR was applied to all color channels of the YCbCr space, the skin detection accuracies were dramatically decreased to unacceptable levels.



a - R channel – nRGB space

b – R channel – YCbCr space

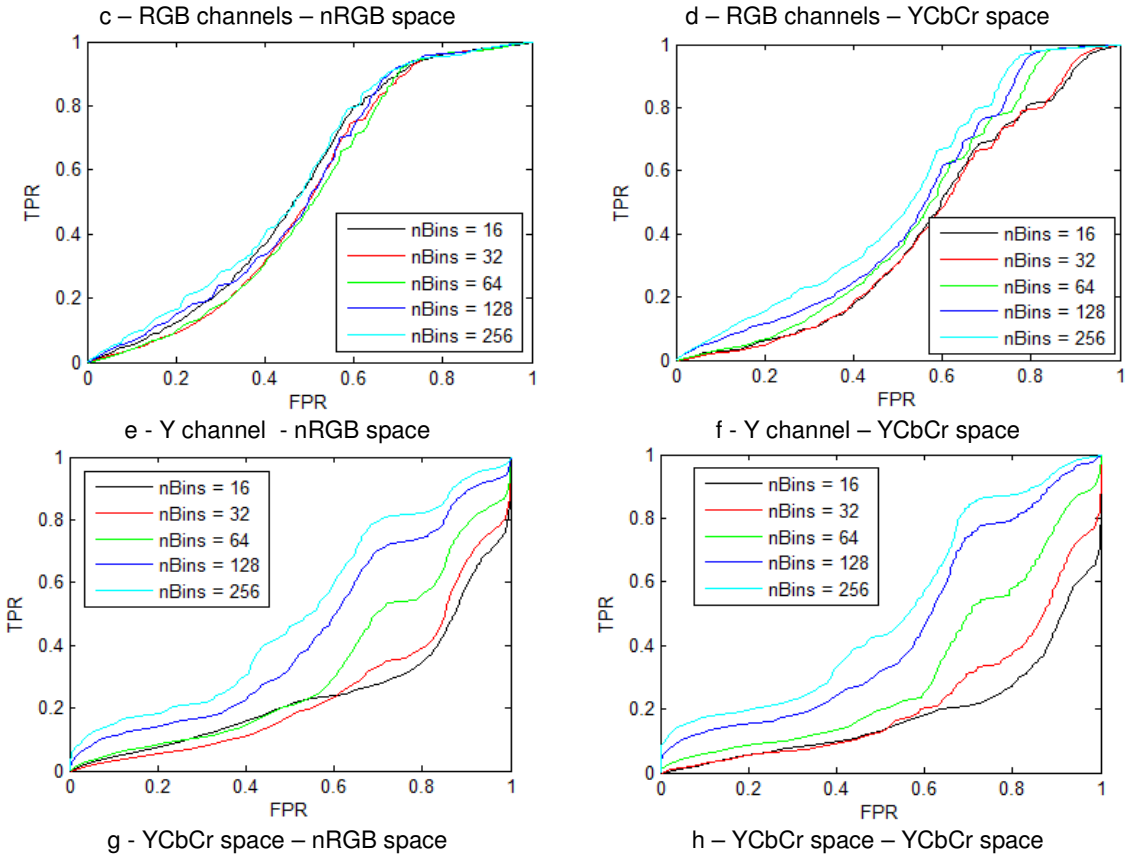


FIGURE 7: The ROC curves obtained from different scenarios when SSR algorithm was used for color correction.

Table 3 shows the results of skin detection obtained from the ROC curves of figure 7. As the results in this table indicate, none of the scenarios have improved the skin detection accuracy. However, among the tested scenarios, when color constancy was applied to all channels of the RGB color space, an accuracy of 73.5% was obtained. In this case, nRGB color space was used for skin detection.

	Color space	Number of bins	Corrected channel	TPR(%)	FPR(%)	Accuracy(%)
Single – Scale Retinex	nRGB	128	R	85	85	50
			RGB	85	38	73.5
		64	R	85	80	52.5
			RGB	85	42	71.5
	YCbCr	128	R	85	47	69
			RGB	85	41	72
		64	R	85	46.5	69.25
			RGB	85	44	70.5

TABLE 3: Skin detection accuracy using SSR for color correction.

Figure 8 shows the result of skin detection when MSR algorithm was used for color correction. The figures are labelled in the same manner as the result of skin detection with SSR algorithm. As the figure indicates, for correcting YCbCr color channels unacceptable results was obtained, as it was the case with the SSR algorithm. Table 4 shows some of the results obtained when MSR algorithm was applied for color correction. As the table indicates, the best result was

obtained when color constancy algorithm was applied to all the channels of the RGB color space, where an accuracy of 73.5% was obtained,

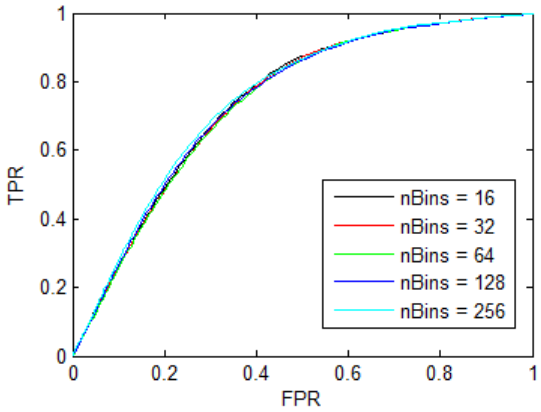
	Color space	Number of bins	Corrected channel	TPR(%)	FPR(%)	Accuracy (%)
Multi – Scale Retinex	nRGB	64	R	85	43	71
		64	RGB	85	41	72
		128	R	85	43	71
		128	RGB	85	38	73.5
	YCbCr	64	R	85	45	70
		64	RGB	85	44	70.5
		128	R	85	44	70.5
		128	RGB	85	42	71.5

TABLE 4: Skin detection accuracy using MSR for color correction.

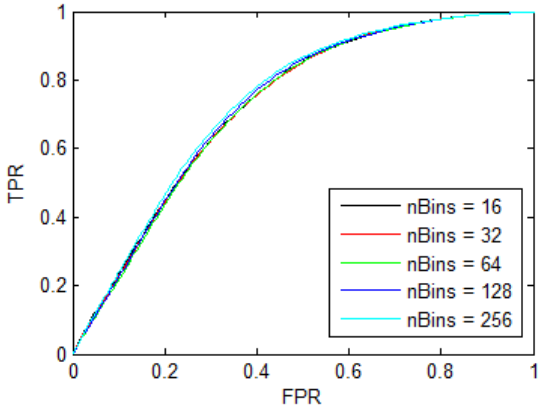
When MWP algorithm was used for color correction, the ROC curves shown in figure 9 where obtained. As this figure indicates, for threshold 150 and 200, acceptable ROC curves were obtained. Table 3 summarizes the result of skin detection after applying the MWP for color correction. As shown in the table, choosing a threshold of 200 resulted in skin detection with accuracy of 86.25% for TPR = 85%. In the case for TPR = 90%, an accuracy of 86% percent was obtained. This result indicates that MWP had better performance compared with the WP algorithm, because for the WP, the saturated value is used for luminance estimation and in most cases a single fully saturated point might decrease the performance of the algorithm.

	Threshold	TPR(%)	FPR(%)	Accuracy(%)
MWP	200	85	12.5	86.25
		90	18	86
		95	27	84
	150	85	23.5	80.75
		90	31	79.5
		95	45	75

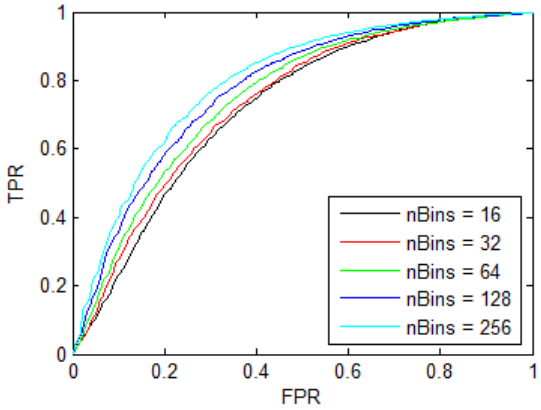
TABLE 5: Skin detection accuracy results after applying the MWP.



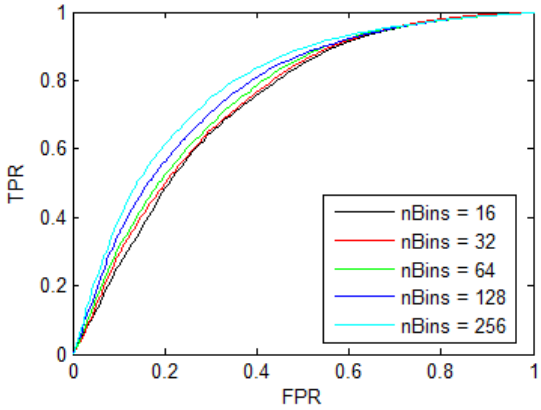
a – R channel - nRGB space



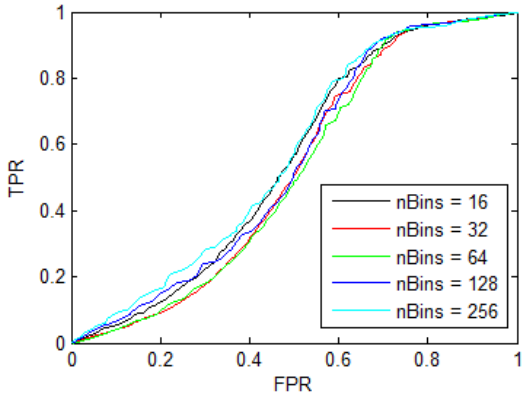
b – R channel - YCbCr space



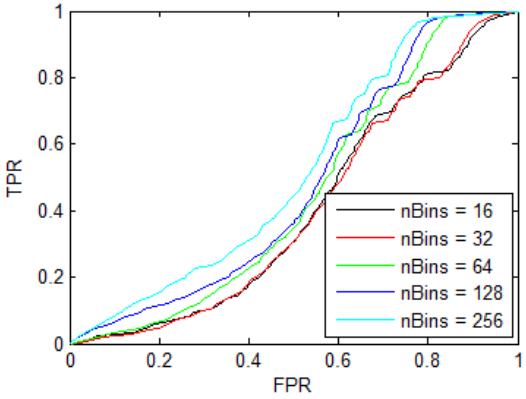
c – RGB channels - nRGB space



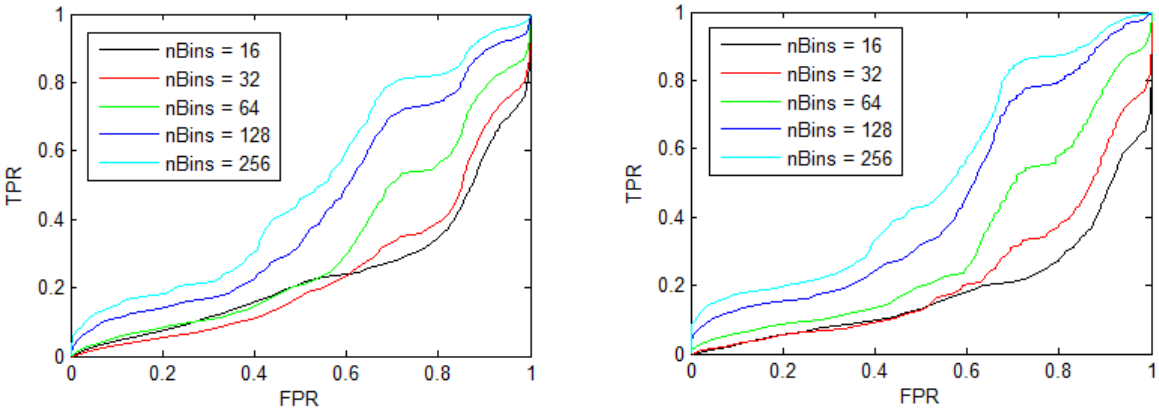
d – RGB channels - YCbCr space



e – Y channel - nRGB space



f – Y channel - YCbCr space



g – YCbCr Channels - nRGB space

h – YCbCr channels - YCbCr space

FIGURE 8: The skin detection ROC curves obtained from different settings of MSR.

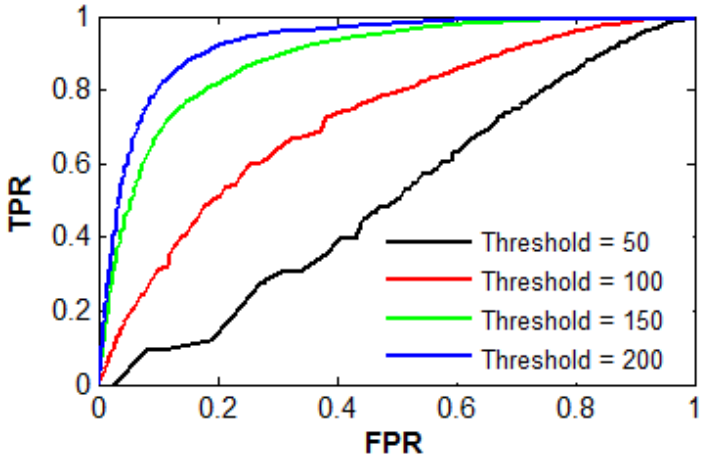


FIGURE 9: The result of skin detection after applying the MWP algorithm.

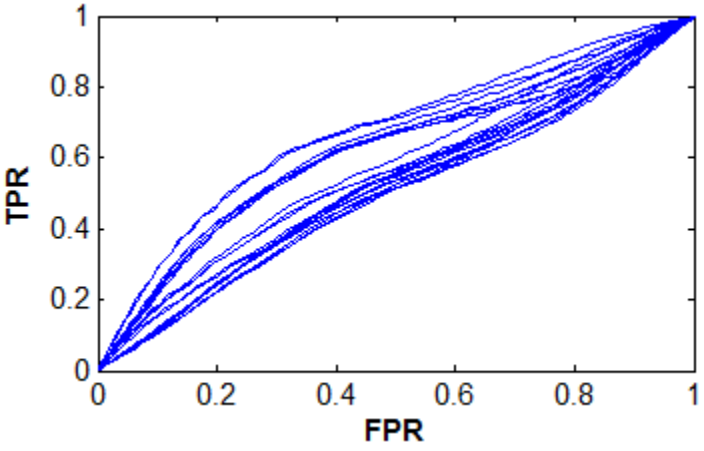
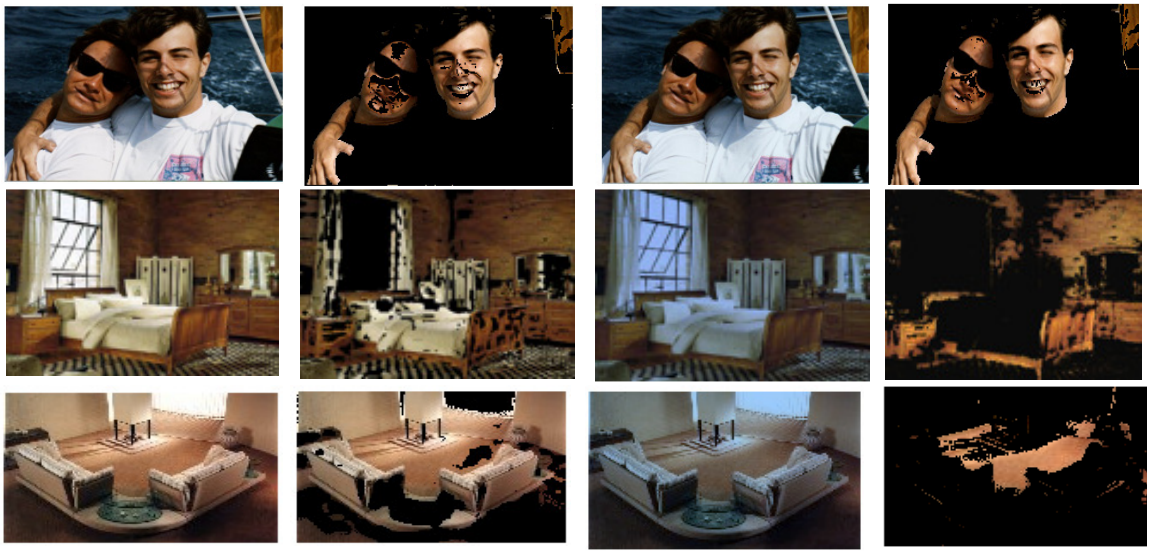


FIGURE 10: Skin detection ROC curves after applying the MSRCR color correction algorithm.

For the case of image correction using the MSRCR algorithm, unacceptable results were obtained. In the best case the accuracy did not exceed 70%. Figure 10 shows the skin detection ROC curves obtained for different test scenarios.



a b c d

FIGURE 11: Examples of detected skin region after applying MWP color correction algorithm. a) Original image, b) skin region detected without color correction, c) color correction image, and d) detected skin region after color correction.

Figure 11 shows the result of skin detection for three samples images. The detected skin region before color correction, the color corrected image and the detected skin region after color correction are shown for each image. The images were color corrected using the MWP algorithm. For the skin image, the detected skin region is slightly improved. For the non-skin images, the number of pixels that have been incorrectly classified as skin are significantly reduced. This results confirms with the main idea of this paper, where, it was suggested that correcting the color of images will produce a more reliable skin color distribution map, and many non-skin images after color correction will produce colors that are different with the human skin color.

Figure 12 compares the result of skin detection without color correction against scenarios were color correction was applied to images prior to skin detection. Only, the result of cases with accuracy greater than that obtained from the skin detection with no color correction are reported in this figure. Overall, WP, GW, and the MWP algorithms showed to be effective for improving skin detection, and color correction using MWP improved the skin detection accuracy by more than 3%.

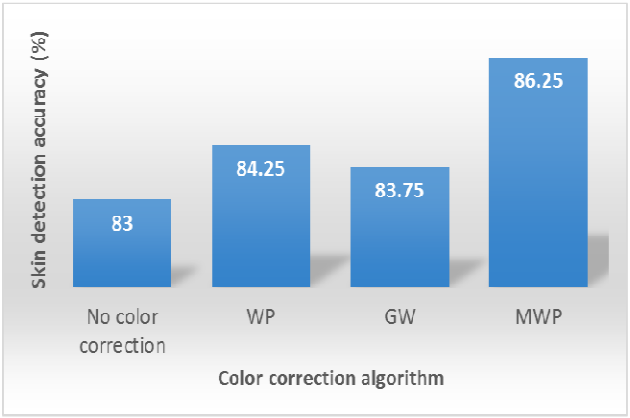


FIGURE 12: Comparison between different skin detection accuracies.

5. CONCLUSION

In this paper, we proposed the use of simple color correction algorithms for improving skin detection accuracy. Several versions of the WP and the GW algorithms were used for color correction and their results were tested against skin detection with no color correction. The results showed that applying the WP and GW prior the skin detection will improved the result of skin detection. Further, the MWP algorithm was the most efficient method for improving skin detection accuracy among the color correction algorithms tested in this paper. Using MWP, a skin detection accuracy of 86.25% was obtained which was an improvement to skin detection without color correction by 3.25%. The various results reported in this paper confirmed that applying simple color correction algorithms prior to skin detection can improve the skin detection accuracy. Future research on color constancy for improving skin detection can be focused on finding color correction algorithms that are especially tailored for the application of skin detection. Overall, MWP was simple to implement and did not require complex computation for obtaining color correction, this algorithm is suitable for improving skin detection accuracy.

6. REFERENCES

[1] OJO, J.A. and S.A. Adeniran, Colour face image database for skin segmentation, face detection, recognition and tracking of Black faces under real-life situations. International Journal of Image Processing (IJIP), 2011. 4(6): p. 600.

[2] Choi, W., C. Pantofaru, and S. Savarese, A general framework for tracking multiple people from a moving camera. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2013. 35(7): p. 1577-1591.

[3] Nadian, A. and A. Talebpour, A New Skin Detection Approach for Adult Image Identification. Research Journal of Applied Sciences, Engineering and Technology, 2012. 4(21): p. 4535-4545.

[4] Shejul, A.A. and U.L. Kulkarni, A secure skin tone based steganography using wavelet transform. International Journal of computer theory and Engineering, 2011. 3(1): p. 16-22.

[5] Kakumanu, P., S. Makrogiannis, and N. Bourbakis, A survey of skin-color modeling and detection methods. Pattern recognition, 2007. 40(3): p. 1106-1122.

[6] Yang, M.-H. and N. Ahuja. Gaussian mixture model for human skin color and its application in image and video databases. in Proc. SPIE: Storage and Retrieval for Image and Video Databases VII. 1999.

[7] Bergasa, L.M., et al., Unsupervised and adaptive Gaussian skin-color model. Image and Vision Computing, 2000. 18(12): p. 987-1003.

[8] Soriano, M., et al., Adaptive skin color modeling using the skin locus for selecting training pixels. Pattern Recognition, 2003. 36(3): p. 681-690.

[9] Wang, Y. and B. Yuan, A novel approach for human face detection from color images under complex background. Pattern Recognition, 2001. 34(10): p. 1983-1992.

[10] Mohammed, K., B.C. Ennehar, and T. Yamina, Skin detection using gaussian mixture models in YCbCr and HSV color space. Global Journal on Technology, 2012. 1.

[11] Berbar, M.A., Skin colour correction and faces detection techniques based on HSL and R colour components. International Journal of Signal and Imaging Systems Engineering, 2014. 7(2): p. 104-115.

[12] Cao, X.Y. and H.F. Liu, A skin detection algorithm based on Bayes decision in the YCbCr color space. Applied Mechanics and Materials, 2012. 121: p. 672-676.

[13] Tao, L., et al. A circuit of configurable skin tone adjusting method base on exact skin color region detection. in Electron Devices and Solid-State Circuits (EDSSC), 2011 International Conference of. 2011. IEEE.

[14] Al Tairi, Z.H. and M.I. Saripan, Skin segmentation using YUV and RGB color spaces. Journal of information processing systems, 2014. 10(2): p. 283-299.

[15] Nadian, A. and A. Talebpour. Pixel-based skin detection using sinc function. in Computers & Informatics (ISCI), 2011 IEEE Symposium on. 2011. IEEE.

[16] Nadian, A., A. Talebpour, and M. Basseri. Regional skin detection based on eliminating skin-like lambertian surfaces. in Computers & Informatics (ISCI), 2011 IEEE Symposium on. 2011. IEEE.

[17] Jedynak, B., H. Zheng, and M. Daoudi. Statistical models for skin detection. in Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on. 2003. IEEE.

[18] Hammami, M., Y. Chahir, and L. Chen, Webguard: A web filtering engine combining textual, structural, and visual content-based analysis. Knowledge and Data Engineering, IEEE Transactions on, 2006. 18(2): p. 272-284.

[19] Shih, J.-L., C.-H. Lee, and C.-S. Yang, An adult image identification system employing image retrieval technique. Pattern Recognition Letters, 2007. 28(16): p. 2367-2374.

[20] Vezhnevets, V., V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. in Proc. Graphicon. 2003. Moscow, Russia.

[21] Hu, W., et al., Recognition of pornographic web pages by classifying texts and images. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2007. 29(6): p. 1019-1034.

[22] Lee, J.-S., et al., Naked image detection based on adaptive and extensible skin color model. Pattern recognition, 2007. 40(8): p. 2261-2270.

[23] Jang, S.-W., et al., An adult image identification system based on robust skin segmentation. Journal of Imaging Science and Technology, 2011. 55(2): p. 20508-1.

[24] Cho, K.-M., J.-H. Jang, and K.-S. Hong, Adaptive skin-color filter. Pattern Recognition, 2001. 34(5): p. 1067-1073.

[25] Kawulok, M., J. Kawulok, and J. Nalepa, Spatial-based skin detection using discriminative skin-presence features. Pattern Recognition Letters, 2014. 41: p. 3-13.

[26] Sun, H.-M., Skin detection for single images using dynamic skin color modeling. Pattern recognition, 2010. 43(4): p. 1413-1420.

[27] Do, H.-C., J.-Y. You, and S.-I. Chien, Skin color detection through estimation and conversion of illuminant color under various illuminations. Consumer Electronics, IEEE Transactions on, 2007. 53(3): p. 1103-1108.

[28] Shoyaib, M., et al., Skin detection using statistics of small amount of training data. Electronics letters, 2012. 48(2): p. 87-88.

[29] Cheddad, A., et al., A skin tone detection algorithm for an adaptive approach to steganography. Signal Processing, 2009. 89(12): p. 2465-2478.

[30] Jones, M.J. and J.M. Rehg, Statistical color models with application to skin detection. International Journal of Computer Vision, 2002. 46(1): p. 81-96.

[31] Ebner, M., Color constancy. Vol. 6. 2007: John Wiley & Sons.

[32] Rahman, Z.-u., D.J. Jobson, and G.A. Woodell. Resiliency of the multiscale retinex image enhancement algorithm. in Color and Imaging Conference. 1998. Society for Imaging Science and Technology.

[33] Ebner, M. and J. Hansen, Depth map color constancy. Bio-Algorithms and Med-Systems, 2013. 9(4): p. 167-177.

[34] Gijsenij, A., T. Gevers, and J. Van De Weijer, Generalized gamut mapping using image derivative structures for color constancy. International Journal of Computer Vision, 2010. 86(2-3): p. 127-139.

[35] Jobson, D.J., Z.-U. Rahman, and G.A. Woodell, Properties and performance of a center/surround retinex. Image Processing, IEEE Transactions on, 1997. 6(3): p. 451-462.

[36] Jobson, D.J., Z.-U. Rahman, and G.A. Woodell, A multiscale retinex for bridging the gap between color images and the human observation of scenes. Image Processing, IEEE Transactions on, 1997. 6(7): p. 965-976.

Fusion of Multispectral And Full Polarimetric SAR Images In NSST Domain

Gh. S. El-Tawel *ghada_eltawel@ci.suez.edu.eg*
Computer Science Dept., Faculty of Computers and Informatics,
Suez Canal University,
Ismailia, Egypt

A. K. Helmy *akhelmy@narss.sci.eg*
Data Reception, Analysis and Receiving Station Affairs Division,
National Authority for Remote Sensing and Space Sciences, Cairo, Egypt

Abstract

Polarimetric SAR (POLSAR) and multispectral images provide different characteristics of the imaged objects. Multispectral provides information about surface material while POLSAR provides information about geometrical and physical properties of the objects. Merging both should resolve many of object recognition problems that exist when they are used separately. Through this paper, we propose a new scheme for image fusion of full polarization radar image (POLSAR) with multispectral optical satellite image (Egyptsat). The proposed scheme is based on Non-Subsampled Shearlet Transform (NSST) and multi-channel Pulse Coupled Neural Network (m-PCNN). We use NSST to decompose images into *low frequency* and *band-pass* sub-band coefficients. With respect to low frequency coefficients, a fusion rule is proposed based on local energy and dispersion index. In respect of *sub-band* coefficients, m-PCNN is used to guide how the fused *sub-band* coefficients are calculated using image textural information.

The proposed method is applied on three batches of Egyptsat (Red-Green-infra-red) and radarsat2 (C-band full-polarimetric HH-HV and VV-polarization) images. The batches are selected to react differently with different polarization. Visual assessment of the obtained fused image gives excellent information on clarity and delineation of different objects. Quantitative evaluations show the proposed method can superior the other data fusion methods.

Keywords: Multi-spectral Data Fusion, POLSAR, NSST, m-PCNN.

1. INTRODUCTION

Image fusion is a process of incorporating different images originating from different sources to create more reliable information than that from individual sources, and recently it has received great attention in the remote sensing field.

In the processing of optical images, many land cover types and surface materials are identical in their spectral characteristics. This leads to great difficulty in image segmentation, classification and feature extraction [1] [2]. Usually, Optical satellites use different sensors (visible, near infrared and shortwave infrared) to form images of the earth's surface. Different targets reflect and absorb in a different way at different wavelengths. Thus, their spectral signatures can characterize the targets.

Synthetic aperture radar (SAR) system hits the objects over the earth with a guided microwave [3], then the object either absorb or scatter these waves in all directions. Absorption or scattering of incident wave depends mainly on the physical characteristics of the object. The SAR system records only part of the scattered wave in the direction of the receiving antenna. Some SAR systems transmit and receive waves with different polarization (POLSAR). In this, the system

records the polarization of returned waves and measures both the intensity and phase of the backscattered waves. It mainly characterizes intrinsic structural and dielectric properties of the target [3].

Therefore, the imaging system for an optical satellite images and radar satellite signals are evidently different. The optical images are mainly characterized by spectral resolution, which is a measure of its ability to discriminate features in electromagnetic spectrum [4]. Polarimetric Radar images provide a tool to identify different features based on dielectric properties and surface roughness. Integration of spectral characteristics of an object, devised from multispectral images, and physical properties (and surface roughness) originated from Polarimetric SAR images will provide a great role in many remote sensing applications.

In this paper, we propose a new scheme for integrating Multispectral optical images (MS) and Multi-polarization POLSAR images. Non-Sampled Shearlet Transform (NSST) is used to decompose the input images into low and band-pass sub-bands. For low frequency coefficients, we introduce an adaptive weight fusion structure based on regional local energy and local image texture features. Different textural factors, gradient, entropy, and spatial frequency, are taken as multiple stimuli to Multi channel Pulse Coupled Neural Network (m-PCNN) to guide the fusion process of *band-pass* sub-bands. The rest of the paper is organized as follows: Section 2 and 3 discs in the short-term the theory of the NSST and m-PCNN. Section 3 demonstrates the main frame of the proposed scheme and pronounces the fusion algorithm. Experimental results and the evaluations are discussed in Section 4. We present the conclusions in the last section.

2. POLARIMETRIC SYNTACTIC APERTURE RADAR (POLSAR)

POLSAR images uses that fact that the status of the received scattered signal reflects the characteristics of the illuminated objects such as roughness and dielectric constants [5]. Accordingly the polarimetric images can be used efficiently to recognize these properties. The scattering matrix [S] is a kind of relation between the incident and scattered wave [5,6] and is being expressed by:

$$\begin{bmatrix} E_V^r \\ E_V^r \end{bmatrix} = [S] \begin{bmatrix} E_V^t \\ E_V^t \end{bmatrix} = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix} \begin{bmatrix} E_V^t \\ E_V^t \end{bmatrix} \qquad (1)$$

Where $E_H^r, E_V^r, E_H^t, and\ E_V^t$ are the received and transmitted electric fields of corresponding polarizations respectively, and S_{mn} is the matrix elements and defined as:

$$S_{mn} = |S_{mn}| e^{\,i\varphi mn} \qquad (2)$$

The incident and scattered wave are a complex quantity (amplitude and phase) and usually expressed in polarization term (the direction of incident / received wave). Four different combinations of transmitted and received polarizations are listed below.

HH: horizontal transmission and reception.
HV: horizontal transmission and vertical reception.
VV: vertical transmission and vertical reception.
VH: vertical transmission and horizontal reception.

A SAR system used in this paper (RADARSAT-2) has a single antenna for both transmission and reception, the relation $S_{HV} = S_{VH}$ [7] holds for the rest of this paper.

From equation 2, the main parameters that characterized POLSAR data are the amplitudes ($|S_{HH}|$, $|S_{HV}|$, $|S_{VV}|$) and the phases (φ_{HV} and φ_{VV}). The phase values are not absolute values [7], it is a relative to a certain phase plan, generally HH-polarization element is chosen as a reference

phase. In this paper, we use the first three parameters, $|S_{HH}|$, $|S_{HV}|$, $|S_{VV}|$, to be fused with multispectral Egypt-sat data (see app-1).

3. THE NON-SUBSAMPLED SHEARLET TRANSFORM (NSST)

Multi-resolution analysis tools, Discrete Wavelet Transform (DWT), have been widely applied to image fusion [8, 9]. DWT are mainly depends on multi-scale geometric analysis, and has many advantages such as localization and direction. On the other hand, the wavelet transform suffers from imperfect directionality (directional selectivity is very limited and cannot get optimal detail information) moreover; it is not shift-invariant, result in degraded information and bad fusion output [10]. In order to get better signal representation researchers introduce new signal analysis tools and used extensively in image fusion, including: Curvelet [11], Ridgelet [12], Contourlet [10,13], and so on.

Easley et al. proposed Non-Subsampled Shearlet Transform (NSST) [14], which is the mixture of non-subsampled Laplacian pyramid transform and different shearing filters. NSST provides a multi-scale and multi-directional framework which decomposes into one low-frequency sub-band (signifies the approximation component of the source image) and a series of directional band pass sub-bands. NSST also satisfies the prerequisite of the shift-invariance property. So it can capture more further information on different directional sub bands than that of the wavelet transform and contourlet transform. The decomposition of shearlet is close to contourlet transform, but it has an advantage over contourlet transform that the number of directions in NSST for the shearing filter is non-limited.

Additionally, inverse contourlet requires inverting directional filter banks, instead of a summation of the shearing filter in case of inverse shearlet transform. Consequently the implementation of shearlet is more efficient computationally [14].

Through this work, we used NSST to decompose the input images into low and sub-bands coefficients, apply a fusion rule followed by the inverse of NSST to construct the fused image.

4. PULSE COUPLED NEURAL NETWORK

The Pulse Coupled Neural Network “PCNN” is a neural model with single layer architecture. To model an image with a PCNN, we consider the following: the input neurons represent image pixels, pixel’s information (e.g. Intensity or texture) are represented as an external stimulus received by each neuron and the relation between neighbored pixels is represented as internal stimuli fed to each neuron. In this model each neuron connects with its surrounded neighbors. PCNN uses an internal activation system to accumulate the stimuli until it surpasses a dynamic threshold, resulting in a pulse output. Images generated at different iterations indicate the fine details of the input image (edges and small objects). The PCNN model is fully described in [15, 16].

Many researchers use PCNN for data fusion for instant Wang and Ma design single PCNN for medical image fusion [17, 18]. Miao introduces an adaptive system for image fusion with different spatial resolution by the adaptive linking coefficient of PCNN [19]. Others integrate PCNN with multi-layer decomposition to get the fused image [20, 21].

Recently a parallel version of PCNN is introduced in which many PCNNs are working in parallel, in this each network operates on a separate channel of the input [17]. We present an image fusion method based on dual pulse couple neural network [22]. Wang presented an excellent review of application of PCNN [23].

Our proposed scheme of data fusion is based on m-PCNN that is a modified version of PCNN. This model is proposed by Wang and Ma, [17, 18]. m-PCNN can extend the number of inputs that depends on practical request. We use some texture information as multiple stimuli of m-PCNN, and the output is used as a weight guide for the fusion process.

5. GENERAL FRAMEWORK OF MULTISPECTRAL-POLSAR FUSION ALGORITHM

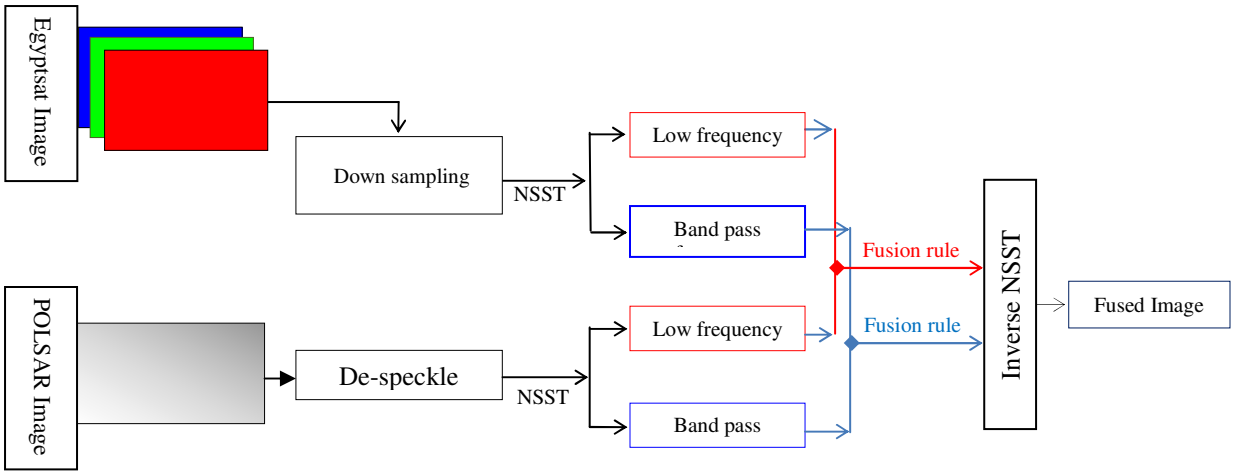


FIGURE 1: Block diagram of image fusion based on the NSST.

Through this paper, we assume that Egyptsat, and POLSAR images have been co-registered, and noted by R (x, y), M (x, y) respectively. F (x, y) is the output fused image. Figure-1 demonstrates the proposed framework of the image fusion process. As a summary, the fusion approach is listed in the following steps.

1- POLSAR image has been de-speckled and the Egyptsat image is down-sampled (from 7.8 to 7 meters) to match the POLSAR spatial resolution.
2- *Low frequency* and *band-pass* sub-bands coefficients of the source images are calculated using NSST.
3- The *low-frequency* sub-band coefficients and the directional *band-pass* sub-bands coefficients of the source images are merged according to specific rules, the rules will be introduced in the next sections.
4- Calculate the inverse NSST to get the fused image F (x, y).

5.1 Fusion Rule for Lowpass Coefficients

As the *lowpass* sub-band coefficients mainly retain the main energy and represent the approximation component of the source images, fusion procedures should be adopted to preserve this information. Many authors process the *lowpass* sub-band coefficients use direct averaging rule [24], which is simple, however, this scheme always results in a low-contrast effect, due to fact that both information (3-bands) of POLSAR and multispectral images are complementary and both of them are desirable in a fused image. A new average weighted formula has been presented to model fusion hierarchy based on regional local energy and local image texture features.

It is difficult to know or even estimate, which bands from POLSAR and multispectral should be fused together (in our case we have three-bands for multispectral image and three-bands represent POLSAR image). Figure 2 shows a schematic diagram of our method to fuse *lowpass* coefficients, and the procedures are summarized as follows:

1- For all bands, the local energy (equation-5) is calculated for each coefficient, using its neighborhood (5 x 5 in our case). This is done for the multispectral and POLSAR bands.

2- Coefficient retains maximum value of energy from POLSAR is fused with a coefficient that has the least energy from multispectral (equation-3). This represents the first-band first-fused coefficient.
3- To get the second-band first-fused coefficient, next coefficient with maximum energy from POLSAR is fused with preceding minimum energy from multispectral coefficient.
4- Finally, to get the third one, coefficient with minimum energy from POLSAR is fused with maximum energy from multispectral images. The mathematical notation of the process calculations is described below.

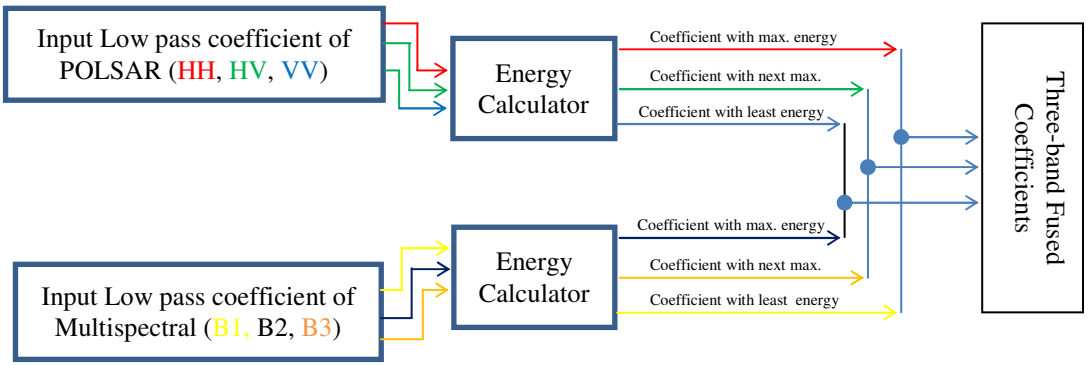


FIGURE 2: Block Diagram of Image Fusion rules for Low Coefficients.

The coefficient of the fused image at location (x, y) can be calculated by:

$$I_F\,(x,y) = \frac{w_1\, I_R(x,y) + w_2 I_M(x,y)}{2} \qquad (3)$$

Where:
$I_R(x,y), I_M(x,y)$, and $I_F(x,y)$ denote the lowpass *subband* coefficient located at(x,y) for POLSAR, multispectral and fused images respectively.

The value of $I_R(x,y)$ is chosen among the coefficients of three POLSAR bands according to the following:

$$I_R(x,y) = C(maximum\ (en_{hh}, en_{hv}, en_{vv})) \qquad (4)$$

Where: C (…) is the *lowpass suband* coefficients.

$en_{xx}(energy)$ is a parameter used to measure the textural uniformity of an image, $hh, hv\ or\ vv$, and calculated as follows:

$$en\,(x,y) = \frac{1}{n \text{ x } n} \sum_{m=-(n-1)/2}^{m=(n-1)/2} \sum_{r=-\frac{n-1}{2}}^{r=\frac{n-1}{2}} C^2(x+m, y+r) \qquad (5)$$

Where *C* is the *lowpass* coefficients of *HH, HV or VH*, and *n* and *m* are those defined the neighborhood areas.

In contrast the value of $I_M(x,y)$ is picked from the coefficients of three multispectral bands according to the following: $I_M(x,y) = C(minimum\,(en_{b1}, en_{b2}, en_{b3})\,)$, again *C (…)* is the *lowpass suband* coefficients.

The weights in the equation (3) can be calculated as follows:

$$w_1 = \frac{maximum\ (en_{hh}, en_{hv}, en_{vv})}{minimum\ (en_{b1}, en_{b2}, en_{b3})} \qquad (6)$$

$$w_2 = \frac{1}{w_1} \qquad (7)$$

In this $if\ w_1 > 1$: *i.e.)* maximum energy picked from POLSAR image is greater than the minimum energy of multispectral image, the proposed weighting strategy leads to maximize the contribution of POLSAR image and minimize that of multispectral image.

On the other hand $if\ w_1 < 1$ *i.e.)* minimum energy picked from multispectral image is greater than the maximum energy of POLSAR image, again our strategy leads to maximize the contribution of multispectral image and minimize that in POLSAR image. The previous procedures are repeated with next maximum and minimum values to obtain 3-bands fused image.

In any case, the high-energy contribution is maximized, while the minimum energy contribution minimizes through generation of the fused image. This procedure can overcome the low contrast drawback of weighting average scheme.

To fine-tune the results of fused coefficients and to obtain better effect than that explained earlier, we modified equations (6, 7) taking into account texture information when calculating the weight factor. We added a dispersion index "D" which is a normalized degree of dispersion of a probability distribution: it is a measure used to enumerate whether a set of observed occurrences is clustered or dispersed with respect to a standard statistical model. It is defined as the ratio of the variance "δ^2" to mean "μ" and calculated as an average mean for both images.

$$D = \frac{\delta^2}{\mu} \qquad (8)$$

Then the new weight

$$\widehat{w}_1 = w_1 + \frac{D_S}{D_m} \qquad (9)$$

Where $D_S\ and\ D_m$ are dispersion index of POLSAR and multispectral images.

5.2 Fusion Rule for High Frequency Sub-band Coefficients

The image edges, corners and other fine details are concentrated in the high-frequency components obtained from shearlet transform. Thus, the clarity and distortion of the fused image depend mainly on how these components are fused. Voting strategy has been popularly applied to composite the high-frequency components. It relies on constructing decision map using different parameters to regulate where the fused coefficients are from image 'A' or image 'B'.

In most existing fusion algorithms, usually, some textural information is used to decide from which the fused image come from [25]. Features such as the entropy, gradient, variance, and the spatial frequency [26, 27] can represent image texture information. Essentially, it can reflect detailed information in different ways, and can be used to discriminate sharp and indistinct regions. These textural measurements can be used independently [26, 27] or may be joint using some specific rules [28]. In our method, we present some texture information (the entropy, gradient and spatial frequency) as an indicator of weighting factor in constructing a fusion rule. These different texture factors are taken as multiple stimuli of m-PCNN. Then the output determines the fusion weight according to the values coming from PCNN that reflect the overall image clarity. The whole process shown in figure 3 and can be summarized as follows:

For the current subband, let $C_I^{l,k}(x,y)$ be the band-pass coefficient at a location (x,y) in the l^{th} band at k^{th} level, I represents HH, HV, VV, b1, b2 or b3.

For the same subband, calculate the maximum gradient($GR_{\max}$), entropy ($EN_{\max}$), and spatial frequency ($Sf_{\max}$) values among all bands.

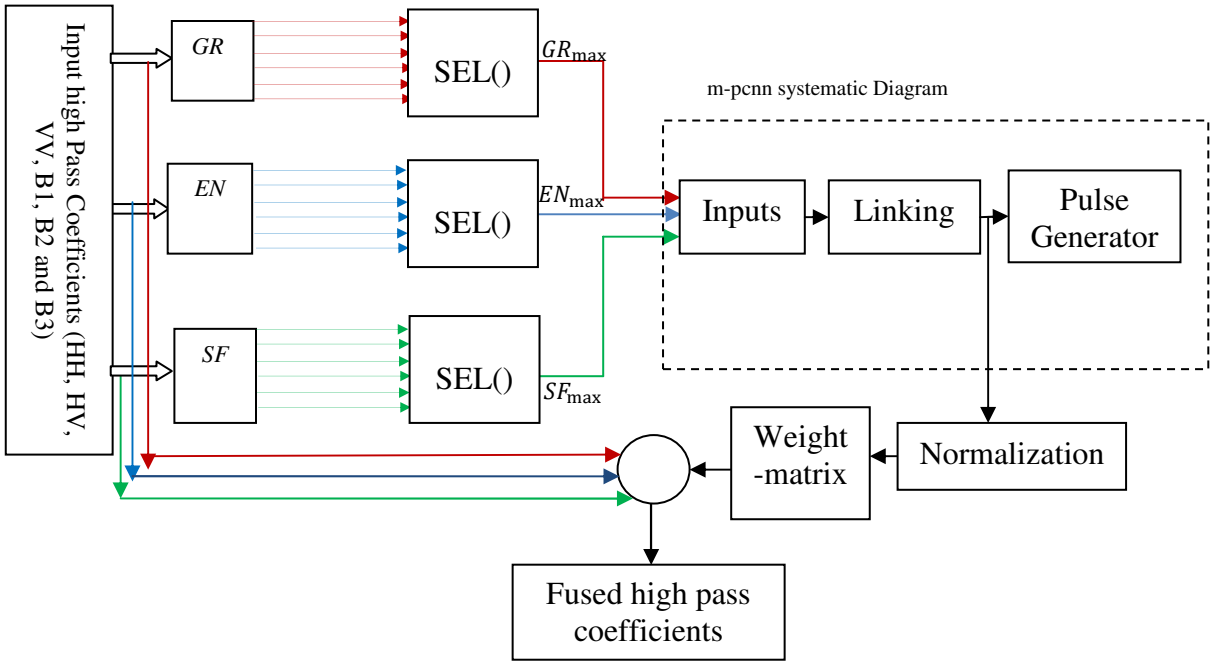


FIGURE 3: Block diagram of image fusion for subbands coefficients.

$$GR_{\max} = maximum\ (\ GR(C_I^{l,k}(x,y))\quad \forall(I) \qquad (10)$$

$$EN_{\max} = maximum\ (\ EN(C_I^{l,k}(x,y))\quad \forall(I) \qquad (11)$$

$$SF_{\max} = maximum\ (\ SF(C_I^{l,k}(x,y))\quad \forall(I) \qquad (12)$$

Where: $GR(C_I^{l,k}(x,y))$ is the gradient of the *high-pass* coefficient at location (x,y) in the l^{th} subband at k^{th} level. Similarly $EN(C_I^{l,k}(x,y))\ and\ sf(C_I^{l,k}(x,y))$ are the largest value of entropy and spatial frequency respectively of the *high-pass* coefficients.

Maximum gradient, entropy, and spatial frequency will be used as different features of image to motivate dual-channel m-PCNN. The fusion will be done between the three winner bands (bands that have maximum gradient, entropy and spatial frequency). We should keep track which bands got those maxima. The coefficient of the fused image at location (x, y) can be calculated by:

$$CI_F\ (x,y) = \frac{w}{3}\sum_{p=1}^{3} I_p\ (x,y) \qquad (13)$$

Where:
$I_p(x,y)$ Denote the high-pass *subband* coefficient located at(x,y) with a maximum value of gradient, entropy, and spatial frequency. W is the weight factor outputs from m-PCNN.

5.3 Assessment Criteria

The visual contrast between the fused images and original images are conducted. Furthermore, quantitative analysis is also applied to the results of different fusion algorithms in terms of correlation coefficients, entropy, average gradient [29] and the $Q^{AB/F}$[30]. The clarifications of these measures are discussed below.

The *correlation* reflects the amount of similarity of two images. The average gradient mirrors the difference between image structure (sharper image normally has a greater average gradient), the entropy specifies the overall randomness level in the image (higher value of entropy, more detailed information will be contained in the image), While $Q^{AB/F}$ measures the amount of edge information transferred from the source images to the fused image using a Sobel edge detector, its larger value, imply better fusion result is.

In the two images f(x,y) and B(x,y) of size M x N, the correlation coefficient of each band is defined as:

$$Correlaton\ coefficient = \frac{\sum_{\forall x}\sum_{\forall y}[(f(x,y) - E_A) \times (B(x,y) - E_B)]}{\sqrt{\sum_{\forall x}\sum_{\forall y}\left[(f(x,y) - (E_f)^2\right] \times \sum_{\forall x}\sum_{\forall y}\left[(B(x,y) - (E_B)^2\right]}} \quad (14)$$

Where E_f and E_B are the mean of two images, respectively.

$$average\ gradient = \frac{1}{(M-1)(N-1)}\sum_{\forall M}\sum_{\forall N}\sqrt{\frac{1}{2}\left[\left(\frac{\partial(f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial(f(x,y)_2}{\partial y}\right)\right]} \quad (15)$$

$$entropy = -\sum_{\forall l} p(l)\ln p(l) \quad (16)$$

Where $p(l)$ means the probability of the gray value (l) appearing in the image.

$$Q^{AB/F} = \frac{\sum_{n-1}^{N}\sum_{m-1}^{M}(Q^{AF}(n,m)W^A(n,m) + Q^{BF}(n,m)W^B(n,m))}{\sum_{n-1}^{N}\sum_{m-1}^{M}(W^A(n,m) + W^B(n,m))} \quad (17)$$

Where $Q^{AF}(n,m) = Q_g^{AF}(n,m)Q_\alpha^{AF}(n,m)$; $Q_g^{AF}(n,m)$ and $Q_\alpha^{AF}(n,m)$ are the edge strength and orientation preservation values respectively; n, m represent the image location; and N ,M are the size of images. $Q^{BF}(n, m)$ is similar to $Q^{AF}(n, m)$. $W^A(n,m)$ and $W^B(n,m)$ reflect the importance of $Q^{AF}(n, m)$ and $Q^{BF}(n, m)$,respectively. The dynamic range of $Q^{\frac{AB}{F}}$ is [0 1], and it should be as close to 1 as possible.

6. EXPERIMENTAL RESULTS

Data fusion algorithms in literatures take advantage of the complementary spatial/spectral resolution characteristics of multi-spectral and panchromatic data for producing spatially enhanced multi-spectral observations. In our instance, spatial resolution is almost the same while the complementary information resides in spectral bands. We seek to integrate information from Red, Green and infrared bands (exist in Egyptsat images) with other information originated from POLSAR data that represent geometric and physical characteristics of objects

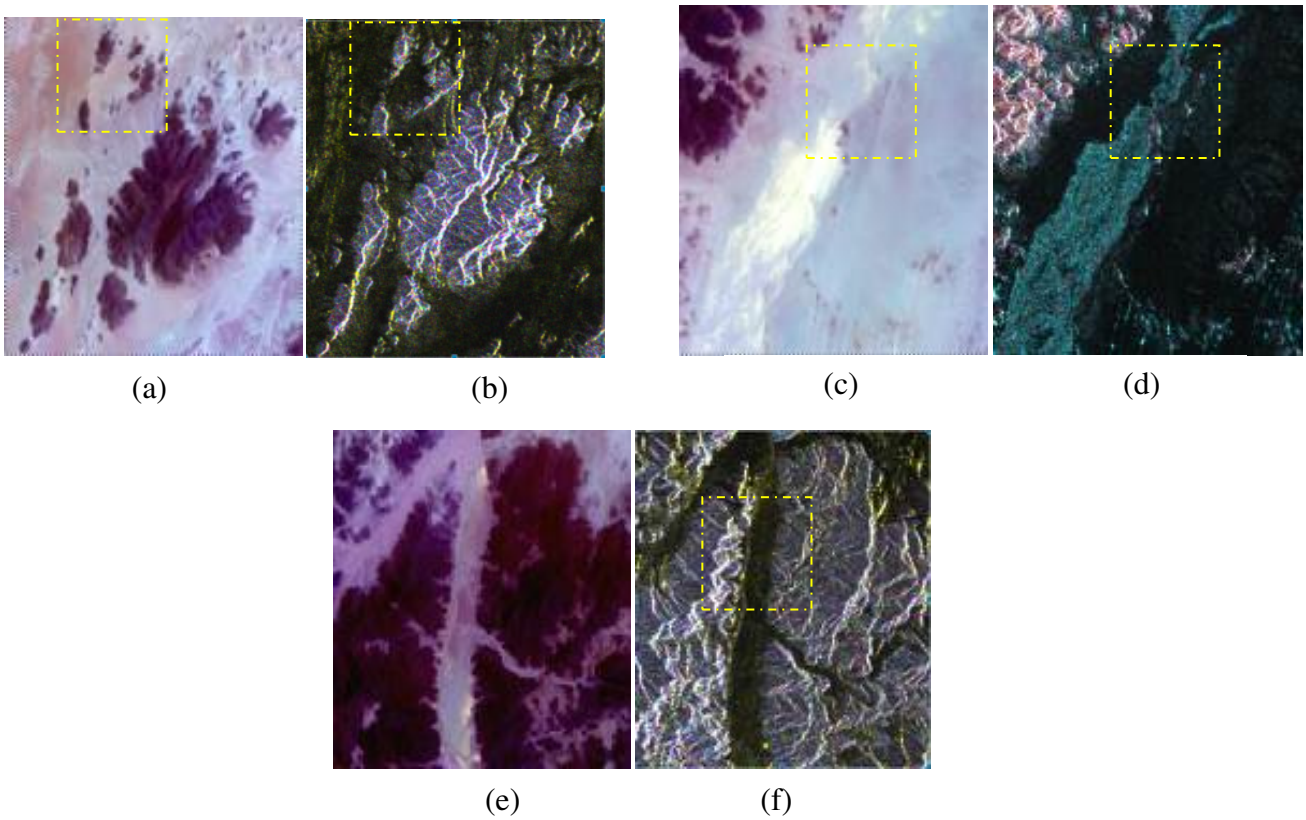
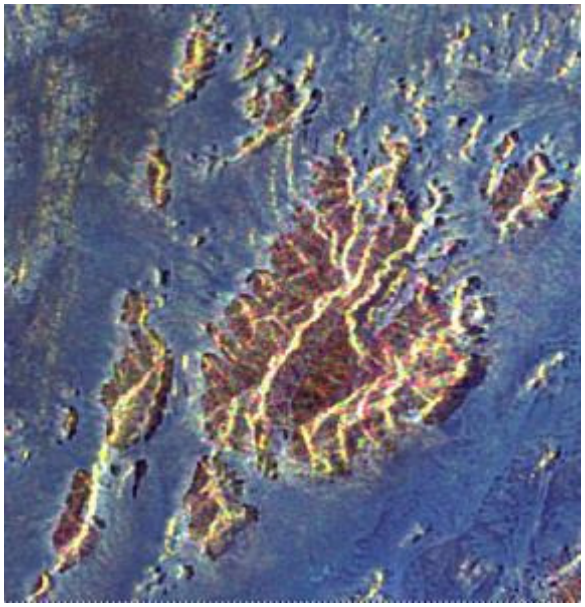


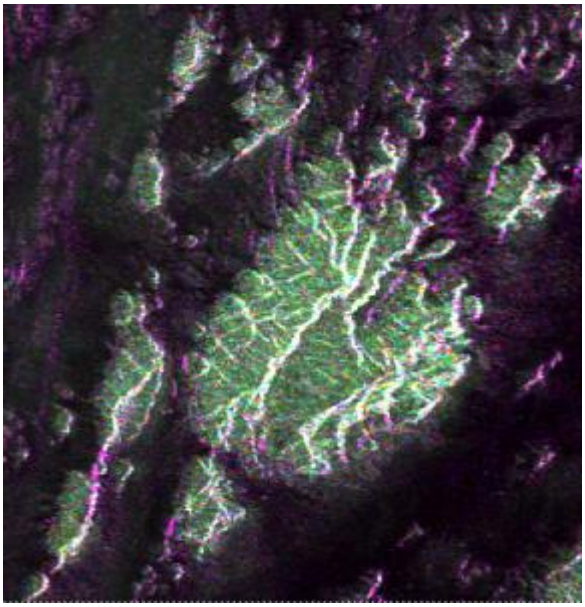
FIGURE 4: Three Input batches a-b-c: Multispectral images (in RGB color composite). b,d,f: C-band raw polarimetric SAR image (R=HH,G=HV,B=VV), *copyright (MDA)*

The proposed scheme is used to merge multispectral, Egyptsat, image (Band 1, Band 2 and Band 3-see-App1), with scattering matrix of POLSAR with *HH, HV and VH* polarization taking into account that *HV=VH*. The POLSAR data set used is C-band Radarsat-2 (RS-2) data in a full parametric mode. Spatial resolution of POLSAR is 7 m; while Egyptsat is 7.8 m., Speckles of POLSAR are reduced using enhanced LEE filter [31], while the spatial resolution of Egypt sat is down-sampled to match that of POLSAR. The data set has been accurately co-registered. To illuminate the results, three batches (pat-1, pat-2 and pat-3) of images are used as shown in figure 4. The batches are carefully chosen to be sensitive to radar-objects interaction (i.e. vertical and horizontal polarized). Comparing the proposed scheme with traditional methods [32] used in remote sensing such as principal component, brovey, IHS… is not appropriate in our case, since these techniques aim to merge multispectral with PAN SAR images.

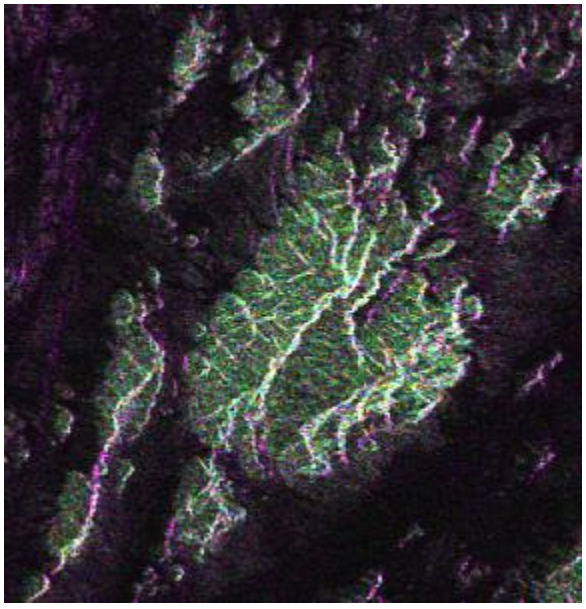
In order to evaluate the proposed method we had made a comparison with three fusion methods, SIST-based [33], PCNN-based [22], and Contourlet-based [13]. In these methods, the fusion process is performed separately with respect to S_{HH} ,S_{HV} and S_{VV} , then the fused results are displayed in RGB. Moreover, the proposed scheme is compared with that proposed by Lei Wang that has the ability to fuse two multispectral images [34]; he used Hidden Markov Tree (HMT) in SIST domain to perform fusion process.



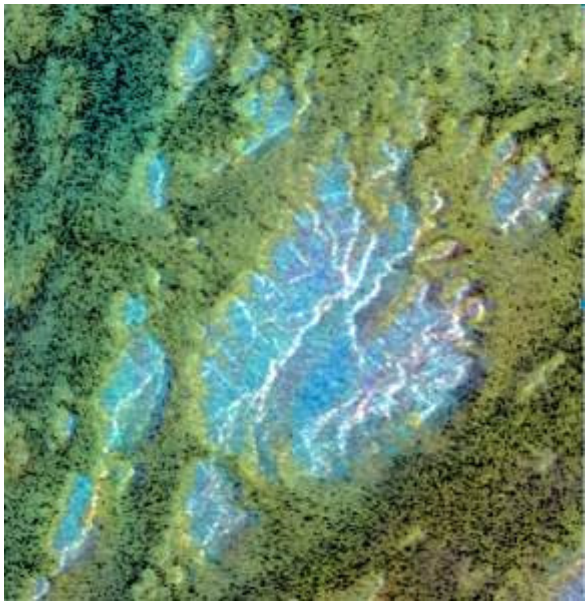
(a)



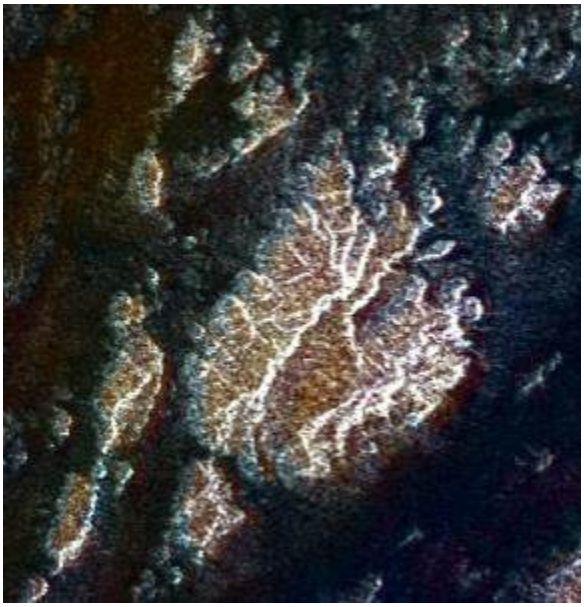
(b)



(c)



(d)



(e)

FIGURE 5: The fusion results: of Fig. 4 (a) and (b): (a)–(e) fused images using the proposed method, SIST-based, Contourlet-based, HMT-based and PCNN-based, respectively.

Figure 4 shows the source images used in this research. We use three batches of images in the fusion experiment; the images are selected to react differently with different POLSAR polarization. One group focuses on a hilly area, second emphasis flat region and the last one contain both of them. Figure 5 shows the fusion results of batch-1. Generally, although all methods inject fair information of the source images into the fused image, but they fail to achieve acceptable transfer of information, specifically that related to low frequency regions, from input images to the fused image. To clarify the visual assessment, figure 6 shows closer look of these images.

Closer looks at these results disclose the following:

As can be seen in figures 5 and 6, the fused image, which is obtained by HMT-based, presents low contrast and vague attendance; Moreover, it showed a loss of edge information and mixing of color in low frequency sub-bands, this due to only use of intensity component in the fusion process [34].

The fused image, SIST-based Method, lost information to some extent in low frequency subbands. In addition, edges suffer from over smoothness.

The fused images outs from Contourlet-based appear noticeable noise at high frequency regions (the edges), as it does not have shift-invariant, which leads humble visual effects.

PCNN method shows an improvement in visual effects, although it suffers from overall haziness appearance. The proposed methods introduced significant migration of information from input images to the fused one in both low and sub-bands frequency.

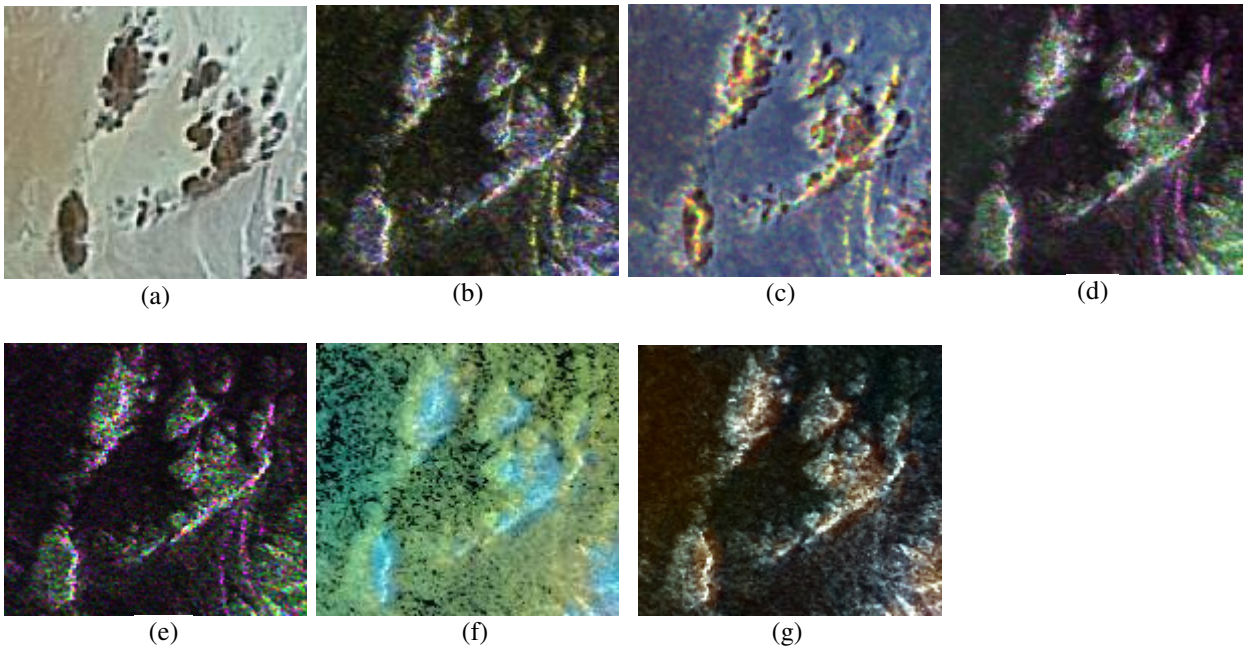


FIGURE 6: Zoomed shot of the fusion results: (a) and (b): Multispectral and POLSAR input images. (c)– (g): zoomed area of fused images using the proposed method, NSST-based, Contourlet-based, HMT-based and PCNN-based respectively.

TABLE 1: Comparison of fusion results with different fusion methods (batch-1, the radar image is sensitive to vertical polarization, it is a steep region)

	Proposed Model	SIST-Based	Contourlet-Based	HMT-Based	PCNN-Based
Correlation	0.88	0.85	0.83	0.8	0.83
Average-gradient	8.644	8.1	8.8	7.8	8.4
Entropy	8.457	8.2	8.5	7.38	8.1
$Q^{AB/F}$	0.792	0.77	0.73	0.61	0.69

According to the quantitative evaluation, table-1 lists different metric measures used through this study, best values achieved by the proposed model regarding the amount of information, transferred from input images to the output, and the strength of the edges. Zoomed areas of the second and third batches of the image are shown in figures 7, 8. Tables 2, 3 show the quantitative measures of fused images.

TABLE 2: Comparison of fusion results with different fusion methods (batch-2, the radar image is sensitive to horizontal polarization, it is almost flat region).

	Proposed Model	SIST-based	Contourlet - Based	HMT-Based	PCNN-Based
Correlation	0.88	0.81	0.78	0.7	0.785
Average-gradient	6.192	6.1	5.87	5.4	5.7
Entropy	7.069	7	7.1	6.5	6.9
$Q^{AB/F}$	0.75	0.7	0.66	0.711	0.74

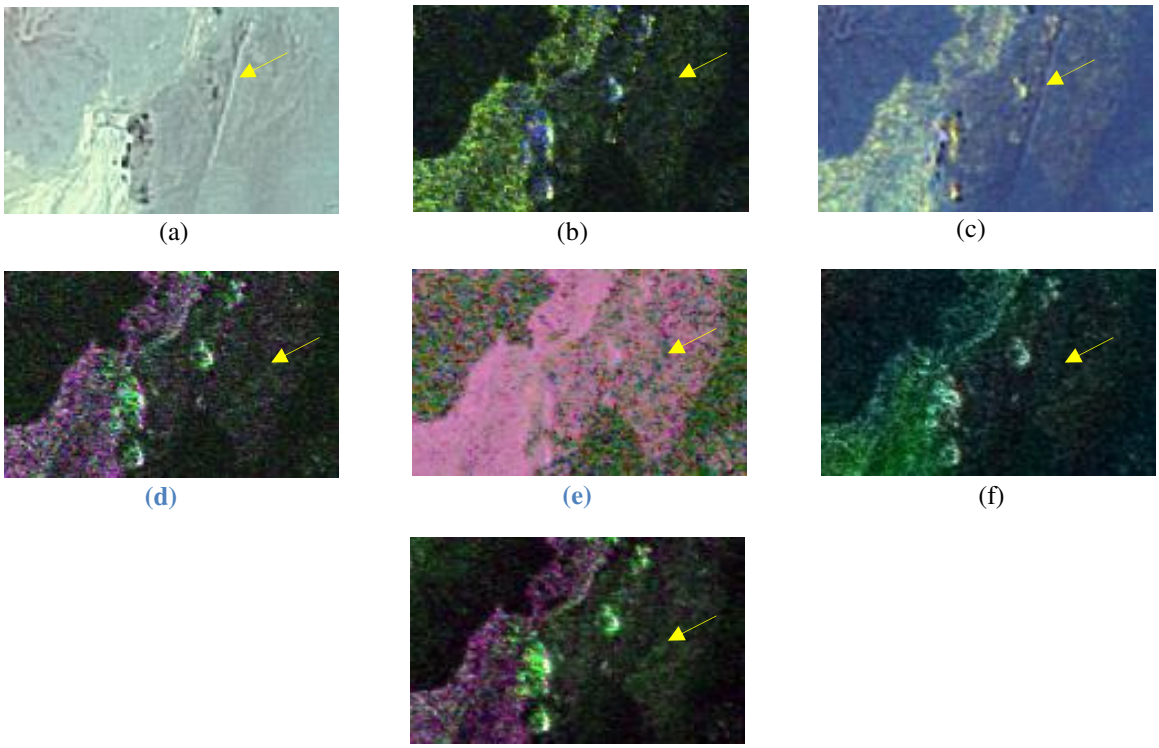


FIGURE 7: Zoomed area of the second batch of images: (a) and (b): Multispectral and POLSAR input images. (c)– (g): fused images using the proposed method, NSST-based, Contourlet-based, HMT-based and PCNN-based respectively.

Figure 7, 8 give same indication, some high frequency components eliminated from fusion results; it was pointed by the yellow arrow in figure 7. In Figure 8, it is found that the valley structures are fully preserved in the proposed method, while the color information is variegated between multispectral and POLSAR images.

TABLE 3: Comparison of fusion results with different fusion methods (batch-3, the radar image is sensitive to both horizontal and vertical polarization).

	Proposed Model	**SIST**	**Contourlet - Based**	**HMT-Based**	**PCNN-Based**
Correlation	0.91	0.91	0.88	0.73	0.86
Average-gradient	7.2	6.87	6.99	7.04	5.9
Entropy	8.15	7.59	8.1	7.9	6.5
$\mathbf{Q^{AB/F}}$	0.68	0.679	0.523	0.61	0.59

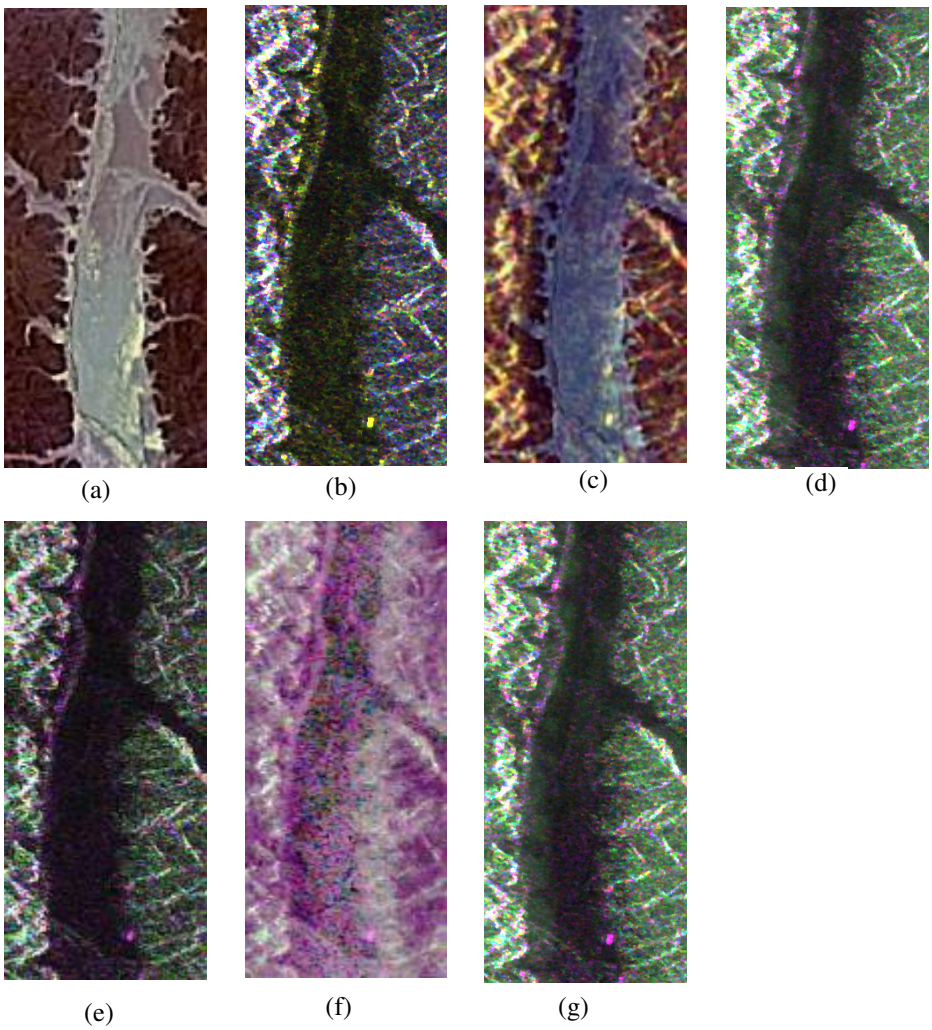


FIGURE 8: Zoomed area of the third batch of images: (a) and (b): Multispectral and POLSAR input images. (c)– (g): fused images using the proposed method, NSST-based, Contourlet-based, HMT-based and PCNN-based respectively.

7. CONCLUSION

This research investigates the fusion process of full polarimetric POSAR data, RS2, with Multispectral optical imagery (Egyptsat). By applying the fusion process, we obtain a new image that can be considered neither optical nor POLSAR. It is a synthesized image that is produced for better image understanding. To meet this requirement, we propose a new weighting average scheme based on NSST and m-PCNN. Firstly, input images are transformed into *low* and band-pass sub-bands. Low frequencies of both images are fused to each other by relating maximum and minimum energy of both images-bands with a reciprocal way, in addition, the fusion process takes into account a texture distribution by adding a dispersion index to the fused coefficients. Secondly, m-PCNN is used to guide the fusion process of band-pass sub-bands by incorporating edge information measurements. The experimental results indicate that the fused image has strengthened object structure detail.

ACKNOWLEDGMENTS

We express our deepest thanks and appreciation to the Canadian Space Agency (CSA) for providing the Radar Sat 2 images for this study under the agreement of the project ID5128. We also acknowledge the National Authority for Remote Sensing and Space Sciences for resourcing this work.

8. REFERENCES

[1] King, RL. 2000. "A challenge for high spatial, spectral, and temporal resolution data fusion" In: Proceedings, IEEE international geoscience and remote sensing symposium, IGARSS no. 6: 2602–2604. doi: 10.1109/IGARSS.2000.859654.

[2] Zhu, X., and Yin L. 2001. "Data fusion of multi-sensor for petroleum geological exploration" In: Proceedings, international conferences on infotech. and info-net, ICII, Beijing no.1: 70–75. doi: 10.1109/ICII.2001.982723.

[3] Sabry R., and Vachon P. W. 2014. "A Unified Framework for General Compact and Quad Polarimetric SAR Data and Imagery Analysis." IEEE Transactions on Geoscience and Remote Sensing 52(1): 582-602. doi:10.1109/TGRS.2013.2242479.

[4] Campbell, J. B. 2002. Introduction to Remote Sensing, New York London: The Guilford Press.

[5] Giorgio, L. 2014. "A novel approach to polarimetric SAR data processing based on Nonlinear PCA" Pattern Recognition no. 47:1953–1967. doi: 10.1016/j.patcog.2013.11.009.

[6] Kajimoto, M. 2013. "Urban density estimation frompolarimetric SAR images based on a POA correction method" IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 6(3): 1418–1429. doi: 10.1109/JSTARS.2013.2255584.

[7] Jong-Sen, Lee, and Eric Pottier. 2009. Polarimetric Radar Imaging: From Basics to Applications. CRCPress: BocaRaton.

[8] Huang, W., and Jing Z.L. 2003. "Evaluation of focus measures in multi-focus image fusion" Pattern Recognition Letter no. 28: 493–500. doi: 10.1016/j.patrec.2006.09.005.

[9] Shutao, Li., and Bin Yang. 2008. "Multi-focus image fusion using region segmentation and spatial frequency" Image and Vision Computing no. 26: 971–979. doi: 10.1016/j.imavis.2007.10.012.

[10] Zhang, Q., and Guo, B.L. 2009. "Multifocus fusion using the non subsampled contourlet transform" Signal Processing 89(7): 1334–1346. doi: 10.1016/j.sigpro.2009.01.012.

[11] Starck, J.L., Candes, E.J., and Donoho, D.L. 2002. "The curvelet transform for image denoising" IEEE Transaction on Image Processing 11(6): 131–141. doi: 10.1109/TIP.2002.1014998.

[12] Candes, E. 1998. "Ridgelets: Theory and Applications", Technical report, Department of Statistics, Stan-ford University, https://statistics.stanford.edu/sites/default/files/1998-17.pdf.

[13] Huafeng Li., Chai, Yi., and Zhaofei Li. 2013. "Multi-focus image fusion based on nonsubsampled contourlet transform and focused regions detection" Optik - International Journal for Light and Electron Optics 124(1): 40–51. doi: 10.1016/j.ijleo.2011.11.088.

[14] Easley, G., Labate, D., and Lim, W.Q. 2008. “Sparse directional image representations using the discrete shearlet transform” Applied and Computational Harmonic Analysis 25 (1): 25–46. doi: 10.1016/j.acha.2007.09.003.

[15] Ranganath , H. S. and Kuntimad, G., 1999. “Object Detection Using Pulse Coupled Neural Networks” IEEE Transactions on Neural Networks 10, (3): 615–620. doi: 045-9227(99)03181-1.

[16] John L. Johnson and Mary Lou Padgett. 1999. “PCNN Models and Applications” IEEE Transactions on Neural Networks. 10(3): 480- 498. doi: 1045-9227(99)03191-4.

[17] Wang, Z., and Yide, Ma. 2008. “Medical image fusion using m-PCNN” Information Fusion 9 (2): 176– 185. doi: 10.1016/j.inffus.2007.04.003.

[18] Wang, Z., and Yide. Ma. 2007 “Dual-channel PCNN and its application in the field of image fusion” Proc. of the 3rd International Conference on Natural Computation no. (1):755–759. doi: 10.1109/ICNC.2007.338.

[19] Miao, Q. and Wang, B. 2005. “A novel algorithm of multi-focus image fusion using adaptive PCNN”, "A novel adaptive multi-focus image fusion algorithm based on PCNN and sharpness" Proc. SPIE 5778, Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV, (704): doi:10.1117/12.603092.

[20] Xu, B., and Chen, Z. 2004. “A multisensor image fusion algorithm based on PCNN” Proc. of the 5th World Congress on Intelligent Control and Automation no. (4): 3679–3682. doi: 10.1109/WCICA.2004.1343284.

[21] Li, W. and Zhu, X. 2005. “A new image fusion algorithm based on wavelet packet analysis and PCNN”, Proc. of the 4th International Conference on Machine Learning and Cybernetics no. 9: pp. 5297–5301. doi: 10.1109/ICMLC.2005.1527879.

[22] El-taweel, Ghada Sami and Helmy, Ashraf Khaled. 2013. “Image fusion scheme based on modified dual pulse coupled neural network”, IET Image Processing 7(5): 407–414, doi: 10.1049/iet-ipr.2013.0045.

[23] Zhaobin Wang, Yide Ma , Feiyan Cheng, Lizhen. 2010. “Yang Review of pulse-coupled neural networks” Image and Vision Computing no. 28: 5–13, doi:10.1016/j.imavis.2009.06.007.

[24] Wang, Zhao-hui, Wang Jia-qi, Zhao De-gong, and Wei FU. 2012. “Image fusion based on shearlet and improved PCNN” Laser Infrared. 42(2): 213–216. http://caod.oriprobe.com/order.htm?id=29200739&ftext=base.

[25] Xuan Liu, Yue Zhou, and Jiajun Wang. 2014. “Image fusion based on shearlet transform and regional features” Int Journal Electronics and Communications (AEÜ) 68(6): 471-477. doi:10.1016/j.aeue.2013.12.003.

[26] Jingbo Zhang, et. al., 2008 . “Multi-focus image fusion using quality assessment of spatial domain and genetic algorithm” Conference on Human System Interactions 8(2): 71– 75. doi: 10.1109/HSI.2008.4581411.

[27] Liu, Wei, YIN Ming, LUAN Jing, and Guo Yu. 2013. “Adaptive image fusion algorithm based on Shift-invariant shearlet transform” Acta Photonica Sinica 42(4): 496-503. doi: 10.3788/gzxb20134204.0496.

[28] Lei, Wang, and Bin Li. 2012. "Multi-modal medical image fusion using the inter-scale and intra-scale dependencies between image shift-invariant shearlet coefficients" Information Fusion no. 19: 20-28. doi: /10.1016/j.inffus.2012.03.002.

[29] Wang, Z., Bovik, A.C., Sheikh, H.R., and Simoncelli, E.P. 2004. "Image quality assessment: from error visibility to structural similarity", IEEE Transaction on Image Processing 13 (4): 600–612. doi: 10.1109/TIP.2003.819861.

[30] Petrovic, V., and Xydeas, C. 2000. "On the effects of sensor noise in pixel-level image fusion performance" Proc. of the Third International Conference on Image Fusion no. 2: 14–19. doi: 10.1109/IFIC.2000.859842.

[31] Lee, J.S., Grunes, M. R. and Grandi, G. De. 1999. "Polarimetric SAR Speckle Filtering and its Implication for Classification", IEEE Transaction on Geoscience and Remote Sensing 37(5): 2363-2373. doi: 10.1109/36.789635.

[32] Jixian, Zhang. 2010. "Multi-source remote sensing data fusion: status and trends" International Journal of Image and Data Fusion 1(1): 5-24, doi: 10.1080/19479830903561035.

[33] Liu X, et al., .2014. "Image fusion based on shearlet transform and regional features". International Journal of Electronics Communications (AEÜ) 68(6): 471-477. doi:10.1016/j.aeue.2013.12.003.

[34] Lei Wang, and Tian Lian-fang. 2014. "Multi-modal medical image fusion using the inter-scale and intra-scale dependencies between image shift-invariant shearlet coefficients" Information Fusion no. (19): 20-28. doi:10.1016/j.inffus.2012.03.002.

Appendix 1. The spectral resolutions of the Egyptsat-1 data

Bands	Description	Wavelength (μm)	Resolution (m)
Band 1	Green	0.51-0.59	7.80
Band 2	Red	0.61-0.68	7.80
Band 3	Near infrared	0.80-0.89	7.80
Band 4	Panchromatic	0.50-0.89	7.80
Band 5	Mid infrared	1.10-1.70	39.5

Circular Traffic Signs Recognition Using The Number of Peaks Algorithm

Khaled M. Almustafa *kalmustafa@psu.edu.sa*
Prince Sultan University.
P.O.Box No. 66833
Riyadh 11586, K.S.A

Abstract

Smart cars nowadays include embedded computers to guide the driver in his trip. An important application that should be added to any car is the detection and recognition of traffic signs. In this paper, we focus on the recognition of a wide set of circular traffic signs using the Number of Peaks Algorithm [1]. After detecting a traffic sign, the algorithm draws three horizontal lines and three vertical lines across the image. The number of peaks (crossing from a black pixel to a white pixel) is calculated for each of the six lines as the image is scanned from right to left (for horizontal lines) or top to bottom (for vertical lines). The resulting numbers of peaks are used by the decision-tree-like search algorithm to distinguish between 51 circular road signs with a mean detection time of 8 milliseconds, 100% detection rate and in a fairly noisy environment.

Keywords: Traffic Signs Recognition, Pattern Recognition, Image Processing, Autonomous Cars.

1. INTRODUCTION

Automatic detection and recognition of traffic signs is an important addition to any smart car. Having such a tool in a car would alert the driver to possible obstacles or changes in the road and therefore reduce the possibility of accidents. Previous systems for real time detection of traffic signs are limited to a small set of circular or triangular signs [2, 3]. A system capable of recognizing all traffic signs within an acceptable amount of time for a moving car is highly desirable.

In this paper, a system is described that recognizes a wide set of circular traffic signs. The system uses a novel algorithm, the number of peaks algorithm, to differentiate between 51 circular road signs in a fast and reliable manner. Many researchers studied the detection and recognition of traffic signs since 1996 [1-21]. Most of the approaches to traffic sign recognition follow a two-step algorithm. The first step is the detection of a traffic sign in the image and the second step is to recognize the sign. The second step may involve classification of the sign first into predefined classes, such as triangular signs or speed signs, followed by recognition of a specific sign [4, 5, 6]. In the recognition phase, various approaches are available in the literature [7]. Most approaches are pixel-based using cross-correlation template matching [4, 8] or neural networks [5]. However, other approaches exist that are feature-based. For example, in [9] statistical properties, such as moments found from the binary images of the central part of sign candidates, were used. Local edge orientations and density at arbitrary fixation points were used [10].

A different approach was also presented in [6, 7] based on the similarities of the detected sign and the sign images stored in the template. Equiangular polygons are detected in a filtered image then a discrete-color image of this object is compared with model images. Another technique [11] uses error correcting output codes (ECOC) to build a system for multi-class classification of traffic signs. In all cases classification methods consider a limited number of signs such as six circular

signs [2] or blue traffic signs [3] and are characterized by varying recognition rate 81% [3], 76% to 91% [7], and 98.66% [11].

Section 2 introduces the number of peaks algorithm. Section 3 introduces the decision trees used in the algorithm and details how a specific sign is distinguished based on number of peaks in horizontal and vertical lines. Section 4 discusses the performance of the algorithm. Section 5 concludes the paper and gives suggestions for future research.

2. THE NUMBER OF PEAKS ALGORITHM

The number of peaks algorithm works after a traffic sign is detected. Figure 1 shows all possible images of circular signs recognized by the Number of Peaks Algorithm. Figure 1 shows each sign along with a number assigned to it. This number is used in the flowcharts of the algorithm to distinguish between the signs as shown in the next section.

FIGURE 1: Circular Traffic signs recognized by the Number of Peaks Algorithm and their number representation.

Once a traffic sign is detected, its negative image representation is generated as shown in Figure 2.

FIGURE 2: A sign and its negative image representation.

Then a factor f is selected from a discrete set {1/3, 1/4, 1/5, 1/9, 1/10}. This factor is used to select the positions of three horizontal lines and three vertical lines to be drawn across the image. Lines T, H and B (in Figure 3) are horizontal lines. Line T is at a distance f of the total image length taken from the image top. Line H is drawn at the middle of the image horizontally. Line B is

drawn at a distance f of the total image length taken from the image bottom. Similarly, lines R, V and L are three vertical lines drawn across the image at distances f of the total width from the image right boundary, at the middle and f of the total width from the image left boundary respectively.

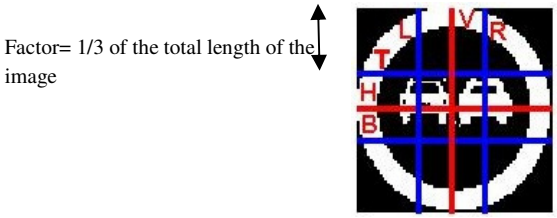


FIGURE 3: Crossing lines of an Image.

F=1/3	H	T	B	V	L	R	
26	2	3	2	2	1	1	
50	1	1	1	1	1	1	
21	2	1	1	2	2	2	
24	2	1	2	2	2	2	
25	2	1	2	2	2	4	
28	2	2	5	2	2	2	
30	2	2	2	2	2	2	
31	2	3	2	2	2	2	
23	3	2	3	2	2	3	
32	3	2	3	3	2	2	R 1/5 (1)
33	3	2	3	3	2	2	R 1/5 (2)
27	3	3	3	2	2	2	
51	3	3	2	3	2	2	
29	4	3	2	2	2	4	
37	4	3	5	4	2	2	
18	6	4	4	3	2	4	
20	4	4	2	2	3	3	
10	4	5	3	3	3	4	
16	5	3	3	3	3	3	
19	4	4	3	2	3	3	
4	4	4	4	3	3	4	
5	4	4	4	4	3	4	
39	3	4	4	5	3	3	
45	4	4	4	3	3	3	
35	10	4	5	5	3	6	
13	6	2	2	4	4	2	
6	5	4	7	4	4	3	
8	3	4	5	4	4	3	
34	3	3	4	3	4	3	
9	4	3	3	2	4	4	
42	4	3	3	4	4	4	
7	5	4	5	3	4	4	
2	5	3	4	4	4	4	R 1/5 (2)
38	5	3	4	4	4	4	R 1/5 (3)
17	6	4	3	5	4	4	
36	6	3	5	2	4	4	
41	4	3	2	3	4	5	
40	6	2	2	2	5	5	
1	7	4	5	4	5	6	
3	3	4	5	4	5	4	
43	4	3	5	4	5	4	
14	5	3	6	5	5	4	
15	2	4	6	4	5	4	
46	7	5	7	7	5	5	
47	8	4	7	6	5	7	
11	4	4	3	6	6	4	
12	8	4	4	5	6	4	
22	4	4	5	4	6	3	
44	3	9	5	6	6	5	
48	8	5	7	7	6	6	
49	9	7	5	7	7	6	

TABLE 1: Grouping of the Signs by the number of Peaks of the L line.

A Peak is defined as a crossing from a black pixel to a white pixel [1, 22] as the image is scanned from right to left (for lines T, H and B) or top to bottom (for lines R, V and L). A line starting with white pixels is considered to have one peak at the beginning. Line T in Figure 3 contains two sections where there is a transition from black pixels to white pixels. This means that for the line labeled H, the algorithm would return a value of 2.

3. DECISION TREES FOR THE NUMBER OF PEAKS ALGORITHM

The number of peaks for all 6 lines using factors 1/3 and 1/5 were calculated for all the circular traffic signs considered (51 in total). Those numbers were studied to find out the minimum number of lines that can be used to recognize each image. For all the images, we started out with the line that would divide those images into the maximum number of groups. Then use another line to keep dividing the groups until recognizing all the images. The results are summarized in Table 1. Table 1 is sorted according to the lines that were used to identify the signs. Given a circular traffic sign to be recognized, the algorithm starts with a factor of 1/3. The number of peaks for the L line is found. This number divides all the circular traffic signs into seven groups as shown in Table 1. If this number is 1, then the studied sign is either number 26 or number 50. The number of peaks for the V line is then found. If this value is 2 then the sign is number 26. If this value is 1, then the sign is number 50.

These results can also be seen in the flowchart shown in Figure 4. If the number of peaks for the L line is 3, then the candidate sign is a sign belonging to the set {20, 10, 16, 19, 4, 5, 39, 45, 35}. The number of peaks for additional lines should be found (B then T then R then V) to identify those signs. The remaining parts of the flowchart for the nodes labeled H, R1 and B2 are continued in Figures 5, 6 and 7 because of space limitation.

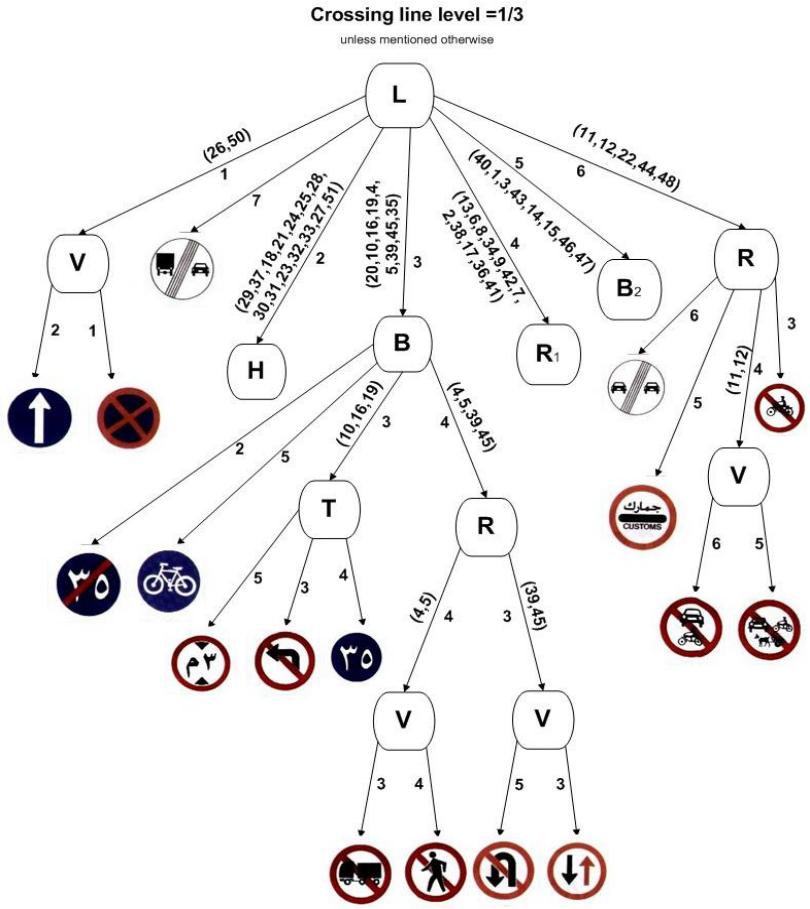


FIGURE 4: Flowchart for the First test of the algorithm.

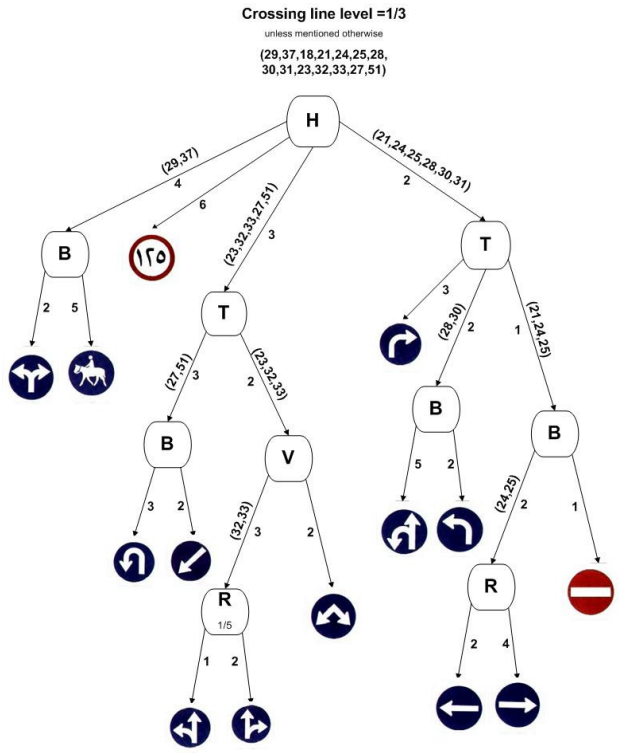


FIGURE 5: Flowchart for node H of Figure 4.

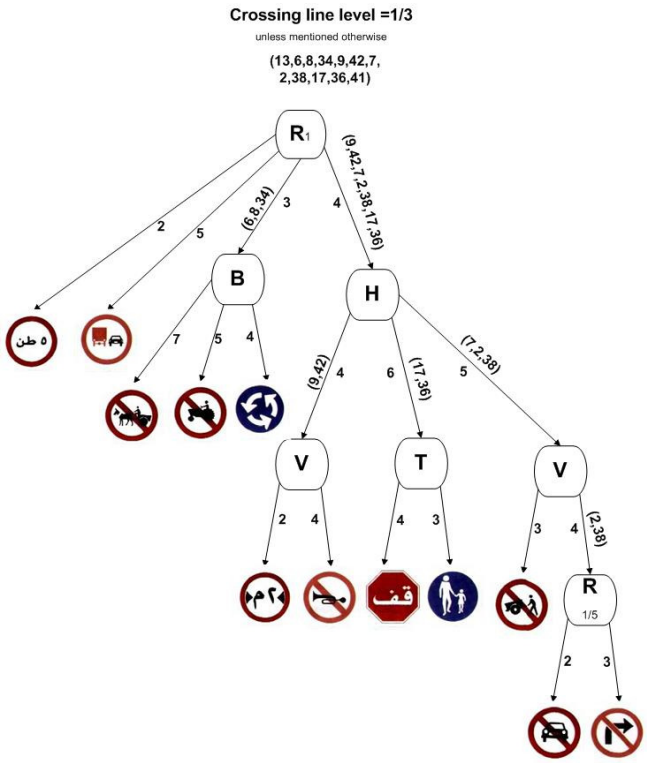


FIGURE 6: Flowchart for node R1 of Figure 4.

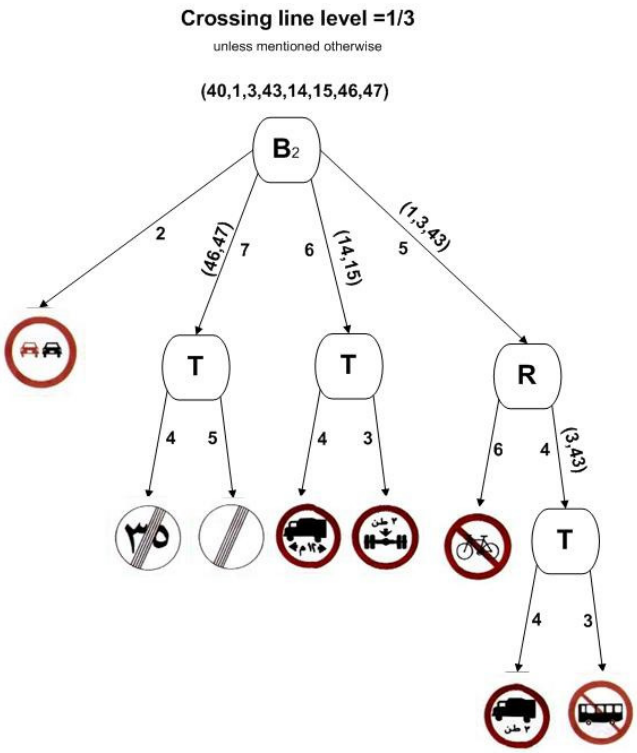


FIGURE 7: Flowchart for node B2 of Figure 4.

4. RESULTS

The algorithm was repeated 1000 times to recognize each of the 51 signs on an HP 8200 Elite with an I5 2500 CPU at 3.30 GHz. The average detection time for a sign is 8 milliseconds.

In order to test the performance of the Number of Peaks Algorithm in blurred images, noise was artificially introduced to the signs considered and the algorithm was run on the resulting noisy images. Artificially introduced errors were simulated using equation (1)

$$P = Q + \mu N \qquad (1)$$

Where matrix Q is representing the image, and N is of a same size as Q, which consists of uniformly distributed pseudo-random numbers and μ is a constant, which is used to control the amount of noise to introduce in the image. Performance is measured using the Frobenius norm of (P - Q). The Frobenius norm of an m × n matrix A is defined by the following equation (2).

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} \qquad (2)$$

A graph of the mentioned norm as a function of μ is shown in Figure 8. We can see that the solid line shows the averaged calculated threshold over the range of values of μ for different type of circular signs where the noisy image can be recognized by the algorithm, and the dashed line shows the averaged threshold for the value of μ where the character would not be recognized by the Human eye. The human eye threshold was determined based on our responses to the given noisy images. It turns out that the value of the threshold of recognition is equal to μ = 1.08, with an equivalent human eye threshold of 1.7. This means that our system would be able to work within a fairly noisy environment even when the amount of noise changes the features important to the algorithm drastically. However the system is still inferior to a human being in the same environment.

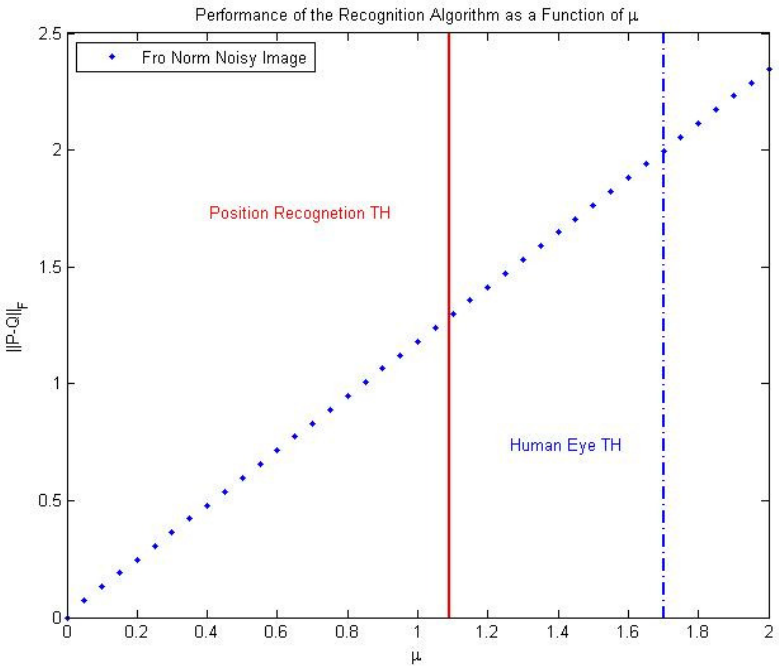


FIGURE 8: Performance of the Frobenius norm.

Even in artificially induced noisy images, the detection rate of the Number of Peaks algorithm is 100% up to some level of noise added.

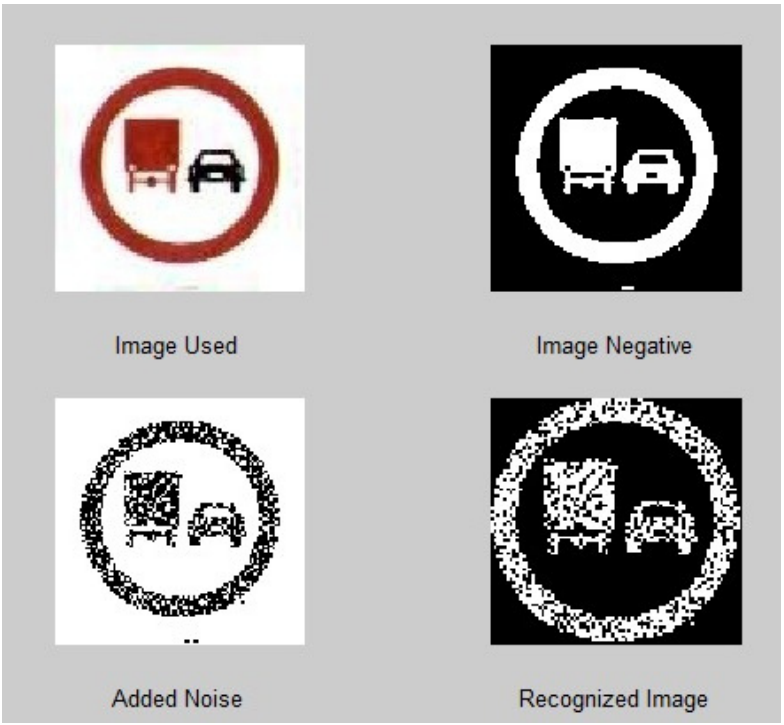


FIGURE 9: A sign with added noise having μ=1.08.

Figure 9 shows in the top row an image used in this simulation and its negative generated image. In the second row, the noisy image having a μ=1.08 is shown and the corresponding negative image that was successfully recognized by the number of peaks algorithm.

5. CONCLUSION AND FUTURE RESEARCH

In this paper an algorithm for recognition of 51 circular road signs was presented. The algorithm uses a simple method that requires a minimum number of image processing combined with a decision-tree-like search algorithm. The algorithm was repeated 1000 times to recognize each of the 52 signs on an HP 8200 Elite with an I5 2500 CPU at 3.30 GHz. The average detection time for a sign ranged between 8 milliseconds for sign number 11 to 9.5 milliseconds for sign number 23, with an average detection time of 8.47 milliseconds. Even in artificially induced noisy images, the detection rate of the Number of Peaks algorithm is 100% up to some value of the noise parameter μ.

The next step of this research is to extend this sign recognition algorithm to recognize signs of any shape using the same principle. Flowcharts similar to the ones presented in Figures 4 to 7 should be drawn to detect all traffic signs. The system should then be integrated into a full real time system to detect and then recognize all classes of traffic signs in any country. Shape recognition can be investigated to help detect and locate the sign in real time images. The algorithm should be implemented on portable hardware and installed in smart cars. Since the time to detect and recognize a sign should be minimized for the system to be useful in high speed zones, a parallel version of this algorithm should be created and tested on multicore processors and compared to the performance of implementing this algorithm on Field Programmable Gate Arrays (FPGA).

6. ACKNOWLEDGMENT

The authors would like to extend their sincere thanks to Prince Sultan University (PSU), Riyadh, K.S.A. for its support to the project.

7. REFERENCES

[1] L. Hamandi, K. AlMustafa, R. Zantout, H. Obeid, Recognition of Triangular Traffic Signs Using the Number of Peaks Algorithm, accepted for publication in The 2nd International Conference On Advances In Computational Tools For Engineering Applications, Dec. 12-15, 2012, Beirut, Lebanon.

[2] K. Baba, Y. Hirai, Real-time recognition of traffic signs using opponent color, Proceedings of the 14th World Congress on Intelligent Transport Systems (ITS) 2007, pp. 5127-5138.

[3] M. H. Alsibai, Y. Hirai , Real-Time Recognition of Blue Traffic Signs Designating Directions, International Journal of Intelligent Transportation Systems Research, May 2010, Volume 8, Issue 2, pp 96-105.

[4] G. Piccioli, E. De Micheli, P. Parodi, M. Campani, A robust method for road sign detection and recognition, Image and Vision Computing 14 (3) (1996) pp. 209–223.

[5] A. de la Escalera, L.E. Moreno, M.A. Salichs, J.M. Armingol, Road traffic sign detection and classification, IEEE Transactions on Industrial Electronics 44 (6) (1997) pp. 848–859.

[6] P. Paclík, J. Novovicova, P. Pudil, P. Somol, Road signs classification using the Laplace kernel classifier, Pattern Recognition Letters 21 (13–14) (2000) pp. 1165–1173.

[7] A. Ruta, Yongmin Li,Xiaohui Liu./ Real-time traffic sign recognition from video by class-specific discriminative features, Pattern Recognition 43(2010) pp. 416—430.

[8] A. de la Escalera, J.M. Armingol, J.M. Pastor, F.J. Rodríguez, Visual sign information extraction and identification by deformable models for intelligent vehicles, IEEE Transactions on Intelligent Transportation Systems 5 (2) (2004) pp. 57–68.

[9] P. Paclík, J. Novovicová, R.P.W. Duin, Building road-sign classifiers using a trainable similarity measure, IEEE Transactions on Intelligent Transportation Systems 7 (3) (2006) pp. 309–321.

[10] X.W. Gao, L. Podladchikova, D. Shaposhnikov, K. Hong, N. Shevtsova, Recognition of traffic signs based on their colour and shape features extracted using human vision models, Journal of Visual Communication and Image Representation 17 (4) (2006) pp. 675–685.

[11] Escalera et. al. Traffic sign recognition system with β-correction Machine Vision and Applications (2010) 21, pp. 99–111.

[12] Y. Aoyagi, T. Asakura, A study on traffic sign recognition in scene image using genetic algorithms and neural networks, Proceedings of the 1996 IEEE IECON 22nd International Conference on Industrial Electronics, Control, and Instrumentation, vol. 3, 1996, pp. 1838–1843.

[13] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, T. Koehler, A system for traffic sign detection, tracking and recognition using color, shape, and motion information, in: Proceedings of the IEEE Intelligent Vehicles Symposium, 2005, pp. 255–260.

[14] P. Douville, Real-time classification of traffic signs, Real-Time Imaging 6 (3) (2000) pp. 185–193.

[15] C.-Y. Fang, S.-W. Chen, C.-S. Fuh, Roadsign detection and tracking, IEEE Transactions on Vehicular Technology 52 (5) (2003) pp. 1329–1341.

[16] H. Fleyeh, Color detection and segmentation for road and traffic signs, in: Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, Singapore, vol. 2, 2004, pp. 809–814.

[17] M.A. Garcia-Garrido, M.A. Sotelo, E. Martin-Gorostiza, Fast traffic sign detection and recognition under changing lighting conditions, in: Proceedings of the IEEE International Conference on Intelligent Transportation Systems, 2006, pp. 811–816.

[18] Hatzidimos, Automatic Traffic Sign Recognition in Digital Images, Proceedings of the International Conference on Theory and Applications of Mathematics and Informatics - ICTAMI 2004, Thessaloniki, Greece, pp. 174 – 184.

[19] J. Miura, T. Kanda, Y. Shirai, An active vision system for real-time traffic sign recognition, in: Proceedings of the IEEE Conference on Intelligent Transportation Systems, Darborn, MI, USA, 2000, pp. 52–57.

[20] A. Ruta, Y. Li, X. Liu, Towards real-time traffic sign recognition by class-specific discriminative features, in: Proceedings of the 18th British Machine Vision Conference, Coventry, United Kingdom, vol. 1, 2007, pp. 399–408.

[21] A. Ruta, Y. Li, X. Liu, Traffic sign recognition using discriminative local features, in: Proceedings of the 7th International Symposium on Intelligent Data Analysis, Ljubljana, Slovenia, 2007, pp. 355–366.

[22] K. AlMustafa, R. Zantout, H. Obeid, “Peak Position, Recognizing Characters in Saudi License Plates”, 2011 IEEE GCC Conference and Exhibition for Sustainable Ubiquitous Technology, Dubai, United Arab Emirates, February 19-22, 2011, pp. 186-189.

INSTRUCTIONS TO CONTRIBUTORS

The *International Journal of Image Processing (IJIP)* aims to be an effective forum for interchange of high quality theoretical and applied research in the Image Processing domain from basic research to application development. It emphasizes on efficient and effective image technologies, and provides a central forum for a deeper understanding in the discipline by encouraging the quantitative comparison and performance evaluation of the emerging components of image processing.

We welcome scientists, researchers, engineers and vendors from different disciplines to exchange ideas, identify problems, investigate relevant issues, share common interests, explore new approaches, and initiate possible collaborative research and system development.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJIP.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with Volume 9, 2015, IJIP will be appearing with more focused issues. Besides normal publications, IJIP intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

LIST OF TOPICS

The realm of International Journal of Image Processing (IJIP) extends, but not limited, to the following:

- Architecture of imaging and vision systems
- Character and handwritten text recognition
- Chemistry of photosensitive materials
- Coding and transmission
- Color imaging
- Data fusion from multiple sensor inputs
- Document image understanding
- Holography
- Image capturing, databases
- Image processing applications
- Image representation, sensing
- Implementation and architectures
- Materials for electro-photography
- New visual services over ATM/packet network
- Object modeling and knowledge acquisition
- Autonomous vehicles
- Chemical and spectral sensitization
- Coating technologies
- Cognitive aspects of image understanding
- Communication of visual data
- Display and printing
- Generation and display
- Image analysis and interpretation
- Image generation, manipulation, permanence
- Image processing: coding analysis and recognition
- Imaging systems and image scanning
- Latent image
- Network architecture for real-time video transport
- Non-impact printing technologies
- Photoconductors

- Photographic emulsions
- Prepress and printing technologies
- Remote image sensing
- Storage and transmission
- Photopolymers
- Protocols for packet video
- Retrieval and multimedia
- Video coding algorithms and technologies for ATM/p

CALL FOR PAPERS

Volume: 9 - **Issue: 2**

i. Submission Deadline : February 28, 2015 **ii. Author Notification:** March 31, 2015

iii. Issue Publication: April 2015

CONTACT INFORMATION

Computer Science Journals Sdn BhD

B-5-8 Plaza Mont Kiara, Mont Kiara

50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6204 5627

Fax: 006 03 6204 5628

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2014
COMPUTER SCIENCE JOURNALS SDN BHD
B-5-8 PLAZA MONT KIARA
MONT KIARA
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6204 5627
FAX: 006 03 6204 5628
EMAIL: cscpress@cscjournals.org