Editor-in-Chief
Professor Hu, Yu-Chen

INTERNATIONAL JOURNAL OF
IMAGE PROCESSING (IJIP)
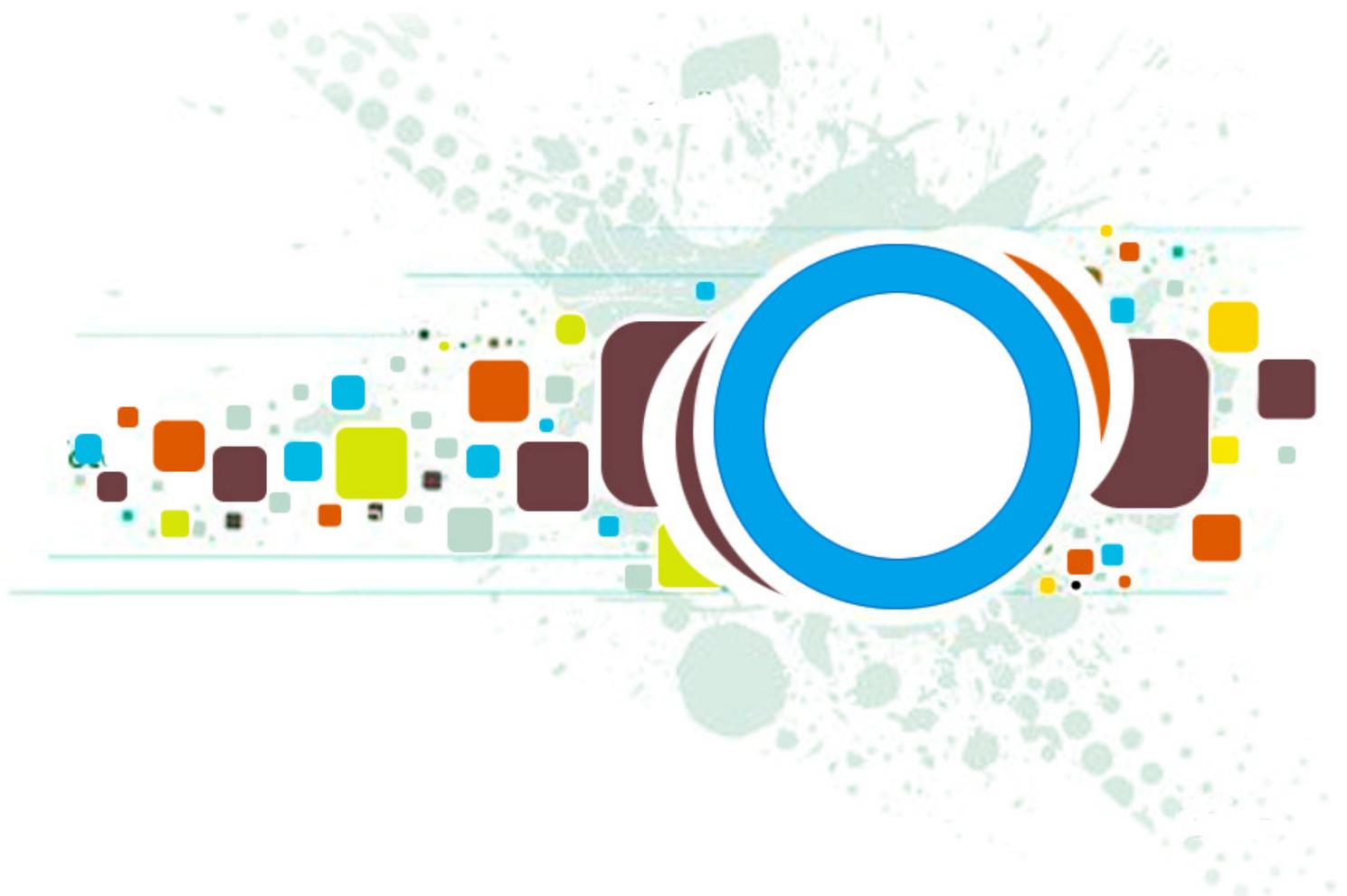
INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

VOLUME 6, ISSUE 2, 2012

EDITED BY
DR. NABEEL TAHIR

INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

CSC Publishers, 2012

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 second issue of volume six 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. Starting with volume 6, 2012, IJIP appears in 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)

Dr. Khalid Mohamed Hosny
Zagazig University
Egypt

Dr. Chin-Feng Lee
Chaoyang University of Technology
Taiwan

Professor Santhosh.P.Mathew
Mahatma Gandhi University
India

Dr Hong (Vicky) Zhao
Univ. of Alberta
Canada

Professor Yongping Zhang
Ningbo University of Technology
China

Assistant Professor Humaira Nisar
University Tunku Abdul Rahman
Malaysia

Dr M.Munir Ahamed Rabbani
Qassim University
India

Dr Yanhui Guo
University of Michigan
United States of America

Associate Professor András Hajdu
University of Debrecen
Hungary

Assistant Professor Ahmed Ayoub
Shaqra University
Egypt

Dr Irwan Prasetya Gunawan
Bakrie University
Indonesia

Assistant Professor Concetto Spampinato
University of Catania
Italy

Associate Professor João M.F. Rodrigues
University of the Algarve
Portugal

TABLE OF CONTENTS

Volume 6, Issue 2, April 2012

Pages

Image Compression and Reconstruction Using a New Approach by Artificial Neural Network

K.Siva Nagi Reddy   *sivanagireddykalli@gmail.com*
Associate Professor, Dept of ECE,
Montessori Siva Sivani Institute of Science and Technology,
College of Engineering, Mylavaram,
Vijayawada, Andhra Pradesh, INDIA.

Dr.B.R.Vikram   *vikramom2007@gmail.c*
Professor, Dept of ECE,
Vijay Rural Engineering College,
Nizamabad, Nizamabad (D.t),
Andhra- Pradesh, INDIA.

L. Koteswara Rao   *kots.lkr@gmail.com*
Asst. Professor, Dept of ECE,
Faculty of Science & Technology,
IFHE (University), Hyderabad, India

B.Sudheer Reddy   *sudheerbommareddy@gmail.com*
Associate Professor, Dept of CSE,
Montessori Siva Sivani Institute of Science and Technology,
College of Engineering, Mylavaram,
Vijayawada, Andhra Pradesh, INDIA.

Abstract

In this paper a neural network based image compression method is presented. Neural networks offer the potential for providing a novel solution to the problem of data compression by its ability to generate an internal data representation. This network, which is an application of back propagation network, accepts a large amount of image data, compresses it for storage or transmission, and subsequently restores it when desired. A new approach for reducing training time by reconstructing representative vectors has also been proposed. Performance of the network has been evaluated using some standard real world images. It is shown that the development architecture and training algorithm provide high compression ratio and low distortion while maintaining the ability to generalize and is very robust as well.

Key words: Artificial Neural Network, Image Processing (ANN), Multilayer Perception (MLP) and Radial Basis Functions (RBF),Normalization, Levenberg-Marquardt, Jacobian

---

1. INTRODUCTION

Artificial Neural networks are simplified models of the biological neuron system and therefore have drawn their motivation from the computing performed by a human brain. A neural network, in general, is a highly interconnected network of a large number of processing elements called neurons in an architecture inspired by the brain. Artificial neural networks are massively parallel adaptive networks of simple nonlinear computing elements called neurons which are intended to abstract and model some of the functionality of the human nervous system in an attempt to partially capture some of its computational strengths. A neural network can be viewed as comprising eight components which are neurons, activation state vector, signal function, pattern of connectivity, activity aggregation rule, activation rule, learning rule and environment.

K.Siva Nagi Reddy, Dr.B.R.Vikram, L.Koteswara Rao, B.Sudheer Reddy

Recently, artificial neural networks [1] are increasingly being examined and considered as possible solutions to problems and for application in many fields where high computation rates are required [2]. Many People have proposed several kinds of image compression methods [3]. Using Artificial neural network (ANN) technique with various ways [4, 5, 6, 7]. A detail survey of about how ANN can be applied for compression purpose is reported in [8,9,10,11].Broadly, two different categories for improving the compression methods and performance have been suggested. Firstly, develop the existence method of compression by use of ANN technology so that improvement in the design of existing method can be achieved. Secondly, apply neural network to develop the compression scheme itself, so that new methods can be developed and further research and possibilities can be explored for future. The typical image compression methods are based on BPNN techniques. The Back propagation Neural Network (BPNN) is the most widely used multi layer feed forward ANN. The BPNN consists of three or more fully interconnected layers of neurons. The BP training can be applied to any multilayer NN that uses differentiable activation function and supervised training [12].

The BPNN has the simplest architecture of ANN that has been developed for image compression but its drawback is very slow convergence. In [13] suggested mapping the gray levels of the image pixels and their neighbors in such a way that the difference in gray levels of the neighbors with the pixel is minimized and then the CR and network convergence can be improved. They achieved this by estimating a Cumulative Distribution Function (CDF) for the image. They used CDF to map the image pixels, then, the BPNN yields high CR and converges quickly. In [14] used BPNN for image compression and developed algorithm based on improved BP. The blocks of original image are classified into three classes: background blocks, object blocks and edge blocks, considering the features of intensity change and visual discrimination Finally, In [15] presented an adaptive method based on BPNN for image compression/decompression based on complexity level of the image by dividing image into blocks, computing the complexity of each block and then selecting one network for each block according to its complexity value. They used three complexity measure methods such as: entropy, activity and pattern-based to determine the level of complexity in image blocks.

This paper is organized as follows. In section II we discuss Methodology (Image compression using ANN) III Describes the Neural network models. IV Describes the multi-layer perception neural network and its approach that is directly developed for image compression. In section V describe the Process steps for compression. VI explains the experimental results of our implementation are discussed and finally in section VII we conclude this research and give a summary on it.

2. METHODOLOGY (IMAGE COMPRESSION USING ANN)

2.1 Introduction to Image Compression
Image Processing is a very interesting and a hot area where day-to-day improvement is quite inexplicable and has become an integral part of own lives. Image processing is the analysis, manipulation, storage, and display of graphical images. An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations. Image processing is a module that is primarily used to enhance the quality and appearance of black and white images. It also enhances the quality of the scanned or faxed document, by performing operations that remove imperfections. Image processing operations can be roughly divided into three major categories, Image Enhancement, Image Restoration and Image Compression. Image compression is familiar to most people. It involves reducing the amount of memory needed to store a digital image.

K.Siva Nagi Reddy, Dr.B.R.Vikram, L.Koteswara Rao, B.Sudheer Reddy

Digital image presentation requires a large amount of data and its transmission over communication channels is time consuming. To rectify that problem , large number of techniques to compress the amount of data for representing a digital image have been developed to make its storage and transmission economical. One of the major difficulties encountered in image processing is the huge amount of data used to store an image. Thus, there is a pressing need to limit the resulting data volume. Image compression techniques aim to remove the redundancy present in data in a way, which makes image reconstruction possible. Image compression continues to be an important subject in many areas such as communication, data storage, computation etc.

In order to achieve useful compression various algorithms were developed in past. A compression algorithm has a corresponding decompression algorithm that, given the compressed file, reproduces the original file. There have been many types of compression algorithms developed. These algorithms fall into two broad types, 1)   Loss less algorithms, and 2)   Lossy algorithms.

A lossless algorithm reproduces the original exactly. Whereas, a lossy algorithm, as its name implies, loses some data. Data loss may be unacceptable in many applications. For example, text compression must be lossless because a very small difference can result in statements with totally different meanings. There are also many situations where loss may be either Unnoticeable or acceptable. But various applications require accurate retrieval of image, wherein one such application is medical processing.
So Image Compression enhances the progress of the world in communication.

2.2 NETWORK Selection for Compression (Back propagation)
The first step to solve the problem is to find the size of the network that will perform the desired data compression. We would like to select a network architecture that provides a reasonable data reduction factor while still enabling us to recover a close approximation of the original image from the encoded form. This network used is a feed forward network consists of   three layers, one Input Layer (IL) with N neurons, one Output Layer (OL) with N neurons and one (or more) Hidden Layer (HL) with Y neurons. All connections are from units in one layer to the other. The hidden layer consists of fewer units than the input layer, thus compress the image. The size of the output and the input layer is same and is used to recover the compressed image. The network is trained using a training set of patterns with desired outputs being same as the inputs using back propagation of error measures. Using the back-propagation process the network will develop the internal weight coding so that the image is compressed for ratio of number of input layer nodes to the number of hidden layer nodes equal to four. If we then read out the values produced by the hidden layer units in our network and transmit those values to our receiving station, we can reconstruct the original image by propagating the compressed image to the output units in identical networks.

The number of connections between each two layers in NN is calculated by multiplying the total number of neurons of the two layers, then adding the number of bias neurons connections of the second layer (bias connections of a layer is equal to the number of layer neurons). If there are N_i neurons in the input layer, N_h neurons in the hidden layer and No neurons in the output layer, the total number of connections is given by equation:

$$\text{Network Size:}(N_w)= [(N_i*N_h)+N_h]+[(N_h*N_o)+N_o]$$
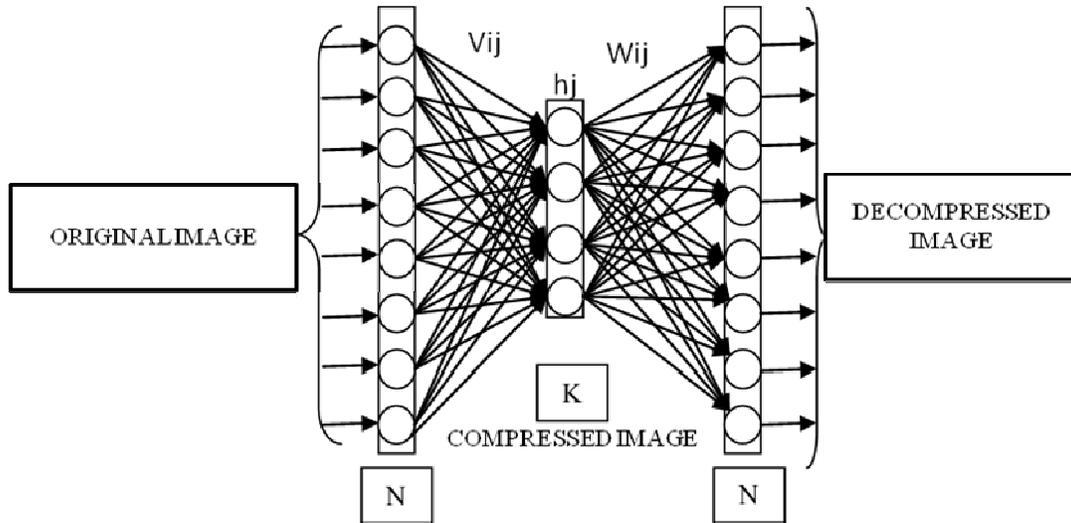
FIGURE 1: Neural Networks Architecture

2.3 Image Pre-Processing

2.3.1 RGB to Y Cb Cr and back to RGB.
The Y, Cb, and Cr components of one color image are defined in YUV color coordinate, where Y is commonly called the luminance and Cb, Cr are commonly called the chrominance. The meaning of luminance and chrominance is described as follows

◆   **Luminance:** received brightness of the light, which is proportional to the total energy in the visible band.
◆   **Chrominance:** describe the perceived color tone of a light, which depends on the wavelength composition of light chrominance is in turn characterized by two attributes – hue and saturation.
1.   **hue:** Specify the color tone, which depends on the peak wavelength of the light
2.   **saturation:** Describe how pure the color is, which depends on the spread or bandwidth of the light spectrum
3.

The RGB primary commonly used for color display mixes the luminance and chrominance attributes of a light. In many applications, it is desirable to describe a color in terms of its luminance and chrominance content separately, to enable more efficient processing and transmission of color signals. Towards this goal, various three-component color coordinates have been developed, in which one component reflects the luminance and the other two collectively characterize hue and saturation. One such coordinate is the YUV color space. The $[\text{Y Cb Cr}]^{\text{T}}$ values in the YUV coordinate are related to the $[\text{R G B}]^{\text{T}}$ values in the RGB coordinate by

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.164 & 0.000 & 1.596 \\ 1.164 & -0.392 & -0.813 \\ 1.164 & 2.017 & 0.000 \end{bmatrix} \begin{bmatrix} Y-16 \\ Cb-128 \\ Cr-128 \end{bmatrix}$$

FIGURE 2: Color conversion matrices for RGB and Y Cb Cr

Similarly, if we would like to transform the YUV coordinate back to RGB coordinate, the inverse matrix can be calculated from (1.1), and the inverse transform is taken to obtain the corresponding RGB components.

2.3.2 Spatial Sampling of Color Component
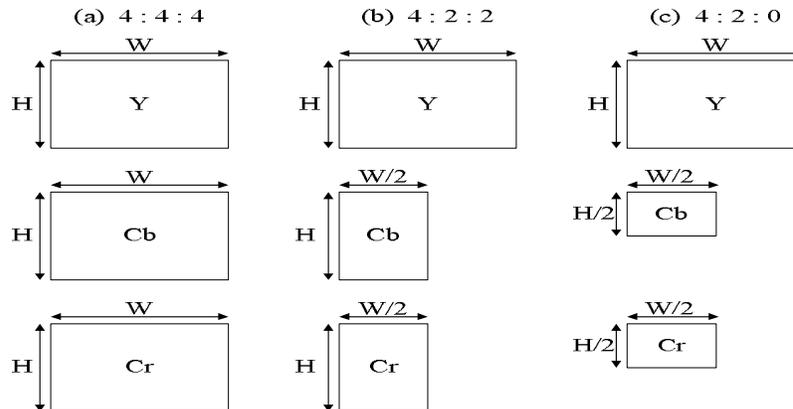
FIGURE 3: Three different chrominance down sampling format

Because the eyes of human are more sensitive to the luminance than the chrominance, the sampling rate of chrominance components is half that of the luminance component. This will result in good performance in image compression with almost no loss of characteristics in visual perception of the new up sampled image. There are three color formats in the baseline system:

◆ **4:4:4 formats**: The sampling rate of the luminance component is the same as those of the chrominance.

◆ **4:2:2 formats**: There are 2 Cb samples and 2 Cr samples for every 4 Y samples. This leads to half number of pixels in each line, but the same number of lines per frame.

◆ **4:2:0 formats**: Sample the Cb and Cr components by half in both the horizontal and vertical directions. In this format, there are also 1 Cb sample and 1 Cr sample for every 4 Y samples.

At the decoder, the down sampled chrominance components of 4:2:2 and 4:2:0 formats should be up sampled back to 4:4:4 formats.

2.3.3 Image Normalization and Segmentation

The image data is represented by the pixel value function f(X,Y) where X and Y correspond to the spatial coordinates within the image with pixel values between 0 and 2k-1, where k is the number of bits which represent each pixel in the image, usually k = 8, then the pixel values would lie within the range [0-255]. The ANN requires inputs with real type and the sigmoid function of each ANN neuron requires the input data to be in the range [0-1]. For this reason the image data values must be normalized. The normalization is the process of linearly transformation of image values from the range [0-255] into another range that is appropriate for ANN requirements to obtain valuable results and to speed up the learning .In this paper, the image is linearly transformed from range [0-255] to range [0-1]. Image segmentation is the process of dividing the image into sub images, each of which is considered to be a new separate image. In this paper, the image is segmented by dividing it into non overlapping blocks with equal size to simplify the learning/compressing processes. The input layer units of ANN are represented by one-dimensional vector. Image rasterization is the process of converting each sub image from a two-dimensional block in to a one dimensional vector.

2.4 Initialization of ANN Learning Parameters

The weight connections of the ANN are represented by two weight matrices. The first matrix is V which represents the weight connections between the input and hidden layer units and the second matrix is W which represents the weight connections between the hidden and output layer

units. These two weight matrices must be initialized to small random numbers because the network may be saturated by large values of the weights. The learning rate value usually reflects the rate of network learning and its value (between 0.1 and 0.9) is chosen by the user of the network. Values that are very large can lead to instability in the network and unsatisfactory learning, while values that are too small can lead to excessively slow learning.

2.5 Training the ANN

The input image is split up into blocks or vectors of 4×4, 8 ×8 or 16×16 pixels. These vectors are used as inputs to the network. The network is provide by the expected (or the desired) output, and it is trained so that the coupling weights, $\{w_{ij}\}$, scale the input vector of N -dimension into a narrow channel of Y -dimension (Y < N) at the hidden layer and produce the optimum output value which makes the quadratic error between output and the desired one minimum. In fact this part represents the learning phase, where the network will learn how to perform the task. In this process of leering a training algorithm is used to update network weights by comparing the result that was obtained and the results that was expected. It then uses this information to systematically modify the weight throughout the network till it finds the optimum weights matrix.
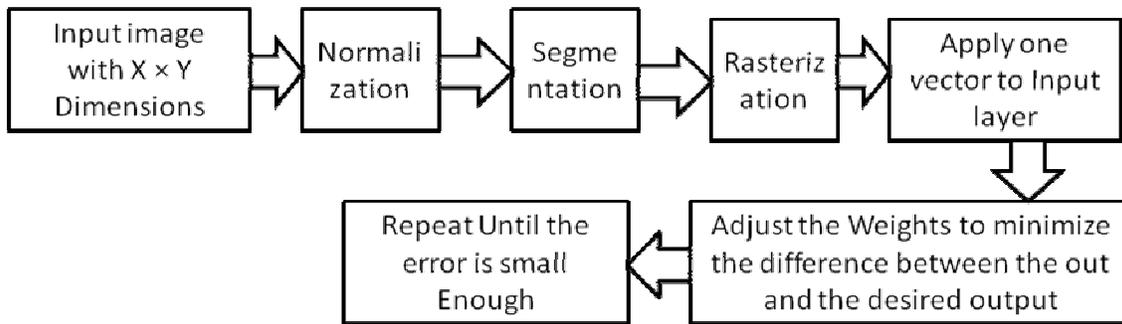
FIGURE 4: Block diagram of ANN training

2.6 Encoding

The trained network is now ready to be used for image compression which, is achieved by dividing or input images into normalization and segmentation. The segmented image blocks to rasterization and that vector is given to input layer then the output of hidden layer renormalized to represent the compressed image.
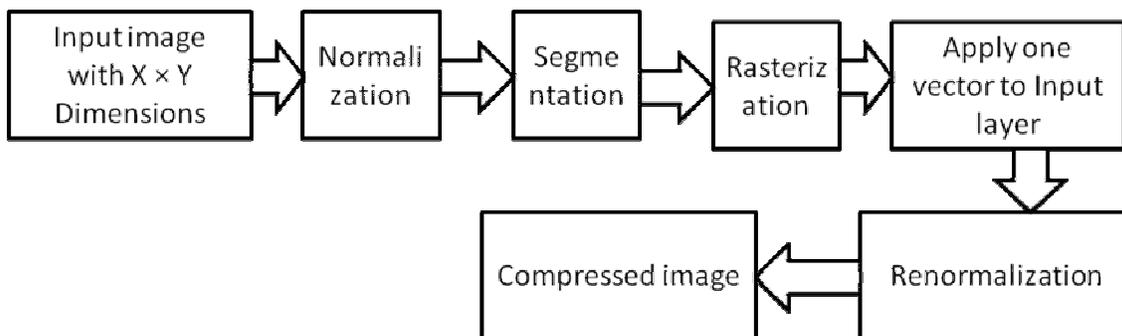
FIGURE 5: Block diagram of the Encoding Steps

2.7 Decoding

To decompress the image; first the compressed image is renormalized then applies it to the output of the hidden layer and get the one vector of the hidden layer output is normalized then it rasterization to represent the reconstruct the image. Fig. 6 show the decoder block diagram.
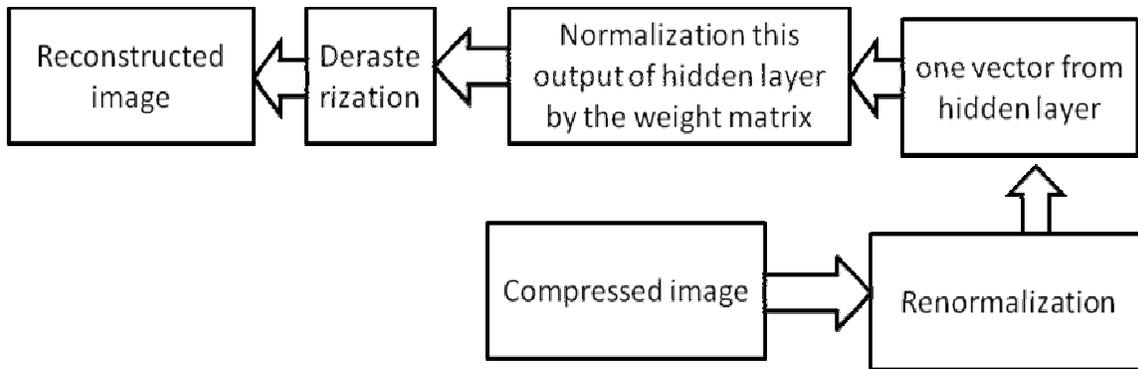
FIGURE 6: Block diagram of the Decoding Steps

3. NURALNETWORK MODELS

Artificial neural network (NN) techniques have been applied to solve complex problems in the fields of image processing and image compression. Multilayer perception (MLP) and radial basis functions (RBF) network are particularly efficient models for classification and image compression.

3.1 Multi layer Perception

Basic multilayer perception (MLP) building unit is a model of artificial neuron. This unit computes the weighted sum of the inputs plus the threshold weight and passes this sum through the activation function (usually sigmoid).

$$v_j = \theta_j + \sum W_{ji} x_i = \sum W_{ji} x_i \tag{1}$$

$$y_j = \Phi_j (v_j) \tag{2}$$

Where v_j is a linear combination of inputs
$x_1, x_2 \dots x_p$ of neuron j, $w_{jo} = \theta_j$ is the threshold weight connected to the special input $x_0 = -1$, y_j, is the output of neuron j and ϕ_j, (is its activation function. Here we use the special form of sigmoid (non-constant, bounded, and monotone increasing) activation function - logistic function.

$$y_j = 1/1 + \exp (v_j) \tag{3}$$

In a multilayer perception, the outputs of the units in one layer form the inputs to the next layer. The weights of the network are usually computed by training the network using the back propagation (BP) algorithm .A multilayer perception represents nested sigmoid scheme for single output neuron is

$$F(x, w) = \Phi_j (\sum W_{oj} \Phi_j \dots \Phi_j (\sum W_{jk} \Phi_j \dots)) \tag{4}$$

Where $\phi(.)$ is a sigmoid activation function, w_{oj} is the synaptic weight from neuron j in the last hidden layer to the Single output neuron o, and so on for the other synaptic weights, x_i is the ith element of the input vector **x**. The weight vector **w** denotes the entire set of synaptic weights ordered by layer.

3.2 Radial Basis Function Network

RBF network is based on a multivariable interpolation: Given as set of **N** distinct vectors {$x_i \in R^p$ | I =1 …N} and N real numbers {d_i R | i=1,…,N}, the aim is to find a function f:Rp→R satisfying the condition f(x_i)= d_i, i=1, ,N. RBF approach works with N radial basis functions (RBF) Φ_j where Φi: Rp→R, i=1, ,N and $\Phi_j = \Phi$ (|| **x- C_i** ||), where Φ:R+→R, $x_i \in R^p$, || || is a norm on R_p, **C_i** Rp

are centers of RBFs. Centers are set to $C_i = x_i$ Rp, i=1,…,N. Very often used form of RBF is the Gaussian function Φi (x)=exp($-x^2/2\sigma 2$), where σ is a width (parameter). Functions

$$f(x) = \sum W_{ji} \Phi (\| x - v_j \|) \tag{5}$$

i=1,…,N form the basis of a liner space and interpolation function f is their linear combination Interpolation problem is simple to solve, in contrast to approximation problem (there is N given points and n_0 functions Φ, where $n_0 < N$.), which is more complicated. Then it is a problem to set centers C_i i=1…n_0, also parameter σ of each RBF can be not the same for all RBFs. One possible solution for RBF approximation problem is a neural network solution. RBF network is a feed forward network consisting of input, one hidden and output layer. Input layer distributes input vectors into the network, hidden layer represents RBFs Φ. The linear output neurons compute linear combinations of their inputs. The RBF model is suitable for data interpolation and classification. For this work, the RBF model was trained from the selected input vectors producing the synaptic weight vectors.

4. MULTI-LAYER NEURAL NETWORKS FOR IMAGE COMPRESSION

Multi-Layer neural networks with back-propagation algorithm can directly be applied to image compression. The simplest neural network structure for this purpose is illustrated in Fig. 7. This network has three layers, input, hidden and output layer. Both the input and output layers are fully connected to the hidden layer and have the same number of neurons, *N.* Compression can be achieved by allowing the value of the number of neurons at the hidden layer, K, to be less than that of neurons at both input and output layers ($K \leq N$). As in most compression methods, the input image is divided apart into blocks, for example with 8×8, 4× 4 or 16 ×16 pixels. These block sizes determine the number of neurons in the input/output layers which convert to a column vector and fed to the input layer of network; one neuron per pixel. With this basic MLP neural network, compression is conducted in training and application phases as follow.
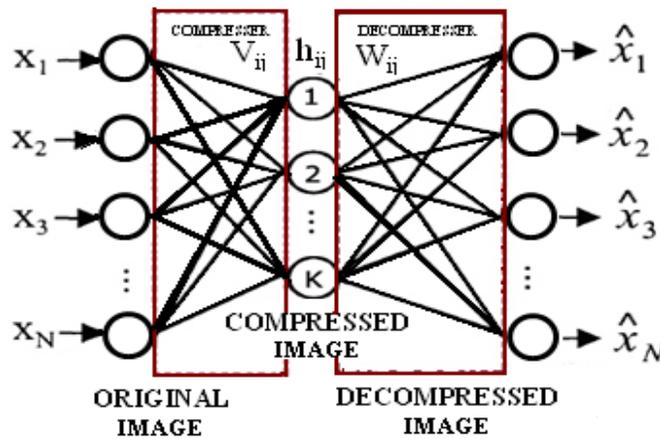
FIGURE 7: Neural Network Structure

4.1 The Neural Network Architecture

The Back propagation (BP) is one of neural networks, which are directly applied to image compression. The network structure is shown in Figure 8. This structure referred to feed forward auto associative type network. The input layer and output layer are fully connected to the hidden layer. Compression is achieved by estimating the value of *K,* the number of neurons at the hidden layer less than that of neurons at both input and the output layers. In the given Architecture "N" is number of neurons in input or output layer and the K is the number of neurons of hidden layer. The input image is split up into a number of blocks; each block has *N* pixels, which is equal to the number of input neurons.
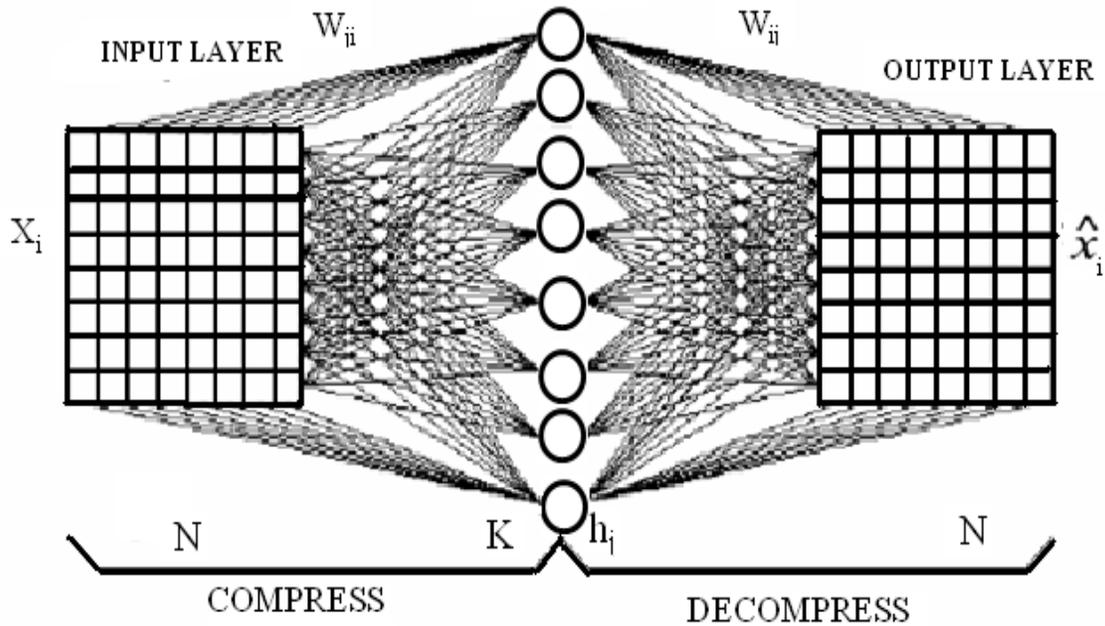
FIGURE 8: The N-K-N neural network. Actual connection

4.2 Training Process

Like all other training processes, in this phase a set of image samples are selected to train the network via the different back propagation learning rules. For compression purpose the target pattern in the output layer neurons of the network will be same as the input pattern. The compression is represented by the hidden layer which is equivalent to compress the input into a narrow channel. Training samples of blocks are converted into vectors and then normalized from their gray-level range into [0, 1]. In accordance with the structure of neural network shown in Figure, the operation for adjusting weights for compressing and de-compressing can be described as the following equations.

$$H_j^{in} = \sum_{i=0}^{N} V_0 X_i \, , h_i = f(H_j^{in}):1 \le j : \tag{6}$$

$$X_j^{in} = \sum_{j=1}^{K} W_0 h_i \, , X_i = g(X_j^{in}):1 \le j \le \tag{7}$$

In the above equations, *f* and *g* are the activation functions which can be linear or nonlinear. *V* and *W* represent the weights of compressor and de-compressor, respectively. The extracted *N* × *K* transform matrix in compressor and *K* × *N* in de-compressor of linear neural network.

The training process of the neural network is iterative and is stopped when the weights converge to their true values. In real applications the training is stopped when the error of equation (8) reaches to a maximum number of iterations limits the iterative process.
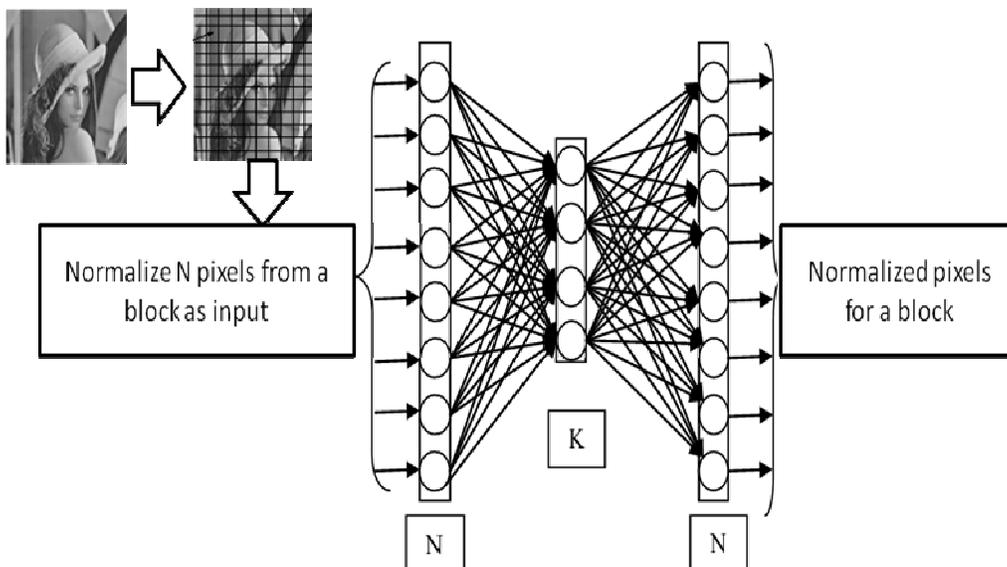
FIGURE 9: Architecture of Neural Network Structure at the time of Training

Preparation of NN Training/testing Set
A testing set consists of sub images that are not included in the training set and it is used to assess the NN performance after training. The preparation of training/testing set includes the following steps:
Step 1: Apply the segmentation process on the image to be used in learning/testing processes.
Step 2: Apply rasterization and normalization on every block segment.
Step 3: Store the results in the training set file.
Step 4: Repeat from Step 1 while there are more images to be used in training process.

4.3 Learning Algorithms

4.3.1 Levenberg-Marquardt Algorithm (LM)
For LM algorithm [16, 17], the performance index to be optimized is defined as

$$F(w) = \sum \left[\sum (d_{kp} - o_{kp})^2 \right] \tag{8}$$

Where $w = [w_1\ w_2\ ....\ w_N]^T$ consists of all weights of the network, d_{kp} is the desired value of the kth output and the pth pattern, o_{kp} is the actual value of kth output and the pth pattern is the number of the weights, P is the number of patterns, and K is the number of network outputs. Equation (8) can be written as

$$F(w) = E^T E \tag{9}$$

In above equation E is the Cumulative Error Vector (for all patterns)

$$E = [e_{11}...e_{k1}\quad e_{12}...e_{k2}...e_{kp}]^T \quad e_{kp} = d_{kp} - o_{kp},\ k=1...\ K,\ p = 1...P$$

From equation (9) the Jacobin matrix is defined as

$$
J(w) = \begin{bmatrix}
\dfrac{\partial e_{11}}{\partial w_1} & \dfrac{\partial e_{11}}{\partial w_2} & \cdots & \dfrac{\partial e_1}{\partial w} \\[6pt]
\dfrac{\partial e_{21}}{\partial w_1} & \dfrac{\partial e_{11}}{\partial w_2} & \cdots & \dfrac{\partial e_1}{\partial w} \\[6pt]
\vdots & & & \vdots \\[4pt]
\dfrac{\partial e_{k1}}{\partial w_1} & \dfrac{\partial e_{k1}}{\partial w_2} & \cdots & \dfrac{\partial e}{\partial w} \\[6pt]
\vdots & \vdots & & \\[4pt]
\dfrac{\partial e_{1p}}{\partial w_1} & \dfrac{\partial e_{1p}}{\partial w_2} & \cdots & \dfrac{\partial e_1}{\partial w_j} \\[6pt]
\dfrac{\partial e_{2p}}{\partial w_1} & \dfrac{\partial e_{2p}}{\partial w_2} & \cdots & \dfrac{\partial e_2}{\partial w_j} \\[6pt]
\vdots & \vdots & \cdots & \vdots \\[4pt]
\dfrac{\partial e_{kp}}{\partial w_1} & \dfrac{\partial e_{kp}}{\partial w_2} & \cdots & \dfrac{\partial e_{kj}}{\partial w_I}
\end{bmatrix}
\tag{10}
$$

And the weights are calculated using the following equation using Newton method

$$
w_{k+1} = w_k - (J_t^{T}(w_k)\, J_t(w_k) + \alpha_t I)^{-1} J_t^{T} (\; \tag{11}
$$

Where "I" is identity unit matrix, "α" is a learning factor and "J" is Jacobin of m output errors with respect to n weights of the neural network. For α=0 it becomes the Gauss-Newton method. If "α" is very large The LM algorithm becomes the steepest decent or the EBP algorithm. The "α" parameter is automatically adjusted for all iterations in order to secure convergence. The LM algorithm requires computing of the Jacobin J Matrix at each iteration step and the inversion of $J^{T}J$ square matrix. Note that in the LM algorithm an N by N matrix must be inverted for all iterations. This is the reason why for large size neural networks the LM algorithm is not practical. We are proposing another method that provides a similar performance, while lacks the inconveniences of LM, and is more stable.

4.3.2 Modification of the LM Algorithm
Instead of the performance index given by (8), the following new performing index is introduced

$$
F(w) = \sum_{k=1}^{k} \left[\sum_{p=1}^{p} (d_{kp} - o_{kp}) \right] \tag{12}
$$

This form of the index, which represents a global error, will later lead to a significant reduction of the size of a matrix to be inverted at each iteration step. Equation (12) can be also written as:

$$
F(w) = \hat{E} \tag{13}
$$

Where $\quad \hat{E} = [\hat{e_1} \quad \hat{e_2} \quad \ldots \hat{e}, \quad \hat{e_k} = \sum_{p=1}^{p}(d_{kp} - o_{kp}) \quad$ At k=1...K

Now the modified Jacobin matrix can be defined as

$$
\hat{J}(w) = \begin{bmatrix}
\dfrac{\partial \hat{e_1}}{\partial w_1} & \dfrac{\partial \hat{e_1}}{\partial w_2} & \cdots & \dfrac{\partial}{\partial} \\[6pt]
\dfrac{\partial \hat{e_2}}{\partial w_1} & \dfrac{\partial \hat{e_2}}{\partial w_2} & \cdots & \dfrac{\partial}{\partial} \\[6pt]
\vdots & & & \vdots \\[4pt]
\dfrac{\partial \hat{e_k}}{\partial w_1} & \dfrac{\partial \hat{e_k}}{\partial w_2} & \cdots & \dfrac{\partial \hat{e}}{\partial w}
\end{bmatrix}
\tag{14}
$$

And the equation (11) can be written using modified Jacobin matrix

$$
w_{k+1} = w_k - (\hat{J}_t^{T}(w_k)\, \hat{J}_t(w_k) + \alpha_t I)^{-1} \hat{J}_t^{T} (w_k \tag{15}
$$

Note   is a K by N matrix, and it leads to inverting an N by N matrix. Here N is the number of weights. This problem can be further simplified using the matrix Inversion. This states that if a matrix O satisfies

$$O = P^{-1} + QR^{-1} \tag{16}$$
$$O^{-1} = P - PQ(R + Q^T PQ)^{-1}Q \tag{17}$$
$$O = \hat{J}_t(w_k)\hat{J}_t^T(w_k) + \tag{18}$$
$$P = \tag{19}$$
$$Q = \hat{J}_t^T() \tag{20}$$
$$R \tag{21}$$

By Substituting equations (18), (19), (20) and (21) into equation (17), that can obtain

$$\left(\quad \hat{J}_t^T(w_k)\hat{J}_t(w_k) + I \right)^{-1} = \frac{1}{\alpha_t}I - \frac{1}{\alpha_t^2}\hat{J}_t^T(w_k)\left(1 + \frac{1}{\alpha_t}\hat{J}_t(w_k)\hat{J}_t^T(w_k)\right)^{-1}\hat{J}_t() \tag{22}$$

Note that in the right side of equation (22), the matrix to be inverted is of size K by K. In every application, N, which is the number of weights, is much greater than K, which is number of outputs. By inserting equation (22) into equation (15) one may have

$$w_{k+1} = w_k - \left[\frac{1}{\alpha_t}I - \frac{1}{\alpha_t^2}\hat{J}_t^T(w_k)\left(I + \frac{1}{\alpha_t}\hat{J}_t(w_k)\hat{J}_t^T(w_k)\right)^{-1}\hat{J}_t(w_k)\right]\hat{J}_t^T(w_k) \tag{23}$$

For single output networks, equation (17) becomes

$$w_{k+1} = w_k - \frac{1}{\alpha_t}\left[I - \frac{\hat{J}_t(w_k)\hat{J}_t^T(w_k)}{\alpha_t + \hat{J}_t(w_k)\hat{J}_t^T(w_k)}\right]\hat{J}_t^T(w_k) \tag{24}$$

As known learning factor, "α" is illustrator of actual output movement to desired output. In the standard LM Method"α" is a constant number. In this method "α" is been modified as "0.01 $E^T E$", Where E is a k x 1 matrix therefore $E^T E$ is a 1 x 1 therefore $[J^T J + \alpha I]$ is invertible. From the equation (24) matrix inversion is not required at all. Therefore if actual output is far than desired output or similarly, errors are large so it converges to desired output with large steps. Likewise when measurement of error is small then actual output approaches to desired output.

NN learning Algorithm Steps
Step 1: Initialization of network weights, learning rate and Threshold error. Set iterations to zero.
Step 2: Total error = zero; iterations → iterations+1.
Step 3: Feed one vector to the input layer.
Step 4: Initialize the target output of that vector.
Step 5: Calculate the outputs of hidden layer units.
Step 6: Calculate the outputs of output layer units.
Step 7: Calculate error (desired output - actual output) and calculate total error.
Step 8: Calculate New Error of output layer units and adjust weights between output and hidden layer.
Step 9: Calculate New Error of hidden layer units and adjust weights between hidden and input layer.
Step 10: While there are more vectors, go to Step 3.
Step 11: If Threshold error >= Total error then stop, otherwise go to Step 2.

5. NN COMPRESSION PROCESS

The Compression Process
Step 1: Read image pixels and then normalize it by converting it from range [0-255] to range [0-1].
Step 2: Divide the image into non-overlapping blocks.
Step 3: Rasterizing the image blocks.
Step 4: Apply the rasterized vector into input layer units
Step 5: Compute the outputs of hidden layer units by multiplying the input vector by the weight matrix (V).
Step 6: Store the outputs of hidden layer units after renormalizing them in a compressed file.
Step 7: If there are more image vectors go to Step 4.
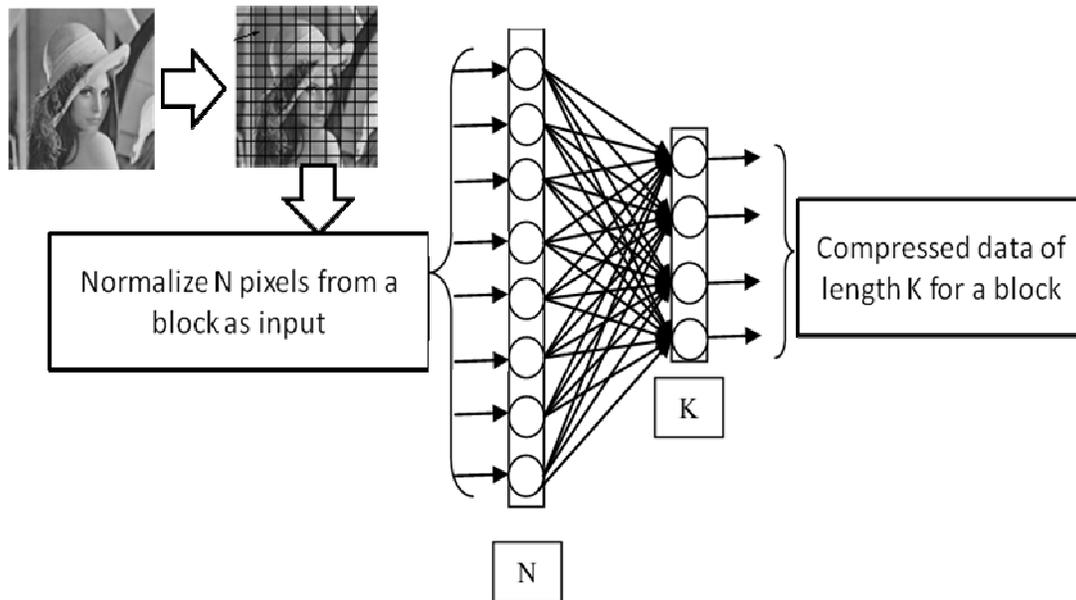
FIGURE10: Neural Network Structure Compression

Decompression Process
Step 1: Take one by one vector from the compressed image.
Step 2: Normalize this vector (it represents the outputs of hidden layer units).
Step 3: The outputs of output layer units by multiplying outputs of hidden layer units by the weight matrix
Step 4: Derasterize the outputs of output layer units to build the sub image
Step 5: Return this sub image to its proper location
Step 6: Renormalize this block and store it in the reconstructed file.
Step 7: If there are more vectors go to Step 1.
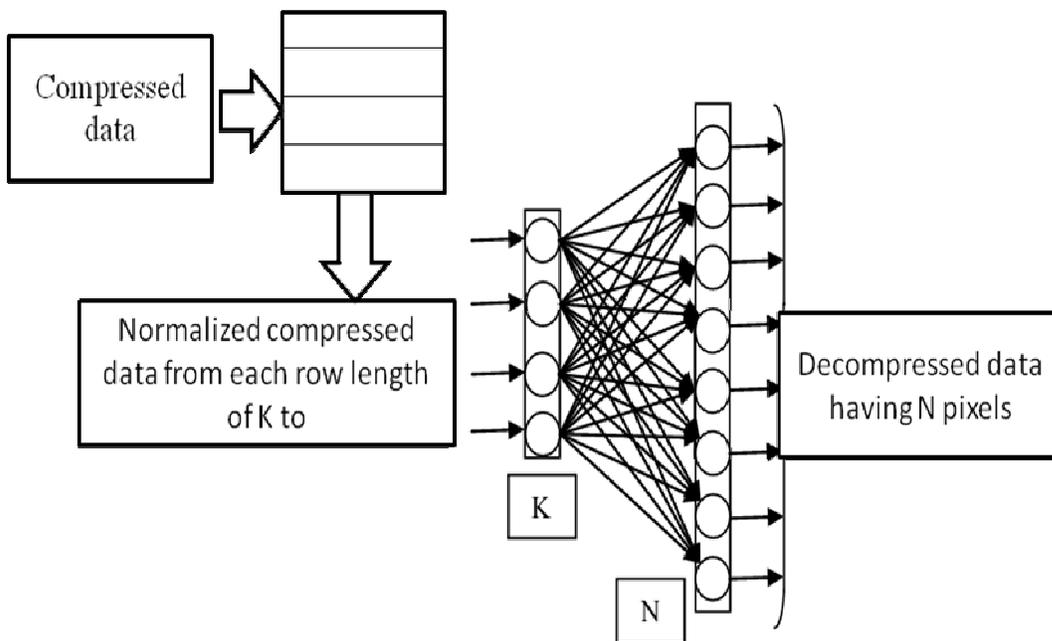
FIGURE 11: Neural Network Structure Decompressions

6. EXPERIMENT RESULTS

The quality of an image is measured using the parameters like Mean Square Error (MSE) and Peak Signal to Noise ratio (PSNR). MSE and PSNR are the parameters which define the quality of an image reconstructed at the output layer of neural network. The MSE between the target image and reconstructed image should be as small as possible so that the quality of reconstructed image should be near to the target image. Ideally, the mean square error should be zero for ideal decompression. The compression-decompression error is evaluated by comparing the input image and decompressed image using normalized mean square error formulas.

In ANN image compression system, the CR is defined by the ratio of the data fed to the input layer Neurons (N_i) to the data out from the hidden layer neurons (N_h). Also the Compression Ratio Performance can be computed by the equation:

$$CR = (1- (N_h/N_i)) \times 100\% \tag{25}$$

Transmission Time is the ratio of number of pixel × number of bits pixel / Modem Speed (Kilo bytes/sec).

The MSE metric is most widely used for it is simple to calculate, having clear physical interpretation and mathematically convenient. MSE is computed by averaging the squared intensity difference of reconstructed image ŷ and the original image, y. Then from it the PSNR is calculated. Mathematically,

$$MSE = 1/CD \, [y_k - \hat{y}_k]^2 \tag{26}$$

In this section, the implementation results of the compression algorithms with structure are studied and compared with Levenberg-Marquardt Algorithm and Modified Levenberg-Marquardt

K.Siva Nagi Reddy, Dr.B.R.Vikram, L.Koteswara Rao, B.Sudheer Reddy

Algorithm Evaluation criteria used for comparison, is compression ratio and the PSNR. For an Image with C rows and D columns PSNR is defined as follow:

$$PSNR = 10 \log_{10} [255^2/MSE] \text{ (DB)} \qquad (27)$$

Where C is Row D is Column .In above equation 255 represents maximum gray level value in an 8-bit image. This criterion is an acceptable measurement for assessing the quality of the reconstructed image. In both structures image blocks of size 8×8 were used and as a result both input and output layer have 64 neurons. In a structure 1, 016 neurons were used in the hidden layer .so it will results in the fixed 4:1 compression ratio. This structure was trained with the 64× 64,128×128,256×256 with standard Image which Includes 1024 training pattern. The experimental results are shown in Table1, 2and 3. One can easily conclude that this structure has a favorable result on the trained Images.

64 x 64	Modified Levenberg-Marquardt Algorithm			Levenberg-Marquardt Algorithm		
Images	PSNR	MSE	TIME	PSNR	MSE	TIME
Lena	32.749	13.8412	312.69	31.1650	14.2209	950.047
Crowd	18.8411	33.9374	297.61	16.369	35.1832	908.957
Pepper	26.9293	17.4648	323.91	23.5447	18.5536	1006.22
Cameraman	22.2501	58.4495	303.02	20.2811	62.6332	876.423

TABLE 1: Performance and comparison of existing and proposed technique for image size 64 X 64

128 x128	Modified Levenberg-Marquardt Algorithm			Levenberg-Marquardt Algorithm		
Images	PSNR	MSE	TIME	PSNR	MSE	TIME
Lena	35.3545	20.6128	1270.701	33.9659	20.2829	3869.427
Crowd	22.4568	33.5796	3515.639	18.6568	35.2746	3635.639
Pepper	28.3607	28.2945	3683.412	27.2607	20.3975	3883.412
Cameraman	29.1466	16.5404	3542.444	28.2466	18.0402	3742.444

TABLE 2: Performance and comparison of existing and proposed technique for image size 128 X 128

256 x256 Images	Modified Levenberg-Marquardt Algorithm			Levenberg-Marquardt Algorithm		
	PSNR	MSE	TIME	PSNR	MSE	TIME
Lena	38.6035	20.1279	3270.101	36.6596	21.1839	3669.274
Crowd	23.4564	30.2496	3115.339	21.5686	35.1796	3515.396
Pepper	32.1603	24.2945	3983.112	28.6076	30.3454	3783.124
Cameraman	34.5466	28.3043	4042.264	30.4662	30.104	3342.444

TABLE 3: Performance and comparison of existing and proposed technique for image size 256 X 256

Camera man
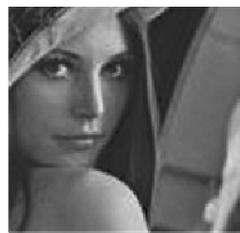
Lena

Pepper

Crowd

Pepper
Crowd

FIGURE 12: Neural Network compressions   **FIGURE 13:** Neural Network Reconstruction

7. CONCLUSION
In this paper the use of Multi -Layer Perception Neural Networks for image compression is reviewed. Since acceptable result is not resulted by compression with one network, a new approach is used by changing the Training algorithm of the network with modified LM Method. The proposed technique is used for image compression. The algorithm is tested on varieties of benchmark images. Simulation results for standard test images with different sizes are presented. These results are compared with L-M method. Several performance measures are used to test the reconstructed image quality. According to the experimental results, the proposed technique with modified L-M method outperformed the existing method. It can be inferred from experimental results as shown in Table 1, 2 and 3 that the proposed method performed well and results higher compression ratio. Besides higher compression ratio it also preserves the quality of the image. It can be concluded that the integration of classical with soft computing based image compression enables a new way for achieving higher compression ratio.

REFERENCES
[1]   R. P. Lippmann, "An introduction to computing with neural network", IEEE ASSP mag., pp. 36-54, 1987.

[2]   M.M. Polycarpou, P. A. Ioannou, "Learning and Convergence Analysis of Neural Type Structured Networks", IEEE Transactions on Neural Network, Vol 2, Jan 1992, pp.39-50.

[3]   K. R Rao, P. Yip, Discrete Cosine Transform Algorithms, Advantages, Applications, Academic Press, 1990

[4]   Rao, P.V. Madhusudana, S.Nachiketh,S.S.Keerthi, K. "image compression using artificial neural network".EEE, ICMLC 2010, PP: 121-124.

[5]   Dutta, D.P.; Choudhury, S.D.; Hussain, M.A.; Majumder, S.; "Digital image compression using neural network" .IEEE, international Conference on Advances in Computing, Control, Telecommunication Technologies, 2009. ACT '09.

[6]   N.M.Rahim, T.Yahagi, "Image Compression by new sub-image bloc Classification techniques using Neural Networks", IEICE Trans. On Fundamentals, Vol. E83-A, No.10, pp 2040-2043, 2000.

[7]   M. S. Rahim, "Image compression by new sub- image block Classification techniques using neural network. IEICE Trans. On Fundamentals of Electronics, Communications, and Computer Sciences, E83-A (10), (2000), pp. 2040- 2043.

[8]   D. Anthony, E. Hines, D. Taylor and J. Barham, "A study of data compression using neural networks and principal component analysis, "in Colloquium on Biomedical Applications of Digital Signal Processing,1989, pp. 1–5.

[9]   G. L. Sicuranzi, G. Ramponi, and S. Marsi, "Artificial neural network for image compression," Electronics Letters, vol. 26, no. 7, pp. 477– 479, March 29 1990.

[10]   M.Egmont-Petersen, D.de.Ridder, Handels, "Image Processing with Neural Networks – a review", Pattern Recognition 35(2002) 2279-2301 [11] M. H. Hassoun, Fundamentals of Artificial Neural Networks, MIT Press, Cambridge, MA, 1995.

[12]   Wasserman, P.D., 1989. Neural Computing: Theory and Practice. Coriolis Group, New York, USA, ISBN: 10: 0442207433, pp: 230.

[13]   Durai S.A. And E.A. Saro, 2006. Image compression with back-propagation neural network using cumulative distribution function. World Acad. Sci. Eng. Technol., 17: 60-64.

[14]   Xianghong, T. And L. Yang, 2008. An image compressing algorithm based on classified blocks with BP neural networks. Proceeding of the International Conference on Computer Science and Software Engineering, Dec. 12-14, IEEE Computer Society, and Wuhan, Hubei pp: 819-822.

[15]   Veisi, H. And M. Jamzad, 2009. A complexity-based approach in image compression using neural networks. Int. J. Sign. Process. 5: 82-92.

[16]   B. M. Wilamowski, Y. Chen, A. Malinowski, "Efficient algorithm for training neural networks with one hidden layer," In Proc. IJCNN, vol.3, pp.1725-728, 1999.

[17]   T. Cong Chen, D. Jian Han, F. T. K. Au, L. G.Than, "Acceleration of Levenberg-Marquardt training of neural networks with variable decay rate", IEEE Trans. on Neural Net., vol. 3, no. 6, pp. 1873 - 1878, 2003. World Academy of Science, Engineering and Technology 6 2005.

Electronic Nose For Black Tea Quality Evaluation Using Kernel Based Clustering Approach

Ashis Tripathy   *ramakrishnam1984@gmail.com*
Assistant Professor
Department of Electronics & Instrumentation Engineering
Siksha O' Anusandhan University
Bhubaneswar, Odisha, 751030, India

A. K. Mohanty   *akmohanty@gmail.com*
Assistant Manager, OPTC Ltd.
Balasore, Odisha, 756029, India.

Mihir Narayan Mohanty   *mihir.n.mohanty@gmail.com*
Associate Professor
Department of Electronics & Instrumentation Engineering
Siksha O' Anusandhan University
Bhubaneswar, Odisha, 751030, India

Abstract

Black Tea is conventionally tested by human sensory panel called "Tea Tasters", who assign quality scores to different tea samples. This paper proposed a method of separation using the device named as electronic noise. The various tea samples have been analyzed using the popular method of separation, like PCA and LDA. For better separation among different scores of tea samples, the kernel based PCA as well as kernel based LDA methods have been considered in this case as the clustering algorithm. The method exhibits a better performance than those of traditional methods. Also the separation index has been evaluated and shows its efficacy.

Keywords: Kernel, Feature Space, Nonlinear Mapping, Electronic Nose, Black Tea, PCA, KPCA, LDA, KLDA.

---

1. INTRODUCTION

The electronic nose technology has been successfully employed for recognition and quality analysis of various food and agro products, viz., wine [1], cola [2], meat [3], fish [4], coffee [5], etc. Instrumental evaluation of black tea quality is quite complex because of the presence of innumerable compounds and their multidimensional contribution in determining the final quality of tea. Present day practice in the tea industry is that experienced tea tasters are employed for this purpose and gradation of tea is done based on their scores. This method is purely subjective and an objective assessment using a low-cost instrument is a dire necessity in the tea industry today. For evaluation of quality of black tea aroma using electronic nose, pioneering work had been done by Dutta et al.[6], where the efficacy of the electronic nose instrument in classifying black tea aroma in different processing stages was established. Co-relation of electronic nose data and tea taster's score has been demonstrated in [7], wherein a stable model was developed and applied on data collected from some gardens of north and north-east India. Electronic nose has also been used successfully for detection of optimum fermentation time during the tea manufacturing process [8].

Different scores of tea samples have been separated using Kernel based Principal Component Analysis (KPCA) and Kernel based Linear Discriminant Analysis (KLDA) clustering techniques. KPCA and KLDA are the excellent statistical learning techniques [9]. Those methods are used to cluster the different groups of data samples. Due to its flexibility and good performance these are

widely applied to various learning scenarios. Clustering of black tea aroma using electronic nose has been considered. The applicability of kernel principal component analysis and kernel linear discriminant analysis for data clustering has been demonstrated.

In this paper Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), KPCA, KLDA techniques have been examined successfully and their response are shown in fig 1, fig 2, fig 3, fig 4, respectively. Also the separation index is calculated for all the above techniques to identify the clarity of clustering in between different groups of data samples. Here the kernel based clustering is introduced and designed for the quality evaluation of tea samples and also various kernels have been tested.

The paper is organized as follows. Section-1 has introduced the work, Section-2 is the explanation of experimental setup, Section-3 describes the methodology for data analysis. Section-4 explains the separation index as the measuring parameter for separation, Section-5 discusses the result and section-6 concludes the work.

2. EXPERIMENTAL SETUP

2.1. Customized Electronic Nose Setup for Black Tea
A customized electronic nose setup has been developed for quality evaluation of tea aroma. The electronic nose consists of metal oxide semiconductor sensors. Each sensor is meant to sense specific chemicals for various tea samples. Five gas sensors from Figaro, Japan – TGS-832, TGS-823, TGS-2600, TGS-2610 and TGS-2611 constitute the sensor array for the setup. The experimental conditions of the electronic nose for classification of black tea aroma are given as follows:

- Amount of black tea sample = 50 grams,
- Temperature $= 60^0 C \pm 3^0 C$,
- Headspace generation time = 30s,
- Data collection time =100s,
- Purging time = 100s,
- Airflow rate = 5 ml/s.

Dry tea samples have been used during the experiments in order to avoid the effect of humidity. The above experimental conditions have been optimized for black tea quality evaluation on the basis of repeated trials and sustained experimentation.

2.2. Sample Collection
One of the major problems is to collect the tea samples, as the tea industries are spread over dispersed locations in India. Also the quality of tea varies considerably on agro-climatic condition, type of plantation, season of flush and method of manufacturing. Experiments were carried out for approximately one-month duration each at the tea gardens of the following industries:
i)   Khongea Tea Estate
ii)   Mateli Tea Estate
iii)   Glenburn Tea Estate
iv)   Fulbari Tea Estate

The industries have multiple tea gardens spread across north and north-east India and the teas produced in their gardens are sent everyday to the tea tasting centers for quality assessment. All the companies had expert tea tasters and for our experiments, one expert tea taster was deputed by the respective industries to provide taster's score to each of the samples, which were subsequently considered for the discrimination study with the computational model.

3. Data analysis METHODOLOGY

3.1. Principal Component Analysis (PCA)

Principle component analysis has been widely used in modeling the statistics of a set of multi-dimensional data [10]. Based on PCA, KPCA method provides a technique for nonlinear feature extraction in the sample data. The nonlinearity is introduced by first mapping the data from the original input space into a higher dimensional feature space F using a nonlinear map $\Phi : R^N \rightarrow F$,where R is the set of real numbers and N is the dimension of the original input space and linear PCA is then performed in F using the mapped samples $\Phi(X_k)$, X_k being the sample data.

By using PCA data may be expressed and presented in such a way as to highlight their similarities and differences. Since patterns in data can be difficult to observe in data of high dimension, PCA is a powerful tool for analyzing data. In the vector space, PCA identifies the major directions, and the corresponding strengths, of variation in the data. PCA achieves this by computing the eigenvectors and eigenvalues of the covariance matrix of the dataset. Keeping only a few eigenvectors corresponding to the largest eigenvalues, PCA can be also used as a tool to reduce the dimensions of the dataset while retaining the major variation of the data.PCA Plot for 174 data samples are shown in fig.1.
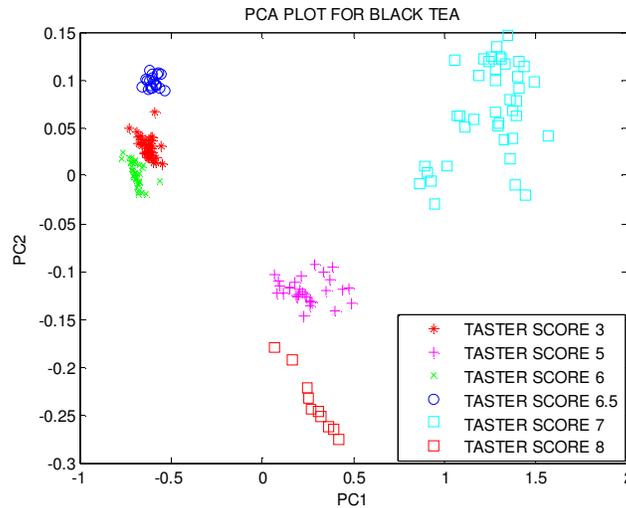
FIGURE 1: PCA plot for black tea samples

3.2. Kernel Principal Component Analysis (KPCA)

KPCA is an extension of principal component analysis using techniques of kernel methods [11]. In general, linear operation of PCA is done in a reproducing kernel Hilbert space with a non-linear mapping. In kernel PCA [12], an arbitrary function Φ is chosen, which is non-trivial. Generally the dimension of arbitrary function Φ is very high. So we generally try to avoid working in the Φ-space, which is known as 'feature space '.By using function Φ we can create the N-by-N kernel K=k(**x,y**)=(Φ(X),Φ(Y)) which represents the inner product of feature space. The kernel function is meant for the high dimensional features representation and is represented by

$$y_k = \sum_{i=1}^{M} \alpha_i^k k(\mathbf{x_i}, \mathbf{x}) \tag{1}$$

where M, is the number of samples , α_i^k is the ith value of κth eigenvector of kernel K, y_k is the κth (q.............M) value of sample after transforming, $\mathbf{x_i}$ is the ith original sample, **x** is the original sample which is to be transformed, q is the sequence number of the first non zero eigen values in an ascending order. The dot product of the samples in the feature space is defined as

$$k(\mathbf{x_i}, \mathbf{x_j}) = \Phi(X_i)^T g \Phi(X_j) \tag{2}$$

where X_i ,X_j $(i,j = 1,2,\ldots\ldots\ldots M)$ are random samples of data sets. We use polynomial kernel function, which is defined as

$$k(\mathbf{x_i}, \mathbf{x_j}) = (\Phi(X_i) g \Phi(X_j) + 1)^d \tag{3}$$

After getting the value of κ, centralization of data is required to perform an effective principal component analysis, we centralize K to become $K^{'}$.

$$K^{'} = K - 1_N K - K 1_N + 1_N K 1_N \tag{4}$$

where 1_N denotes N X N matrix having each value 1.We use $K^{'}$ to perform kernel PCA. KPCA result for 174 data samples are shown in fig.2.
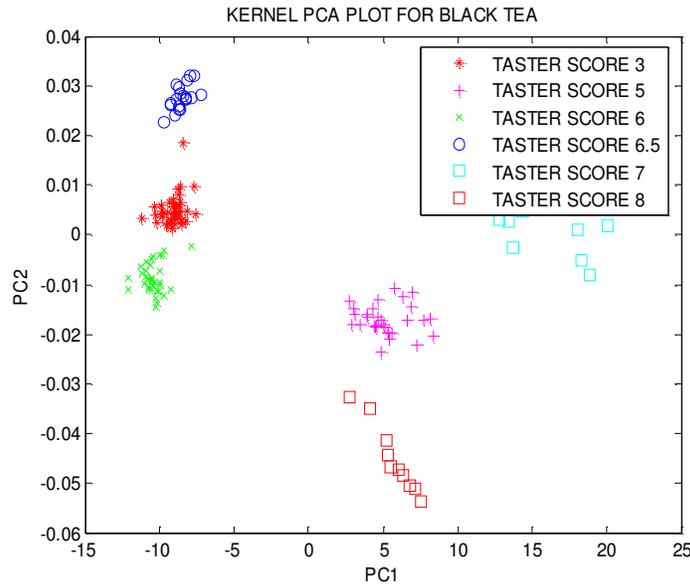
FIGURE 2: KPCA plot for black tea samples

3.3.Linear Discriminant Analysis(LDA)
Dimensionality reduction and classification project high-dimensional data onto a low dimensional space. The data achieves maximum class of separation and is done using Linear Discriminant Analysis (LDA) method [15]. The derived features are linear combinations of the original features. These coefficients are considered from the transformation matrix. In classical LDA, by minimizing the within-class distance and maximizing the between-class distance simultaneously optimal transformation is obtained, thus achieving maximum class discrimination. It has been applied successfully in many applications. Specifically, LDA seeks a transformation matrix W that maximizes the ratio of the 'between class cluster' to the 'within class cluster'.
 The within class scatter matrix 'S_w' and between class scatter matrix 'S_b' are defined as

$$S_w = \sum_{i=1}^{C} \sum_{x \in c_i} (x - \mu_i)(x - \mu_i)^T \tag{5}$$

where
 C - the number of classes ,
 C_i - a set of data belongs to the ith classes and
 μ_i - the mean of ith class.
The within class scatter matrix is the degree of scatter within classes as a summation of covariance matrices of all classes.
 The between class scatter matrix S_b is defined as

$$S_b = \sum_{i=1}^{c} n_i (\mu_i - \mu)(\mu_i - \mu)^T \tag{6}$$

The between class scatter matrix represents the degree of scatter between classes as a covariance matrix of means of all classes. We seek a transformation matrix W that is some sense maximizes the ratio of the between-class scatter and the within class scatter.
The criterion function *J(W)* is defined as

$$J(W) = \frac{W^T S_b^{\Phi} \mathbf{w}}{W^T S_w^{\Phi} \mathbf{w}} \tag{7}$$

The transformation matrix *W* as one that maximizes the criterion function *J(W)* can be obtained. The column of optimal *W* are the generalized eigen vectors $\mathbf{w_i}$ that corresponds to the largest eigen values as

$$S_b \mathbf{w_i} = \lambda_i S_w \mathbf{w_i} \tag{8}$$

The result using LDA for 174 data samples are shown in fig.3.
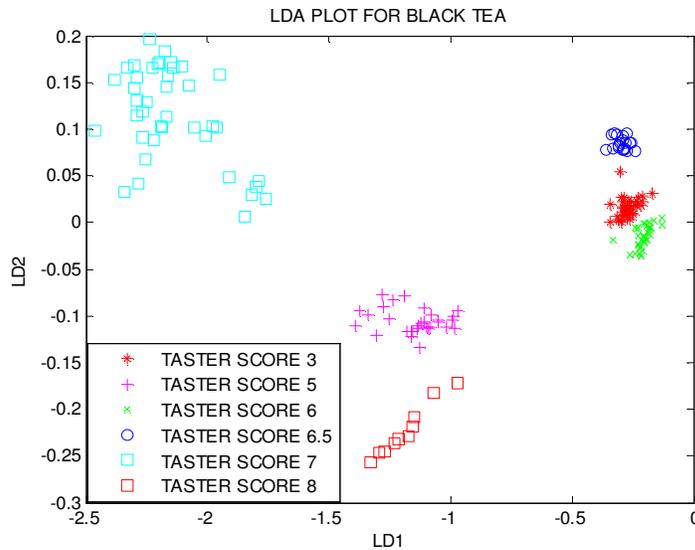
FIGURE 1: LDA plot for black tea samples

3.4. Kernel Linear Discriminant Analysis (KLDA)

The reduction of feature vector components in the feature extraction is done using the standard Linear Discriminant Analysis(LDA). The Kernel Linear Discriminant Analysis (KLDA) is a non-linear expansion of standard LDA to project the feature vectors onto the best discriminant features, while the non-linear projection is implicitly performed by the so called kernel trick. This is a way to represent the scalar-product of non linearly transformed feature vectors without performing the transformation itself. The resulting formulation is expressed as an eigenvalue problem, similar to the linear one. The size of the eigen value problem is equal to the number of input training vectors is a major setback. To get the largest eigen values and the corresponding eigen vectors we use the efficient Kernel Discriminant Analysis (KDA) algorithm.

Since nonlinear extension of LDA, kernel LDA [14] essentially performs LDA in feature space, Φ. Using nonlinear mapping: $\Phi: C \rightarrow f \parallel X \rightarrow \Phi(x)$.
where C is a compact subset of R^N, linearly non-separable configuration becomes separable in Φ.
We can rewrite the objective function as

$$W_{opt} = \arg\max_{w} \left[\left(w^T s_b^{\Phi} w \right) / \left(w^T s_w^{\Phi} w \right) \right] \tag{9}$$

where,

$$S_b^\Phi = \sum_{i=1}^{j} n_i (\mu_j^\Phi - \mu^\Phi)(\mu_j^\Phi - \mu^\Phi)^T \qquad (10)$$

$$S_w^\Phi = \sum_i \left(\Phi(X_i) - \mu_{k_i}^\Phi\right)\left(\Phi(X_i) - \mu_{k_i}^\Phi\right)^T \qquad (11)$$

where Φ is the nonlinear mapping function,
μ_i - the mean of ith class.
X_i – input data set
KLDA result for 174 data samples are shown in fig.4.
Similarly, as the nonlinear extension of LDA, kernel LDA [14] essentially performs LDA in feature space, Φ. Using nonlinear mapping: $\Phi: C \rightarrow f \| X \rightarrow \Phi(x)$; where C is a compact subset of R^N linearly non-separable configuration becomes separable in Φ.
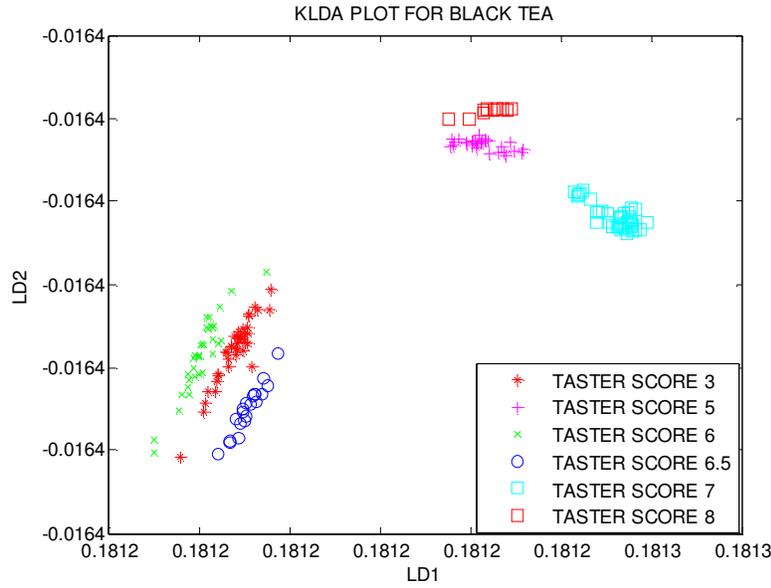
FIGURE 4: KLDA plot for black tea samples

4. SEPARABILITY INDEX
In general, decision on discrimination among data samples is not in proper way. So to extract the clarity of discrimination among the different groups of data samples we need a separability criterion. The separability measure is defined by the ratio of the trace of the 'between class scatter matrix' (S_B) to that of the 'within class scatter matrix' (S_W), and the expressions are given below:

$$S_B = \sum_{i=1}^{c} n_i \left(m_i - m\right)\left(m_i - m\right)^T \qquad (12)$$

$$S_W = \sum_{i=1}^{c} \left(\sum_{j=1}^{n_i} \left(x_{i,j} - m_i\right)\left(x_{i,j} - m_i\right)^T \right) \qquad (13)$$

where C is the number of classes, n_i denotes the number of samples in the ith class, and $x_{i,j}$ denotes the jth sample in the ith class. m_i mean vector of the samples in the ith class and m denotes the mean vector of the samples. Here the separation index values are obtained by using different clustering techniques for the black tea samples.

5. RESULTS AND DISCUSSION

Experimentations with electronic nose have been performed with 174 finished tea samples and sensor output signatures are logged in the computer. Total data size becomes 174×5.i.e.174 samples and each sample having 5 dimensions, as we have five sensors in our electronic nose system. PCA, KPCA, LDA, KLDA techniques have been applied on the 174 data samples .The PCA, KPCA, LDA, KLDA results are shown in Figure1,2,3,and 4 respectively. Due to collection of tea samples from different gardens and difference in their plugging time, samples may have different scores, still it is observed that they are overlapped to each other. Due to agro-climatic condition of a particular location, specific season of flush and clonal variation for the tea plant, the taster score 3 and 6 are overlapped to each other.

In this paper we applied PCA and LDA techniques, but we observed the overlapping nature of data samples, so to avoid this overlapping we applied the nonlinear approach called, KPCA and KLDA techniques, which have given better clarity in clustering as compare to PCA and LDA. Also sometimes it is not possible to properly visualize the clarity of clustering in data samples, so to identify the percentage of clarity of separation we calculated the separation index [18]. We have calculated the separation index for PCA which gives 56.7777, for LDA 56.7964, for KPCA 78.0039, and for KLDA, it is 84.9883. From the above separation index value definitely it has been observed that KPCA has a better clarity than linear PCA, and KLDA has better clarity than linear LDA. Table-1 shows different clustering techniques and their separation index values.

TABLE 1: separation index value in various techniques

Technique	Separation index value
PCA	56.7777
KPCA	78.0039
LDA	56.7964
KLDA	84.9883

6. CONCLUSION

The objective of the work was the clustering of multi sensor array data with tasters' scores in a discriminant fashion using electronic nose. As quality of black tea differ due to seasonal variations and different parameters of tea manufacturing process, it is most important to obtain the performance in discrimination on the different groups of black data samples. The KPCA and KLDA may be helpful as it shows the capability to reduce nonlinearity, and conversion of data samples to high dimensional feature space. Also it is very clear that Kernel approach has a great potential for efficient clustering in different groups of data samples. Further optimization may be applied to reduce the features. Also the accuracy is to be increased. It is left for the future scope.

REFERENCES

[1] J. Lozano, J. P. Santos, M. Aleixandre, I. Sayago, J. Gutierrez, and M. C. Horrillo. "Identification of typical wine aromas by means of an electronic nose." IEEE Sensor J., 6(1) pp. 173-178, 2006.

[2] Kermani, B. G., Schiffman, S. S., and Nagle, H. T, "Performance of the Levenberg-Marquardt neural network training method in electronic nose applications." Sens. Actuators B, 110(1) pp. 13-22, 2005.

[3]   Boothe, D. D. H., and Arnold, J. W."Electronic nose analysis of volatile compounds from poultry meat samples, fresh and after refrigerated storage." J. Sci. Food Agric., 82(3) pp. 315-322, 2002.

[4]   O'Connell, M., Valdora, G., Peltzer, G. and Martin Negri, R., "A practical approach for fish freshness determinations using a portable electronic nose." Sens. Actuators B, **80(2)** pp. 149-154, 2001.

[5]   Pardo, M., and Sberveglieri, G., "Coffee analysis with an electronic nose." IEEE Trans. Instrum. Meas., 51(6) pp. 1334-1339, 2002.

[6]   Dutta, R., Hines, E. L., Gardner, J. W., Kashwan, K. R., and Bhuyan, M., "Tea quality prediction using a tin oxide-based electronic nose: An artificial intelligence approach." Sens. Actuators B, 94, pp. 228-237, 2003.

[7]   Bhattacharyya, N., Bandyopadhyay, R., Bhuyan, M., Tudu, B., Ghosh, D., and Jana, A., "Electronic nose for black tea classification and correlation of measurements with "Tea Taster" marks." IEEE Trans. Instrum. Meas., 57(7), pp. 1313-1321, 2008.

[8]   Bhattacharyya, N., Seth, S., Tudu, B., Tamuly, P., Jana, A., Ghosh, D., Bandyopadhyay, R., Bhuyan, M., and Sabhapandit, S. "Detection of optimum fermentation time for black tea manufacturing using electronic nose." Sens. Actuators B,122(2), pp. 627-634, 2007.

[9]   Scholkopf, B., Smola, A., and Muller, K-R., "Nonlinear component analysis as a kernel eigenvalue problem." Technical Report No.44, Max-Planck Institute, Germany, 1996.

[10]   Wall, M. E., Rechtsteiner, A., Rocha, L. M., "Singular value decomposition and principal component analysis." in A Practical Approach to Microarray Data Analysis, Berrar, D. P., Dubitzky, W., and Granzow, M. Eds., Norwell, MA: Kluwer, (Chapter 5), pp.91–109, 2003.

[11]   Hoffmann, H., "Kernel PCA for novelty detection." Pattern Recognition, 40,pp. 863 – 874, 2007.

[12]   Kim, K.I., Park, S.H., and Kim, H.J "Kernel principal component analysis for texture classification." IEEE Signal Processing Letters, 8(2), pp.39-41., 2001.

[13]   Dy, J. G., and Brodley, C. E., "Feature Selection for Unsupervised Learning." J. Machine Learning Res., 5, pp.845-889, 2004.

[14]   Scholkopf, B., Smola, A., and Muller, K-R.,"Nonlinear Component Analysis as a Kernel Eigenvalue Problem." Neural Computation, vol.10,no.5,pp.1299-1319,1998.

[15]   Duda D.S.R, Hart P. Pattern Classification. Wiley, New York, 2001.

[16]   Belhumeur, N., Hespanha J., and Kriegman, D.," Eigen faces vs. Fisher faces: Recognition Using Class Specific Linear Projection." Proc.ECCV,pp.45-58,1996.

[17]   Liu, C.J, and Wechsler, H.,"A Shape-and Texture-Based Enhanced Fisher Classifier for Face Recognition." IEEE Trans.Image Processing. Vol.10, no.4. pp.598-608, 2001 .

[18]   R. O. Duda, D. G. Stork, P. E. Hart, Pattern classification, 2nd edition, John Wiley and Sons, (pp. 115), (2001).

[19]   Sangita D. Bharkad and Maneshkokare, "Performance evaluation of distance matrices: application to fingerprint recognition." International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), volume: 25, pp.777-806, 2011.

A New Method for Identification of Partially Similar Indian Scripts

Rajiv Kapoor   *rajivkapoor@dce.edu*
Department of Electronics and Communication Engg.
Delhi Technological University
Delhi, India

Amit Dhamija   *dhamija.amit@hotmail.com*
Department of Electrical Engg.
YMCA University of Science and Technology
Faridabad, Haryana, India

Abstract

In this paper, the texture symmetry/non-symmetry factor has been exploited to identify the Indian scripts. Biwavelants have been proposed to obtain the script texture using third order cumulant and bispectra. As the Indian scripts are partially similar to each other, in order to identify them, the samples must include more number of dissimilar characters. The features of individual lines are added repeatedly to enhance the dissimilarity until it reaches to a saturation level which in turn is used to compute a confidence factor i.e. amount of confidence attained in identifying a particular script sample. This variation in confidence factor also gives an estimate of the optimum sample size (number of lines) required for expected results. Cumulants are sensitive to the script curvatures and therefore are most suitable for the partially similar Indian scripts. The double discrete Fourier transform of third order cumulant gives bispectra which estimates the factor of symmetry/non-symmetry in terms of the quadratically coupled frequencies. The envelope of bispectra (biwavelant) obtained using wavelet (db8) provides an accurate behavior of the script texture; which along with Newton-Raphson technique is used to classify the Indian scripts. Various classifiers have been tested for script identification and out of them SVM gives the best results. The method successfully identified the 8 Indian scripts like Devanagari, Urdu, Gujarati, Telugu, Assamese, Gurmukhi, Kannada, and Bangla with desired accuracy.

Keywords: Indian Scripts, Cumulant, Bispectra and Support Vector Machine (SVM)

1. INTRODUCTION

Script identification is a key part of automatic processing of document images. A document script must be known in order to choose an appropriate OCR algorithm. Further processing like indexing or translation of scripts depends on identifying the language used in a document and here again script identification is crucial. Now-a-days documents are stored digitally so as to have quicker access and to save them from any kind of environmental effect. Most of the states in India have their own language of communication and independent scripts. Thus, many official documents are written in regional scripts. Identification of these regional scripts is one of the challenging tasks faced by the designer of an OCR system. Script identification makes the task of analysis and recognition of the text easier by suitably selecting the modalities of OCR. What makes recognition of Indian scripts daunting is their undistinguishable closeness. A number of attempts have already been made to isolate and identify the scripts of the texts in the case of multi-script Indian documents. Patil and Subbareddy [1] developed a system having a feature extractor and a modular neural network. They dilated the documents using 3 x 3 masks in horizontal, vertical, right diagonal, and left diagonal directions. Average pixel distribution was found in these resulting images. A combination of separately trained feed forward neural network was utilized as classifiers for each script. Hochberg [2] approach was to discover frequent character shapes in each script and then look for same instances in new documents. Some

identification techniques have also used the directional features, however to a meager amount. Dhanaya, Ramakrishnan and Pati [3] used basically two features of the scripts like Roman and Tamil. First was Spatial Spread Features like Zonal pixel concentration and character density. Directional Features were detected by using Gabor filter responses. It was concluded that Tamil script has more horizontal lines and strokes while English has more slant strokes. They used Gabor filters to effectively capture the concentration of energies in various directions. Chaudhuri and Pal [4] used skew angle detection for scanned documents containing popular Indian scripts (Devanagari and Bangla). Most characters in these scripts have horizontal line at the top called headlines (Shirorekha). Chaudhuri and Sheth [5] proposed a Gabor filter-based feature extraction scheme for the connected components. Pal and Chaudhuri [6] proposed an automatic technique of separating the text lines using script characteristics and shape-features. Spitz [7] developed techniques for distinguishing the script into two broad classes: Han-based and Latin-based. This classification was based on the spatial relationships of features related to the upward concavities in character structures. Language identification within the Han script class (Chinese, Japanese, and Korean) was performed by analysis of the distribution of optical density in the text images. Tan [8] extracted rotation invariant texture features and then used such features in script identification from document images. Rotation invariant texture features are computed based on the popular multi-channel Gabor filtering technique. Hochberg [9] used features of connected components to classify six different scripts (Arabic, Chinese, Cyrillic, Devanagari, Japanese, and Roman). Srinivasan, Ramakrishnan and Budhlakoti [10] proposed the spatial entropy obtained after decomposing the characters from the document image. The method is not adaptive to the writing styles and moreover after decomposing the characters, the spatial entropy will be definable under so many constraints which have not been discussed. Veena and Sinha [11] proposed a technique using smallest segments of the Devanagari structures to define the Devanagari characters. The method is very time consuming and detection is an issue. Sameer and Lalitha [12] suggested a preliminary technique based upon multiple classifiers like k-means classifier and Minimum Hamming Distance classifier. Anup and Anil [13] could extract temporal information due to online detection recognition and a set of features like Horizontal Inter-stroke Direction for capturing the writing direction like in the case of Arabic which is written from left to right, detection of Shirorekha for Devanagari, average stroke length, number of strokes per unit length, aspect ratio and few more like VD and VID. In these scripts, specific features could work because the scripts chosen for analysis are not related to each other and therefore the task is easier. Second kind of feature is heuristic and depends highly upon the writing style and hence will not work for all Indian scripts because they are highly related. Andrew, Wageeh and Sridharan [14] considered all scripts as texture of their own kind. Yes, this is true but the use of clustering techniques and the wavelet decomposition helps more in case of grey level images as compared to binary. Scripts which are closely related will have similar structures and texture. Texture of the scripts is formed by symmetrical spread of the structural features like horizontal lines, vertical lines and curves. This texture is of binary levels and not like grey ones as in wooden texture. Therefore the kind of features considered by Andrew, Wageeh and Sridhar do not give the high identification accuracy in case of scripts having structural and textural similarity. Morphological reconstruction [15] based upon the continuous erosion and opening was carried out in 4 directions and the average pixel distribution was found as the feature point. MLP [16] has also been used as classifier with the fuzzy-features from the Hough transform. In [17], support vector machine (SVM) based hierarchical classification scheme has been used for the recognition of handwritten Bangla characters. SVM classifier is found to outperform the other classifiers like multilayer perceptron and radial basis function network. [18] elaborates various noises that affect the performance of a script recognition system and the techniques to counter them.

What makes recognition of Indian scripts difficult is their similarity. But, since they are partially similar, their inherent dissimilarity should be enhanced in order to make them completely distinguishable. The features of individual lines are added repeatedly to enhance the dissimilarity until it reaches to a saturation level. As cumulants are sensitive to the script curvatures, they are completely suitable for identifying the Indian Scripts. This paper discusses the use of symmetry/non-symmetry factor of the script texture for identifying the partially similar Indian scripts. Biwavelants have been proposed to obtain the script texture using the third order

cumulant and the bispectra. The double discrete Fourier transform of third order cumulant gives bispectra which estimates the factor of symmetry/non-symmetry in terms of the quadratically coupled frequencies. The envelope of bispectra (biwavelant) provides an accurate behavior of the script texture which along with Newton-Raphson technique is used to classify the Indian scripts. The paper shows that for the proposed feature extraction technique, SVM gives the best classification results and can successfully identify 8 Indian scripts like Devanagari, Urdu, Gujarati, Telugu, Assamese, Gurmukhi, Kannada and Bangla.

The paper is organized as following: Section 2 describes the sample collection and pre-processing of scripts. Section 3 defines the higher order cumulants and the corresponding poly-spectra estimation. It also shows the results obtained by computing the 3rd order cumulant for different script samples. Section 4 describes the optimum parameter selection for the estimation of bispectra. Section 5 introduces biwavelants and shows the corresponding results obtained for the different Indian scripts. Section 6 describes the pre-classification stage using Newton-Raphson Technique. Section 7 elaborates different classifiers used for the proposed feature extraction technique and section 8 gives a comparison of the results obtained with different classifiers. Section 9 concludes the paper.

2. SAMPLE COLLECTION & PRE-PROCESSING

One third of the training and test data set used in this paper was collected from the news papers (*) available online, an equal amount of data was obtained by preparing the documents using different Indian fonts (#) and the last type of data comprised of the handwritten documents (^). The handwritten data was collected on a normal white paper. The documents were written using a blue ball pen. The documents were scanned offline on a canon scanner with 600 dpi resolution. The contents were not fixed and the choice was left to the writer. The total statistics of the sample collected has been mentioned in table (1) below.

Sr. No.	Script	Number of Pages	Number of Lines	Number of Words	Number of Writers
1	Devanagari	5*+10#+5^	210*+351#+100^	7462	5
2	Gujarati	5*+10#+4^	203*+336#+80^	6979	4
3	Gurmukhi	5*+10#+5^	200*+325#+90^	7000	5
4	Telugu	5*+10#+4^	215*+356#+85^	7475	4
5	Kannada	5*+10#+3^	215*+356#+60^	7225	3
6	Bangla	5*+10#+5^	198*+320#+95^	6920	5
7	Assamese	5*+10#+3^	187*+300#+60^	6305	3
8	Urdu	5*+10#+4^	180*+280#+80^	6300	4

Total number of words = 55666

TABLE 1: Number of pages, lines and words collected for each script

The handwritten samples were not just straight lines but had lines and words written irregularly and spread over the whole document. The variety of documents made the task of pre-processing very complex and therefore, the next section is dedicated to the pre-processing.

2.1 Skew Correction

When a document is fed to the optical sensor either mechanically or manually, a few degrees of skew (tilt) is unavoidable. Person scanning the printed data document can also add skew to the text. Hand-written documents written irregularly also contain heavy skew. The lines in sample documents have been written even vertically to each other. To detect the skew angle in the printed documents, we took Radon transform of the whole document image. For an ideal skew free document, peaks corresponding to the horizontal text lines should occur at 90° in Radon image. However the scanned document will actually have peaks at an angle (Ǿ) different from 90° due to the presence of skew. Thus the document is rotated by 90°- Ǿ for skew correction. In figure (1), the Radon transform image of the scanned document showed peaks at Ǿ=80°, thus

document was rotated by 10° anticlockwise to remove the skew error. [19] discusses the method in detail.

(a)   (b)   (c)
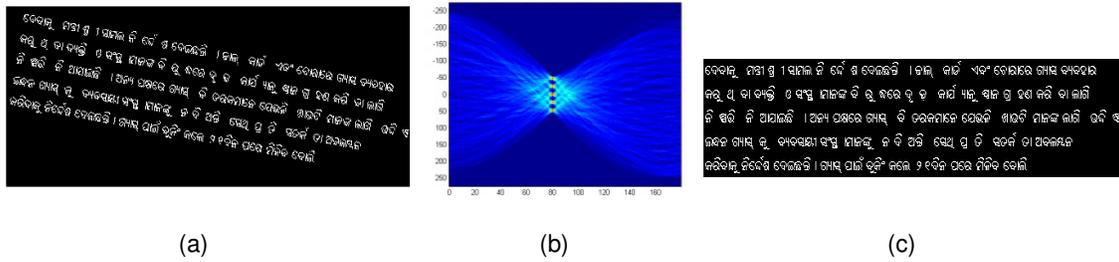
FIGURE 1: (a) The original scanned document image, (b) Radon transform of the document showing peaks at Ø=80° and (c) The document image after skew correction

2.2 Segmentation

Case 1: In figure (2), the lines were separated using horizontal projection and similarly the words were separated using the vertical projection. The printed documents after skew correction could be segmented completely with 100% accuracy. Separated words were concatenated to each other to remove any space in-between them and finally, the words were joined together to make a bigger line. The length of the line was approximately 14 words. These lines were used as input to the next stage of script recognition process. Space between the words was removed to avoid its effect on the result of the cumulant.

(a)   (b)

(c)
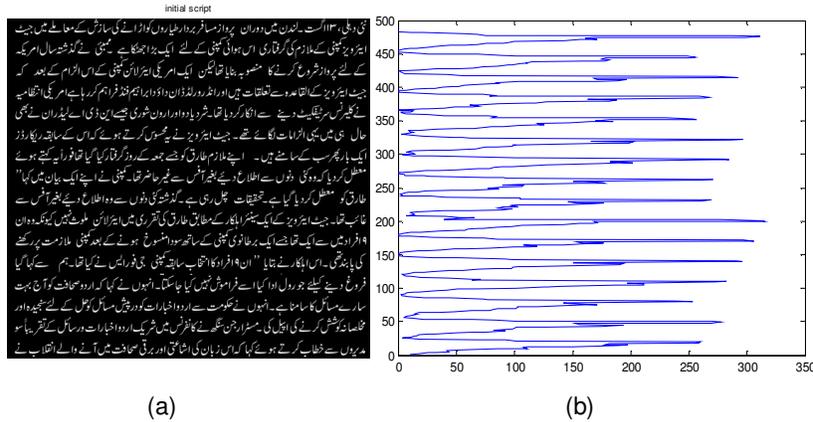
FIGURE 2: First technique - line segmentation
(a) Original script, (b) Horizontal projection and (c) Segmented lines

Case 2:   In some of the hand-written documents, words could not be separated using the projection technique and hence the morphologically conditioned k-means was used to separate them. The structuring element used was a line of three pixel length at an angle of 90°. Size of the structuring element was decided to make the word look like a cluster. When analyzed, the three pixel length was an optimum choice to make the word of any font and size separable. The documents were initially eroded and then k-means clustering method was applied to get the cluster centroids. The major limitation of using k-means is that it requires the optimal number of

clusters as input i.e. the total number of words in the sample document should be known very precisely before applying the unsupervised clustering techniques. MDL (minimum description length) criteria was used [20] to determine the optimum number of clusters (words) for the individual document. k-NN was used to isolate the words of the hand-written document. This method also works perfectly for the documents having words with some ligature connecting them. Figure (3) shows the document and the segmentation results.
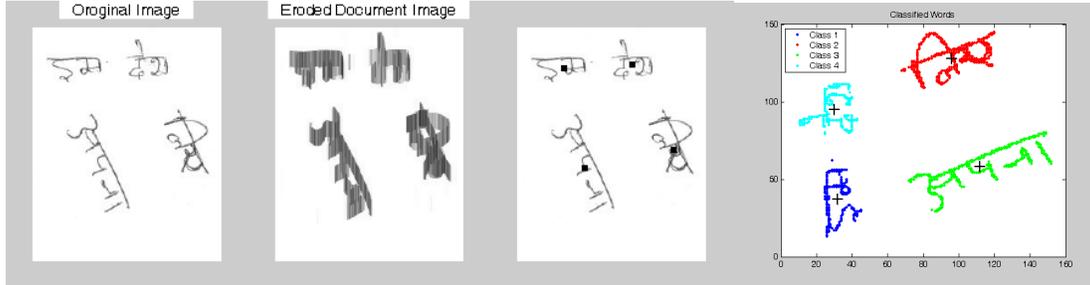
FIGURE 3: Second technique – word segmentation

Space was never allowed to be considered as part of the text for analysis. The script identification is a process which does not consider the space and the carriage return as a part of the text for getting the script features.

3. CUMULANTS

Cumulants are used to extract the inherent features of Indian scripts which are otherwise extremely difficult to extract. Higher order cumulant helps in understanding the multi-dimensional information. Structures are generally specific to the scripts, very complex and some times vary slightly from one script to the other. The paper has successfully attempted to distinguish the Indian Scripts. The first-order cumulant of a stationary process is the mean, $C_{1x} = E\{x(t)\}$. The higher-order cumulants represent central moments and therefore are invariant to the mean shift. Hence, it is convenient to define them under the assumption of zero mean. If the process has a nonzero mean, we subtract the mean and then apply the following definitions to the resulting process. The second, third and fourth-order cumulants of a zero-mean stationary process are defined by equations (1, 2 and 3).

$$C_{2x}(k) = E\{x^*(n)x(n+k)\} \tag{1}$$

$$C_{3x}(k,l) = E\{x^*(n)x(n+k)x(n+l)\} \tag{2}$$

$$C_{4x}(k,l,m) = E\{x^*(n)x(n+k)x(n+l)x^*(n+m)\} - C_{2x}(k)C_{2x}(l-m)$$
$$- C_{2x}(l)C_{2x}(k-m) - M_{2x}^*(m)M_{2x}(k-l) \tag{3}$$

where $M_{2x}(m) = E\{x(n)x(n+m)\}$ and equals $C_{2x}(m)$ for a real valued process. The zero-lag cumulants have special names like $C_{2x}(0)$ is the variance and $\sigma_x^2 : C_{3x}(0,0)$ and $C_{4x}(0,0,0)$ are usually denoted by γ_{3x} and γ_{4x}. We will refer to the normalized quantities $\gamma_{3x}/\sigma_{2x}^3$ as the skewness and γ^4/σ_{2x}^4 as the kurtosis. These normalized quantities are both shift and scale invariant. If $x(n)$ is symmetrically distributed, its skewness is necessarily zero (but not vice versa); if $x(n)$ is Gaussian distributed, its kurtosis is necessarily zero (but not vice versa). Often the terms skewness and kurtosis are used to refer to the un-normalized quantities, γ_{3x} and γ_{4x}. Equation (4) shows that the cumulants of a stationary real-valued process are symmetric in their arguments.

$$C_{2x}(k) = C_{2x}(-k)$$

$$C_{3x}(k,l) = C_{3x}(l,k) = C_{3x}(-k, l-k)$$
$$C_{4x}(k,l,m) = C_{4x}(l,k,m) = C_{4x}(k,m,l) = C_{4x}(-k, l-k, m-k) \tag{4}$$

(a)   (b)

(c)   (d)   (e)

(f)   (g)   (h)
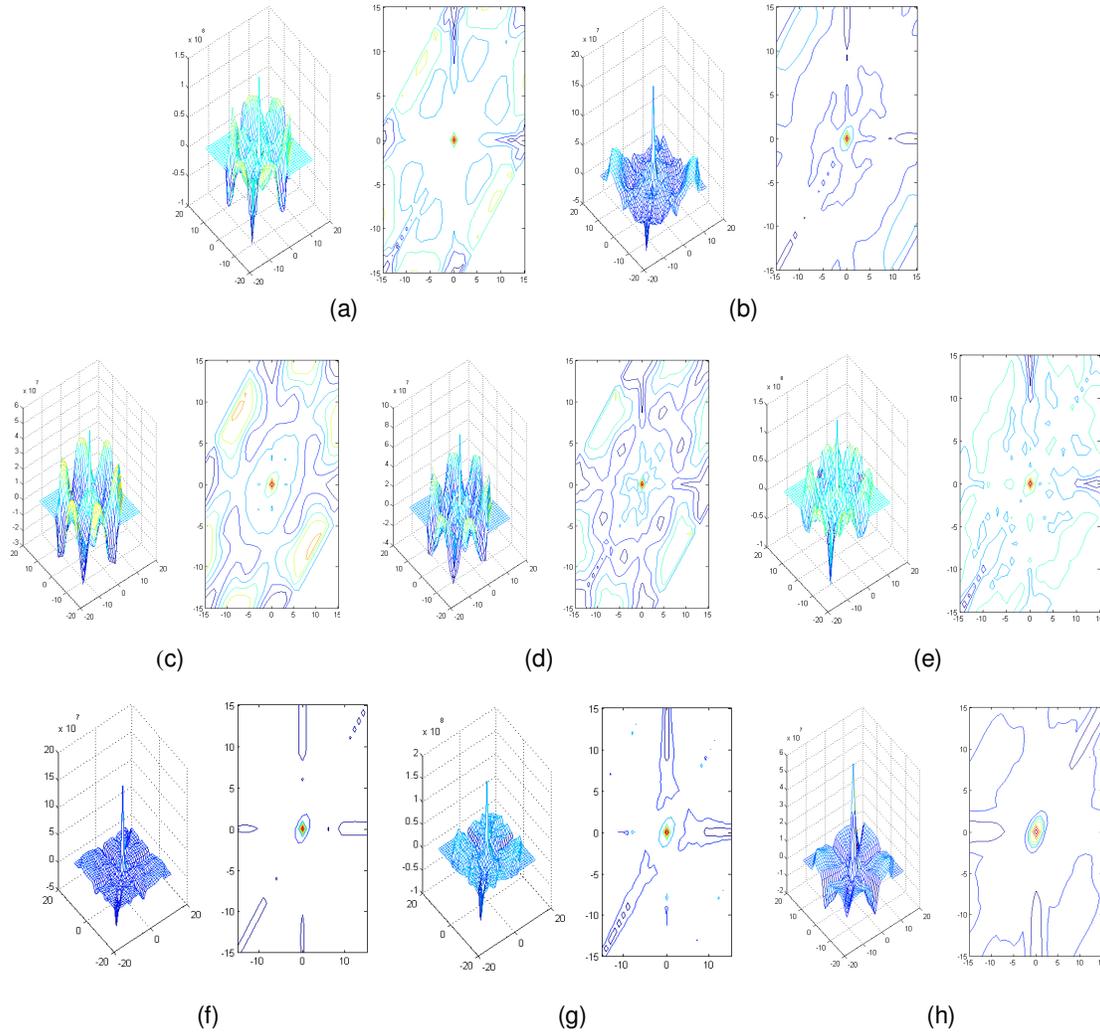
FIGURE 4: 3rd order cumulant computed for different script samples

Figure 4 (a-b) shows the 3rd order cumulant taken for the closely related scripts like Assamese and Bangla. The results demonstrate efficiency of the cumulants to distinguish among the said scripts. Similarly, figure 4(c-h) shows the 3rd order cumulant of the Gujarati, Devanagari, Gurumukhi, Telugu, Kannada and Urdu scripts, respectively. Spectrum of higher order cumulants provides features that are inherent to the script. The kth order poly-spectrum is defined as the FTs of the corresponding cumulant sequence:

$$S_{2x}(f) = \sum_{k=-\infty}^{\infty} C_{2x}(k)\, e^{-j2\pi f\kappa} \tag{5}$$

$$S_{3x}(f_1, f_2) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} C_{3x}(k,l)\, e^{-j2\pi f_1 k}\, e^{-j2\pi f_2 l} \tag{6}$$

$$S_{4x}(f_1, f_2, f_3) = \sum_{k,l,m=-\infty}^{\infty} C_{4x}(k,l,m) \; e^{-j2\pi(f_1k + f_2l + f_3m)} \tag{7}$$

which are the power spectrum, bi-spectrum and tri-spectrum, respectively. In contrast with the power spectrum which is real-valued, bispectra and tri-spectra are complex valued. For a real-valued process, symmetry properties of cumulants are carried forward to the symmetry properties of corresponding poly-spectra. The power spectrum is symmetric: $S_{2x}(f) = S_{2x}(-f)$.

Equation (8) shows the symmetry properties of the bi-spectrum:

$$S_{3x}(f_1, f_2) = S_{3x}(f_2, f_1) = S_{3x}(f_1, -f_1-f_2) = S_{3x}(-f_1, -f_2, f_2) = S_{3x}^*(-f_1, -f_2). \tag{8}$$

Equation (9) shows the symmetry properties of the tri-spectrum:

$$S_{4x}(f_1, f_2, f_3) = S_{4x}(f_1, f_3, f_2) = S_{4x}(f_2, f_1, f_3) = S_{4x}(-f_1, f_2 - f_1, f_3 - f_1) =$$
$$S_{4x}^*(-f_1, -f_2 - f_3) \tag{9}$$

Equation (10) defines the cross-cumulants which are similar to the cross-correlations:

$$C_{xyz}(k,l) = E\{x^*(n)\,y(n+k)\,z(n+l)\} \tag{10}$$

And equation (11) defined the cross-bi-spectrum:

$$S_{xyz}(f_1, f_2) = \sum_{\kappa=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} C_{xyz}(\kappa,l) \; e^{-j2\pi f_1 \kappa} \; e^{-j2\pi f_2 l} \tag{11}$$

Note that the bi-spectrum $S_{3x}(f_1, f_2)$ is a special case of the cross-bi-spectrum obtained when $x = y = z$. The cross-bi-coherence is another useful statistic which is defined in equation (12):

$$bic_{xyz}(f_1, f_2) = \frac{S_{xyz}(f_1, f_2)}{\sqrt{S_{2x}(f_1+f_2)\,S_{2y}(f_1)\,S_{2z}(f_2)}} \tag{12}$$

And the cross-bi-spectrum of three processes is defined in equation (13).

$$b_{xyz}(m,n) = \int \int \ln(S_{xyz}(f_1, f_2)) \; e^{j2\pi f_1 m} \; e^{j2\pi f_2 n} \; df_1 \, df_2 \tag{13}$$

This equation is well-defined only if $S_{xyz}(f_1, f_2)$ is nonzero everywhere. The 3rd order cumulant and its bispectra effectively measure the symmetry/non-symmetry of the structures belonging to different scripts. The results shown in the next section demonstrate that bispectra can effectively differentiate various Indian scripts.

4. BISPECTRA ESTIMATION
The cross-bispectra is estimated as the FT of third-order cross-cumulant of a sequence given by equation (14):

$$I_{xyz}^N(f) = \sum_{k=-N+1}^{N-1} \sum_{l=-N+1}^{N-1} \ddot{C}_{xyz}(k,l) e^{-j2\pi f_1 k} e^{-j2\pi f_2 l}$$
$$= \frac{1}{N^2} X_N^*(f_1 + f_2) Y_N(f_1) Z_N(f_2) \tag{14}$$

Where $X_N(f)$ is the FT of $\{x(n)\}_{n=0}^{N-1}$. This estimate is known as the cross-biperiodogram but it is not a consistent estimate. As in the case of the power spectrum, the estimate can be made consistent by suitable smoothing. The bi-spectrum and the bi-periodogram are special cases obtained when $x = y = z$. Smoothing can be accomplished by multiplying the third-order cumulant estimates by a lag window function. Let $w(t, s)$ be a 2-D window function whose 2-D FT is bounded and nonnegative with the following assumptions given in equation (15):

$$w(0,0) = 1;$$

$$\iint w^2(t,s)dtds < \infty \,;$$

$$\iint f_i^2 W(f_1, f_2)df_1 df_2 < \infty;$$

$$\iint f_i W(f_1, f_2)df_1 df_2 = 0; \tag{15}$$

The window function $w(t,s)$ should also satisfy the symmetry properties of the third-order cumulant. Equation (16) can be used to derive the 2-D lag windows from 1-D lag windows.

$$w(t,s) = w(t)w(s)w(t-s) \tag{16}$$

This satisfies the symmetry conditions of $C_{3x}(m,n)$. Consider the scaled-parameter window $w_M(t,s) = w(t/M, s/M)$ and the smoothed frequency response, given in equation (17).

$$\hat{S}_{xyz}(f_1, f_2) = \sum_{k=-N-1}^{N-1} \sum_{l=-N-1}^{N-1} \hat{C}_{xyz}(k,l) \, w_M(k,l) \, e^{-j2\pi f_1 k} e^{-j2\pi f_2 l} \tag{17}$$

Under the assumption that the cross-bispectrum $\hat{S}_{xyz}(f_1, f_2)$ is sufficiently smooth, the smoothed estimate is known to be consistent with variance given by equation (18).

$$\mathrm{var}(\hat{S}_{xyz}(f_1, f_2)) = \frac{M^2}{N} S_{2x}(f_1 + f_2)S_{2y}(f_1)S_{2z}(f_2)\iint w^2(t,s)dtds \tag{18}$$

for $0 < f_1 < f_2 < \pi$. Note that the implied consistency condition is $M \to \infty$ and $M^2/N \to \infty$ as $N \to \infty$ and $\iint w^2(t,s)dtds < \infty$. Equation (17) is used to estimate the bispectra for $x = y = z$. An alternative approach is to perform the smoothing in the frequency domain. As in the case of power spectra, it is possible to segment the data into K records of length $L = N/K$, compute and average the biperiodogram, and then perform the frequency smoothing using the frequency-domain filter $W_M(f_1, f_2)$ estimated by taking the FT of $w_M(t,s)$. In this case,

$$var\left(\hat{S}_{xyz}(f_1, f_2)\right) = M^2/LK \, S_{2x}(f_1 + f_2)S_{2y}(f_1)S_{2z}(f_2)\iint w^2(t,s)dtds \tag{19}$$

for $0 < f_1 < f_2 < \pi$. Windowing is not required in case K is very large. The following sub-section describes the parameter selection and optimization for the estimation of bispectra.

4.1. Parameters Selection and Optimization
The following table comprises various parameters and their corresponding optimized values required to compute the bispectra of collected script samples. Larger MaxLag gives more number of coupled frequencies and the value lesser than this will make the process of script identification difficult. The value of the MaxLag also depends upon the data size. Here data size means size of the character of a particular script. Maximum value of the MaxLag can be the no. of pixels which describe the height of the character therefore the MaxLag value is proportional to the data size (Height of the character). Hamming window was utilized. The parameters have been optimized for the targeted scripts.

Name of the Parameter	MaxLag	Sample Rate	Window	Scale
Optimized Value	15	5	Hamming	Unbiased

TABLE 2: Optimized parameters for the bispectra estimation of collected samples

The parameters given in table (2) were utilized for the computation of bispectra of various script samples and the results are given below.
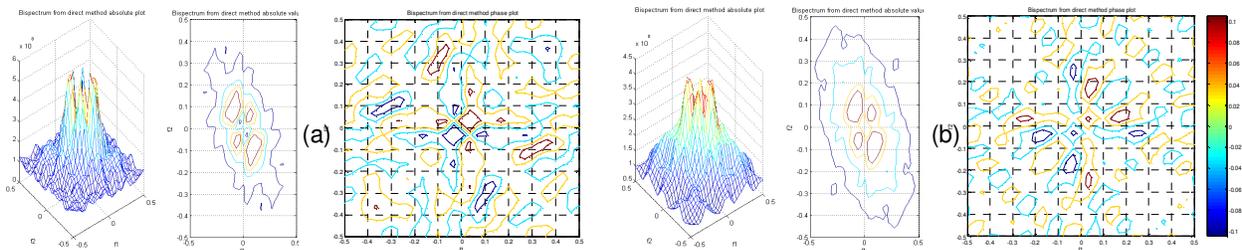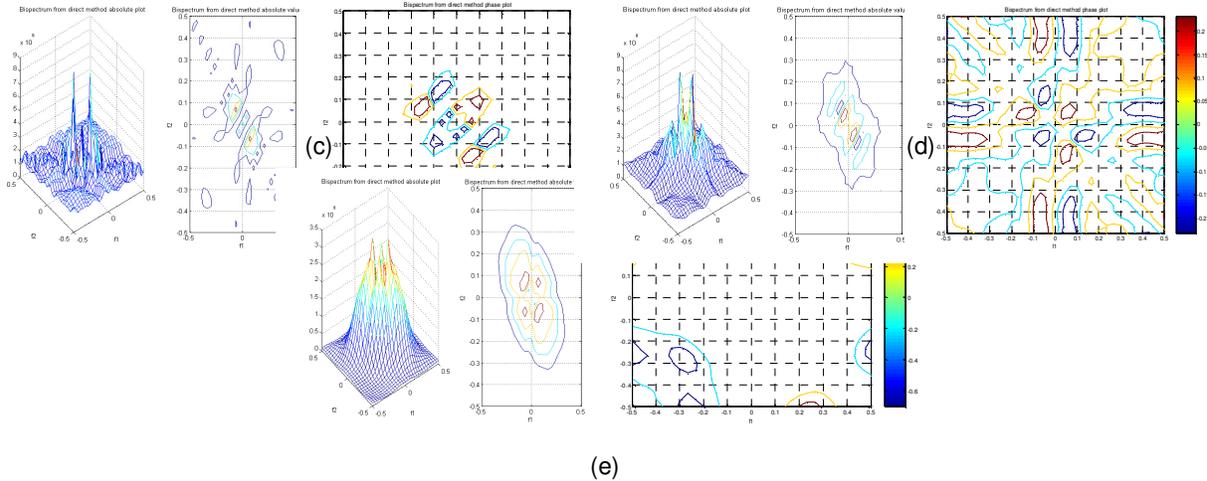
FIGURE 5: (a-e) Bispectra results (magnitude and phase) obtained for Kannada, Telugu, Assamese, Bangla and Urdu scripts, respectively

Figure (5) shows that bispectra can completely distinguish the partially similar Indian scripts.

5. BIWAVELANT

As the Indian scripts are partially similar to each other, in order to identify them, the samples must include more number of dissimilar characters. The features of individual lines are added repeatedly to enhance the dissimilarity until it reaches to a saturation level. The experimental results provided in figure (4) and (5) show that a sample size of 100 lines is sufficient to get the expected results. In order to use bispectra for script identification, the redundant information (high frequency components) is removed keeping only the prominent features (low pass information) which is described further.

5.1 Smoothing Filter vs. Wavelet

Both smoothing filter and wavelet transform can be used to remove the high frequency components from bispectra. But a smoothing filter can't protect the precious details while removing the high pass information, therefore wavelet transform is used. The wavelet transform was introduced in [21], [22], [23] and defined as

$$W_{xy}(b,a) = \frac{1}{\sqrt{a}} \int x(t) y(\frac{t-b}{a}) dt \qquad (20)$$

where $x(t)$ is the signal being transformed and $y(t)$ is the 'analyzing wavelet'. $y(t)$ satisfies the admissibility condition $\int |Y(a)|^2 \frac{d\omega}{\omega} < \infty$ which is equivalent to $\int Y(t)dt = 0$ i.e. a wavelet has zero mean. The use of the wavelet transform as a multi-resolution analysis tool has been widespread involving many applications such as fractal signal analysis, pitch detection and image compression. However, Frisch [24] and Messer [25] took a different interpretation of the continuous wavelet transform and considered it as a two parameter correlation operation where time and dilation are the correlation parameters i.e. $x(t)$ is considered as a received noisy signal with known amplitude, delay and dilated factor. $y(t)$ is the template of the known shape. Therefore using the continuous wavelet transform and an appropriate decision statistic, the detection can be made for a signal buried in Gaussian noise. This interpretation will be later used in the use of wavelants. Two important properties of the cumulants are:

1. The third order cumulant for a Gaussian (or any symmetrically distributed) random process is zero.
2. If a subset of k random variables $\{x_i\}$ is independent of the rest, then the third-order cumulant is zero.

The above formulation exhibits properties closely related to those of cumulants. The higher order wavelant can also be expressed using the Fourier representations of the signals given by equations (21) and (22):

$$W_{XYZ}^3(b_1, a_1; b_2, a_2) = 1/\sqrt[3]{a_1 a_2} \iint S_{3X}(f_1, f_2) Y(a_1 f_1) Z(a_2 f_2) e^{-j(w_1 b_1 + w_2 b_2)} dw_1 dw_2 \qquad (21)$$

$$W_{XXX}^3(b_1, a_1; b_2, a_2) = 1/\sqrt[3]{a_1 a_2} \iint S_{3X}(f_1, f_2) X(a_1 f_1) X(a_2 f_2) e^{-j(w_1 b_1 + w_2 b_2)} dw_1 dw_2 \qquad (22)$$

The 2-D cross-wavelant for an image can be expressed as:

$$W_{xyz}^3(b_{X1}, b_{Y1}, a_{X1}, a_{Y1}; b_{X2}, b_{Y2}, a_{X2}, a_{Y2}) =$$
$$\dots \frac{1}{\sqrt[3]{a_{X1} a_{X2} a_{Y1} a_{Y2}}} \iint X(t_X, t_Y) Y\left(\frac{t_X - b_{X1}}{a_{X1}}, \frac{t_Y - b_{Y1}}{a_{Y1}}\right) Z\left(\frac{t_X - b_{X2}}{a_{X2}}, \frac{t_Y - b_{Y2}}{a_{Y2}}\right) dt_X dt_Y$$
(23)

The equations (21) and (22) define the third-order cross and auto wavelants and equation (23) defines the cross-wavelant for 2D images.

5.2 Properties
When the input used for computing the wavelant is translated and/or followed by dilation then the following properties result

If $\quad x(t), y(t), z(t) \quad$ maps to $\quad W_{XYZ}^3(b_1, a_1; b_2, a_2)$

Then $\quad x\left(\frac{t-\tau}{A}\right), y(t), z(t) \quad$ maps to $\quad W_{XYZ}^3\left(\frac{b_1 - \tau}{A}, \frac{a_1}{A}; \frac{b_2 - \tau}{A}, \frac{a_2}{A}\right)$

and $\quad x(t), \frac{1}{\sqrt{A}} y\left(\frac{t-\tau}{A}\right), z(t) \quad$ maps to $\quad W_{XYZ}^3(b_1 + a_1 \tau, a_1 A; b_2, a_2)$

and $\quad x(t), y(t), \frac{1}{\sqrt{A}} z\left(\frac{t-\tau}{A}\right) \quad$ maps to $\quad W_{XYZ}^3(b_1, a_1; b_2 + a_2 \tau, a_2 A) \qquad (24)$

However, if the input is first dilated and then translated, then the results are given by

If $\quad x(t), y(t), z(t) \quad$ maps to $\quad W_{XYZ}^3(b_1, a_1; b_2, a_2)$

then $\quad x(At - \tau), y(t), z(t) \quad$ maps to $\quad W_{XYZ}^3(Ab_1 - \tau, a_1 A; Ab_2 - \tau, a_2 A)$

and $\quad x(t), \sqrt{A} y(At - \tau), z(t) \quad$ maps to $\quad W_{XYZ}^3\left(\frac{b1 + \tau}{A}, \frac{a_1}{A}; b_2, a_2\right)$

and $\quad x(t), y(t), \sqrt{A} z(At - \tau) \quad$ maps to $\quad W_{XYZ}^3\left(b_1, a_1; \frac{b_2 + \tau}{A}, \frac{a_2}{A}\right) \qquad (25)$

Before applying the wavelet transform, the 2D bispectra is first converted to a 1D frequency response. The following figure shows a comparison of the results obtained at different levels of approximations (low pass filtering) by applying wavelet transform (db8) on the bispectra results.
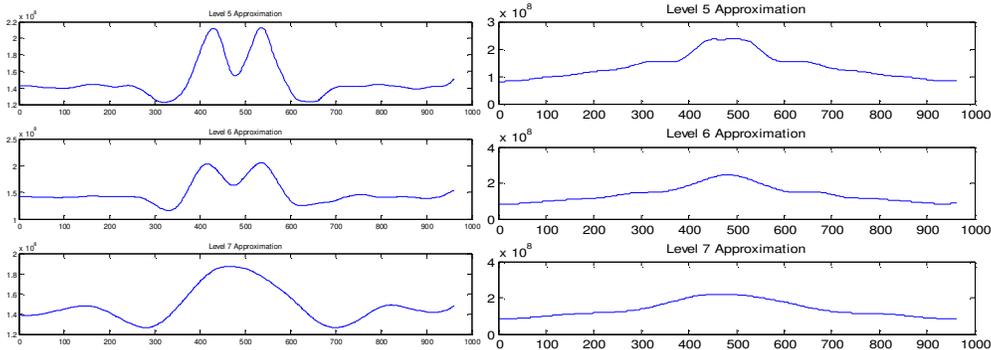
FIGURE 6: Approximation (low pass filtering) results at levels 5, 6 and 7 for
(a) Assamese and (b) Bangla scripts

Figure (6) shows that after the 5th level approximation, we start loosing the precious details and therefore, before using bispectra results for identification/classification, they are approximated only up-to the 5th level. Following figure shows the results obtained for various Indian scripts.

(a)   (b)

(c)   (d)
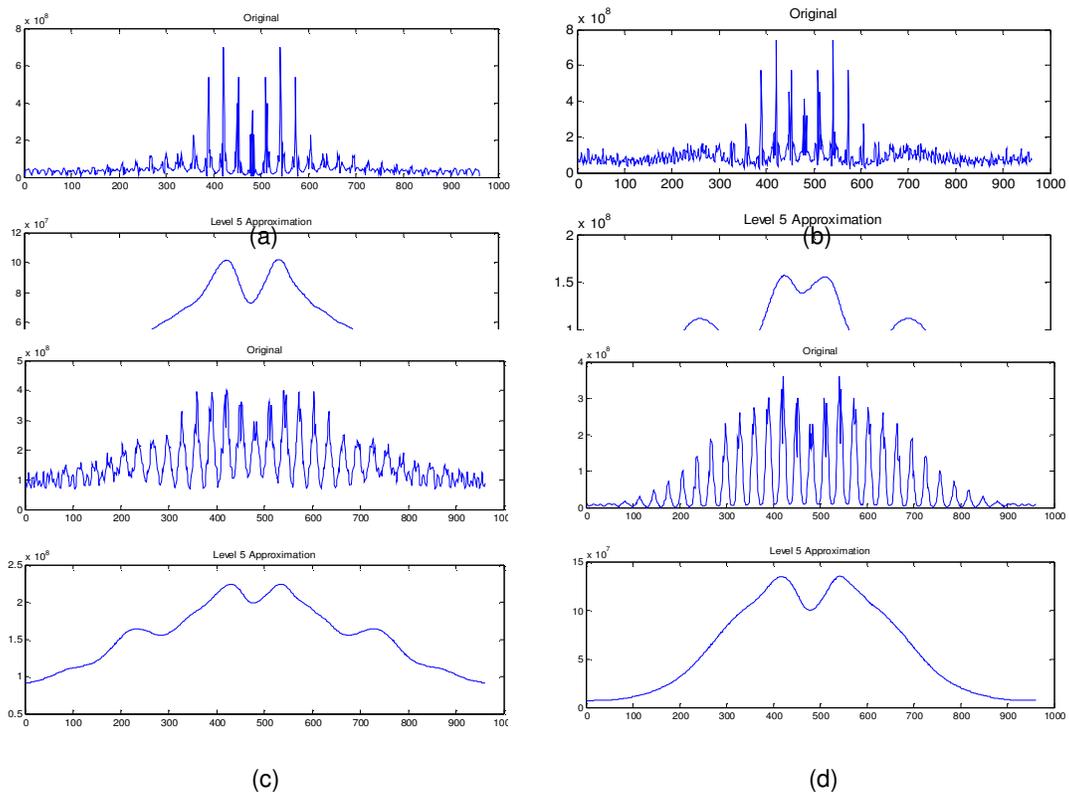
FIGURE 7: Approximated bispectra results for (a) Gujarati, (b) Bangla, (c) Telugu and (d) Urdu scripts

The following figure shows that in addition to the dissimilar results obtained for different scripts, the method gives fairly similar results for same script with different font types and sizes.

(a)   (b)
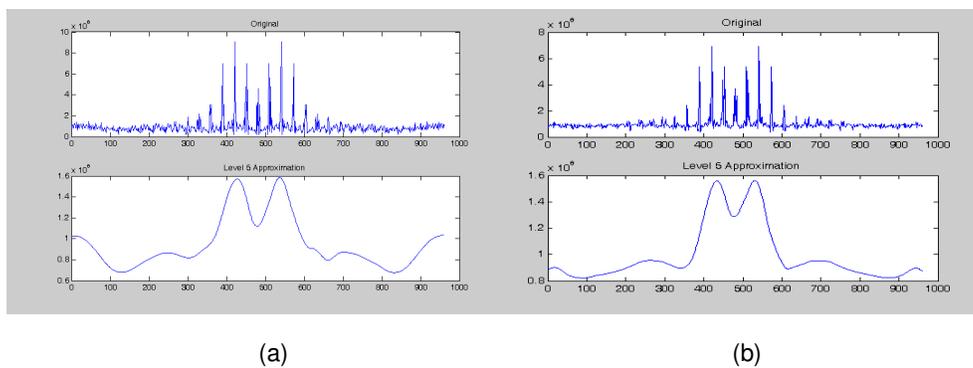
FIGURE 8: Biwavelant results obtained for font size (a) 14 and (b) 16

Figure (8) shows the biwavelant results obtained for Devanagari script for two different font sizes. Figures (7) and (8) illustrate that biwavelant (bispectra + wavelet) gives an envelope of the bispectra which proves to be a convincing feature for script identification.

6. PRE-CLASSIFICATION

6.1. Newton-Raphson Technique

The above results show that a biwavelant envelope can clearly distinguish/identify an Indian script; but using it directly for the classification/identification of a script sample is not suitable because of its high dimensionality. Therefore, Newton-Raphson technique is used to obtain the roots of a biwavelant envelop for each script sample in order to reduce the dimensionality of the feature space. The following table shows that the obtained roots clearly distinguish the Indian scripts.

ROOTS TABLE		
Script	**Root 1**	**Root 2**
Urdu	1.36E+16	2.51E+15
Urdu	1.41E+16	2.45E+15
Urdu	1.39E+16	2.47E+15
Telugu	6.11E+15	1.68E+15
Telugu	6.39E+15	1.66E+15
Telugu	6.35E+15	1.68E+15
Bangla	1.74E+16	1.23E+15
Bangla	1.86E+16	1.26E+15
Bangla	1.62E+16	1.10E+15
Kannada	3.44E+13	1.95E+15
Kannada	3.75E+13	2.12E+15
Kannada	3.94E+13	2.11E+15
Guajarati	4.22E+15	5.10E+14
Guajarati	4.16E+15	5.27E+14
Guajarati	4.18E+15	5.16E+14
Gurmukhi	6.73E+15	9.70E+14
Gurmukhi	5.22E+15	6.60E+14
Gurmukhi	8.86E+15	10.3E+14
Assamese	2.03E+16	4.75E+14
Assamese	2.13E+16	4.78E+14
Assamese	1.97E+16	4.84E+14
Devanagari	1.05E+16	5.93E+14
Devanagari	1.06E+16	6.22E+14
Devanagari	1.06E+16	6.02E+14

TABLE 3: Roots obtained for Indian scripts using the Newton-Raphson technique

Script	No. of Lines Used	Root-1	Root-2
Assamese	110	1.70E+16	3.99E+14
Assamese	105	1.55E+16	3.64E+14
Assamese	100	1.42E+16	3.29E+14
Assamese	90	1.14E+16	2.67E+14
Assamese	80	8.97E+15	2.11E+14
Assamese	70	6.75E+15	1.63E+14
Assamese	60	4.85E+15	1.23E+14
Assamese	50	3.37E+15	8.56E+13
Assamese	40	2.07E+15	5.62E+13
Assamese	30	1.14E+15	3.23E+13
Assamese	20	5.16E+14	1.43E+13
Assamese	11	1.58E+14	4.35E+12
Assamese	6	4.69E+13	1.30E+12
Assamese	3	1.22E+13	3.14E+11

TABLE 4: Variation of the roots for Assamese script with the number of lines used in the paragraph of a script sample

This variation of roots is plotted and shown in the following figure (9).

(a)   (b)
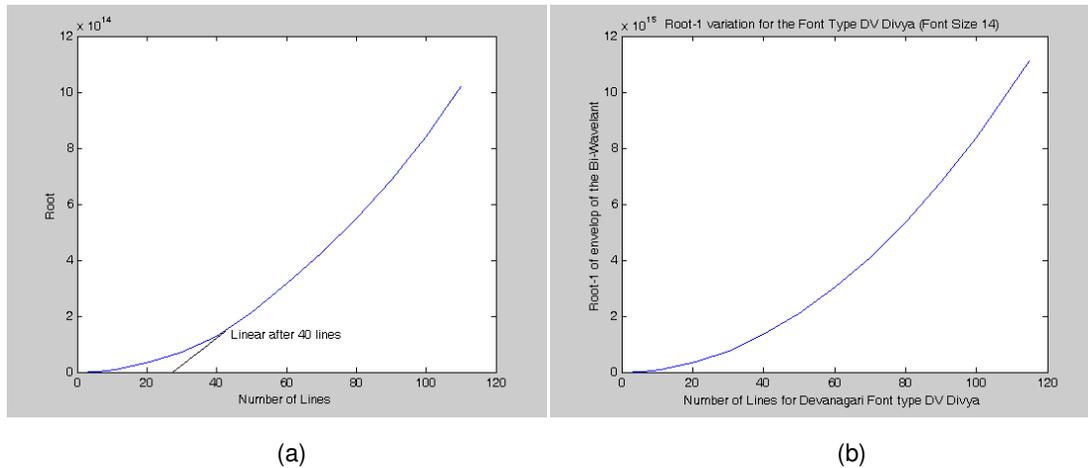
FIGURE 9: Variation of the roots with number of lines used in the paragraph of a script sample for
(a) Guajarati, (b) Devanagari scripts

Figure (9) shows that the variation of roots is linear to the sample size (no. of lines) and it holds for all Indian scripts. The small non-linear portion is common to all scripts and hence, the deferential effect gets cancelled. In order to estimate the level of confidence attained in identifying a script sample and the possibility of making an erroneous decision, two parameters, $ConfidenceLevel$ and $ErrorPossibility$ are defined in equation (26) as a function of the number of lines constituting the test sample:

$$ConfidenceLevel = \frac{\sum_{k=1}^{L-1} Cum(k) - Cum(L)}{\sum_{k=1}^{L-1} Cum(k)}$$

$$ErrorPossibility = 1 - ConfidenceLevel \tag{26}$$

where $Cum(k)$ represent the third order cumulant of the kth line and L is number of lines in the script sample. For a particular script sample $4 \leq L \leq 120$. The following figure shows the variation of $ConfidenceLevel$ and $ErrorPossibility$ with the number of lines in the script sample.

(a)   (b)
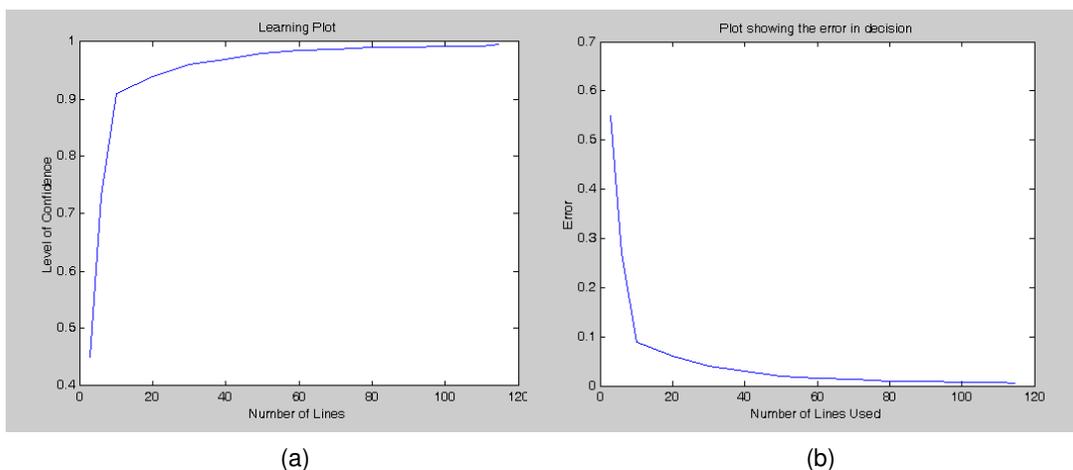
FIGURE 10: Variation of (a) $ConfidenceLevel$ and (b) $ErrorPossibility$
with the number of lines used by the algorithm to identify the script sample

Figure (10) shows that the motive behind using more number of lines for feature extraction is to have a higher confidence level in identifying a script sample.

7. CLASSIFICATION

7.1. k-Nearest Neighbor Classification
The k-means clustering algorithm is a fast, unsupervised, nondeterministic and iterative method for generating a fixed number of disjoint clusters. Each data point is randomly assigned to one of k-initial clusters, such that each cluster has approximately the same number of points. In the subsequent iterations, distance of each point to each of the clusters is calculated using some metric and subsequently moved into the cluster corresponding to the minimum distance. Commonly used metrics are Euclidian distance to the centroid of the clusters or a weighted distance which considers only the closest n-points. The algorithm terminates when no points are moved in a single iteration. As the final result is highly dependent on the initialization of the clusters, the algorithm is often repeated a number of times, with each solution scored according to some evaluation function. The following figure shows the classification results obtained for various Indian scripts using the nearest neighbor classifier.
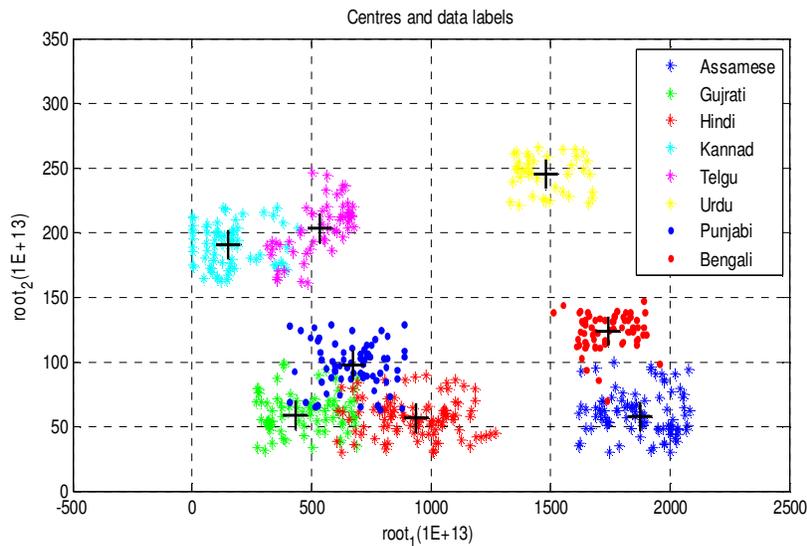
FIGURE 11: Classification results obtained for the Indian scripts

Each point in figure (11) represents the feature vector (root1, root2) corresponding to a script sample. Each sample is classified and associated to one of the eight clusters (scripts) i.e. to a particular script. The clusters are shown with different colors and markers for easy understanding. Centroid of each cluster, represented with + in figure (11), is computed using unsupervised k-means clustering and given in the following table.

CENTROIDS		
Script	**Root 1**	**Root 2**
Urdu	1.48E+16	2.45E+15
Telugu	5.36E+15	2.03E+15
Gujarati	4.31E+15	5.90E+14
Bangla	1.74E+16	1.23E+15
Kannada	1.53E+15	1.90E+15
Assamese	1.87E+16	5.70E+14
Gurmukhi	6.73E+15	9.70E+14
Devanagari	9.39E+15	5.60E+14

TABLE 5: Centroids of the clusters representing individual scripts

7.2.   Multi-Layer Perceptron

The designed MLP network with logistic outputs has been trained with a quasi-Newton optimization algorithm and various other optimized parameters given below in table (6). The multilayer perceptron network takes two dimensional feature vectors as input.

MULTI-LAYER PERCEPTRON NETWORK		
Sr. No.	**Parameter Name**	**Parameter Value**
1	Algorithm Used	quasi-Newton optimization algorithm
2	No. of input neurons	2
3	No. of hidden layer neurons	6
4	No. of output layer neurons	1
5	Rate of weight decay	0.2
6	No. of training cycles	100
7	Activation function for hidden neurons	tanh

TABLE 6: Characteristic parameters of the multilayer network used for the training and classification

The classification results obtained using the above MLP network are shown in the following figure.
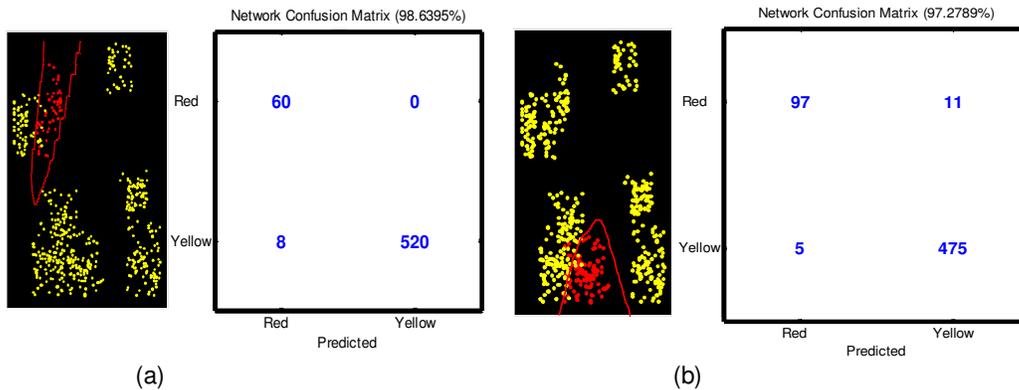
FIGURE 12: Classification results and the corresponding confusion matrix for
(a)   Telugu, (b) Devanagari scripts

7.3. Support Vector Machine

Finally, the one vs. rest support vector machine was used to classify the partially similar Indian scripts. Various optimized parameters used to design the support vector machine are given below.

Sr. No.	Parameter Name	Parameter Value
1	Classifier Type	One vs. Rest
2	Kernel	Radial Basis Function
3	Scale	0.2
4	C	1000
5	Optimizer	libsvm

TABLE 7: Various parameters used to design the support vector machine

The classification results obtained using the proposed support vector machine is given below in the form of a confusion matrix.

Scripts	Assamese	Bangla	Gujarati	Devanagari	Kannada	Gurmukhi	Telugu	Urdu
Assamese	0.9826	0.0174	0	0	0	0	0	0
Bangla	0.0199	0.9801	0	0	0	0	0	0
Gujarati	0	0	0.9442	0.0421	0	0.0137	0	0
Devanagari	0	0	0.0540	0.9172	0	0.0288	0	0
Kannada	0	0	0	0	0.9899	0	0.0101	0
Gurmukhi	0	0	0.0521	0.0464	0	0.9015	0	0
Telugu	0	0	0	0	0.0118	0	0.9882	0
Urdu	0	0	0	0	0	0	0	1.0000

TABLE 8: Classification results obtained for various scripts and the corresponding confusion matrix

8. SCRIPT SAMPLES & CLASSIFICATION RESULTS

The proposed system was tested for eight Indian scripts using both printed and hand written samples as shown in the following figure.

FIGURE 13: Samples of eight Indian scripts used for training and testing of the proposed algorithm

In addition to the above, the proposed system was also tested on same script samples with different font types and sizes as shown in figure (14).

FIGURE 14: Samples of Devanagari script with different font types and sizes

The comparison of identification results obtained with various classifiers mentioned in the previous section is given in table (9).

Sr. No.	Classifier Type	Classification Accuracy (%)
1	Bayes Quadratic	90
2	Decision Layer	91
3	Nearest Neighbour Classifier	87
4	Multi Layer Perceptron	94
5	Support Vector Machine	95

TABLE 9: Classification accuracy with various classifiers

9. CONCLUSION

The method has successfully identified eight Indian scripts and is expected to work for scripts from other nations also. Indian scripts are closely related to each other and as the proposed technique is sensitive to the structural changes in the script, it is able to distinguish them successfully. But, the same sensitivity makes the method vulnerable to noise in the samples, so the document has to be noise free for expected results. However, the pre-processing becomes very complex for the removal of noise from the samples. The features of individual lines were added until they reach to a saturation level. This saturation level in turn helped in determining the confidence level for indentifying a sample. The variation in confidence level with the number of lines in the sample was used to determine an optimum number of lines required in identifying a script. A sample size of 100 lines gives the best result as it considers most of the features in the script. The method works well for both the printed and hand written samples of the scripts, independently. However, it does not work for the sample with a mixture of printed and hand-written lines of a script. Pre-processing of hand-written scripts also adds to the complexity of the method. In such a case of mixed characters, cumulants are not useful as being very sensitive to the curvatures. Indian scripts are partially similar to each other. Because of the partial similarity, we first consolidated on the number of words which enhances the partial dissimilarity and makes it look significant. Then we used the method which is very sensitive to the curvatures and the results were as expected.

REFERENCES

[1]   S. B. Patil and N. V. Subbareddy, "Neural network based system for script identification in Indian documents", *SADHNA,* vol. 27, pp. 82-97, Feb. 2002.

[2]   J. Hochberg, P. Kelly, T. Thomas and L. Kerns, "Automatic script identification from document images using cluster-based templates", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19 (2), pp.176-181, Feb. 1997.

[3]   D. Dhanaya, A. G. Ramakrishnan and P. B. Pati, "Script identification in printed bilingual documents", *SADHNA,* vol. 27, pp. 73-82, Feb. 2002.

Rajiv Kapoor & Amit Dhamija

[4]   B. B. Chaudhuri and U. Pal, "Skew angle detection of digitized Indian script documents", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19 (2), pp.182-186, Feb. 1997.

[5]   S. Chaudhuri and R. Sheth, "Trainable script identification strategies for Indian languages", *International Conference on Document Analysis and Recognition*, pp. 657-660, Sep. 1999.

[6]   U. Pal and B. B. Chaudhuri, "Script line separation from Indian multi-script documents", *International Conference on Document Analysis and Recognition*, pp. 406-409, Sep. 1999.

[7]   A. Spitz, "Determination of the script and language content of document images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.19 (3), pp. 235-245, 1997.

[8]   T. N. Tan, "Rotation invariant texture features and their use in automatic script identification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20 (7), pp. 751-756, Jul. 1998.

[9]   J. Hochberg, K. Bowers, M. Cannon and P. Kelly, "Script and language identification for handwritten document images", *International Journal on Document Analysis and Recognition*, vol. 2, pp. 45-52, Feb. 1999.

[10]   S. H. Srinivasan, K. R. Ramakrishnan and S. Budhlakoti, "Character decomposition", *Indian Conference on Vision Graphics and Image Processing* at ISRO Ahmedabad, 2002.

[11]   V. Bansal and R. M. K. Sinha, "On how to describe shapes of Devanagari characters and use them for recognition", *International Conference on Document Analysis and Recognition*, pp. 410-413, Sep. 1999.

[12]   S. Antani and L. Agnihotri, "Gujarati character recognition", *International Conference on Document Analysis and Recognition*, pp. 418-421, Sep. 1999.

[13]   A. M. Namboodiri and A. K. Jain, "Online handwritten script recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26 (1), pp. 124-130, Jan. 2004.

[14]   A. Busch, W. W. Boles and S. Sridharan, "Texture for script identification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27 (11), pp. 1720-1732, Nov. 2005.

[15]   B. V. Dhandhra, P. Nagabhushan, M. Hangarge, R. Hegadi and V. S. Malemath, "Script identification based on morphological reconstruction in document images", *International Conference on Pattern Recognition*, vol. 2, pp. 950-953, 2006.

[16]   S. Sural and P. K. Das, "Recognition of an Indian script using MLP and Fuzzy features", *International Journal on Document Analysis and Recognition,* pp. 1120-1124, 2001.

[17]   T. K. Bhowmik, P. Ghanty, A. Roy and S. K. Parui, "SVM based hierarchical architectures for handwritten Bangla character recognition", *International Journal on Document Analysis and Recognition*, vol. 12(2), pp. 97-108, July, 2009.

[18]   D. Lopresti, S. Roy, K. Schulz and L. V. Subramaniam, "Special issue on noisy text analytics", *International Journal on Document Analysis and Recognition*, vol. 12(3), Sept. 2009.

[19]   R. Kapoor, D. Bagai and T. S. Kamal, "A new technique for skew detection", *Pattern Recognition Letters, Elsevier Science Direct*, vol. 25(11), pp. 1215-1229, 2004.

[20] I. O. Kyrgyzov, H. Maitre and M. Campedel, "Kernel mdl to determine the number of clusters", *International Conference on Machine Learning and Data Mining,* Jul. 2007.

[21] J. M. Combes, A. Grossman and P. Tchamitchan, "Wavelets, time-frequency methods and phase space", Springer-Verlag, 1989.

[22] A. Grossman and R. Kronland-Martinet, "Time and scale representation obtained through

[23] continuous wavelet transform", Signal Processing IV, Theories and Applications, vol. Elsevier Science Pub. B. V. 1988, pp. 475-482.

[24] S. G. Mallat, "A theory for multiresolution signal decomposition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11 (7), Jul. 1989.

[25] M. Frisch and H. Messer, "The use of the wavelet transform in the detection of an unknown transient signal", *IEEE Transaction on Information Theory*, *Special Issue on Wavelet Transforms and Multiresolution Signal Analysis*, vol. 38(2), pp. 892-897, Mar. 1992.

[26] M. Frisch and H. Messer, "Detection of a transient signal of unknown scaling and arrival time using the discrete wavelet transform", *International Conference on Acoustics, Speech and Signal Processing,* vol. 2, pp. 1313-1316, Apr. 1991.

Analysis of Efficient Wavelet Based Volumetric Image Compression

Krishna Kumar
Department of ECE,   *krishnanitald@gmail.com*
Motilal Nehru NIT
Allahabad, India

Basant Kumar   *singhbasant@mnnit.ac.in*
Department of ECE,
Motilal Nehru NIT
Allahabad, India

Rachna Shah
Department of CSE,   *rachna.shah27@gmail.com*
NIT Kurukshetra, India

Abstract

Recently, the wavelet transform has emerged as a cutting edge technology, within the field of image compression research. Telemedicine, among other things, involves storage and transmission of medical images, popularly known as Teleradiology. Due to constraints on bandwidth and storage capacity, a medical image may be needed to be compressed before transmission/storage. This paper is focused on selecting the most appropriate wavelet transform for a given type of medical image compression. In this paper we have analyzed the behavior of different type of wavelet transforms with different type of medical images and identified the most appropriate wavelet transform that can perform optimum compression for a given type of medical imaging. To analyze the performance of the wavelet transform with the medical images at constant PSNR, we calculated SSIM and their respective percentage compression.

Keywords: JPEG, CT, US, MRI, ECG, Wavelet Transforms, Medical Image Compression

1. INTRODUCTION

With the steady growth of computer power, rapidly declining cost of storage and ever-increasing access to the Internet, digital acquisition of medical images has become increasingly popular in recent years. A digital image is preferable to analog formats because of its convenient sharing and distribution properties. This trend has motivated research in imaging informatics [1], which was nearly ignored by traditional computer-based medical record systems because of the large amount of data required to represent images and the difficulty of automatically analyzing images. Besides traditional X-rays and Mammography, newer image modalities such as Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) can produce up to several hundred slices per patient scan. Each year, a typical hospital can produce several terabytes of digital and digitized medical images.

2. IMAGE COMPRESSION

Both JPEG and wavelet belong to the general class of "transformed based lossy compression techniques." These techniques involved three steps: transformation, quantization, and encoding. Transformation is a lossless step in which image is transformed from the grayscale values in the special domain to coefficients in some other domain. No loss of information occurs in the transformation step. Quantization is the step in which loss of information occurs. It attempts to preserve the more important coefficients, while less important coefficients are roughly approximated, often as zero. Finally, these quantized coefficients are encoded. This is also a lossless step in which the quantized coefficients are compactly represented for efficient storage or transmission of the image [20].

2.1 JPEG Compression

The JPEG specification defines a minimal subset of the standard called baseline JPEG, which all JPEG-aware applications are required to support. This baseline uses an encoding scheme based on the Discrete Cosine Transform (DCT) to achieve compression. DCT is a generic name for a class of

operations identified and published some years ago. DCT-based algorithms have since made their way into various compression methods. DCT-based encoding algorithms are always lossy by nature.
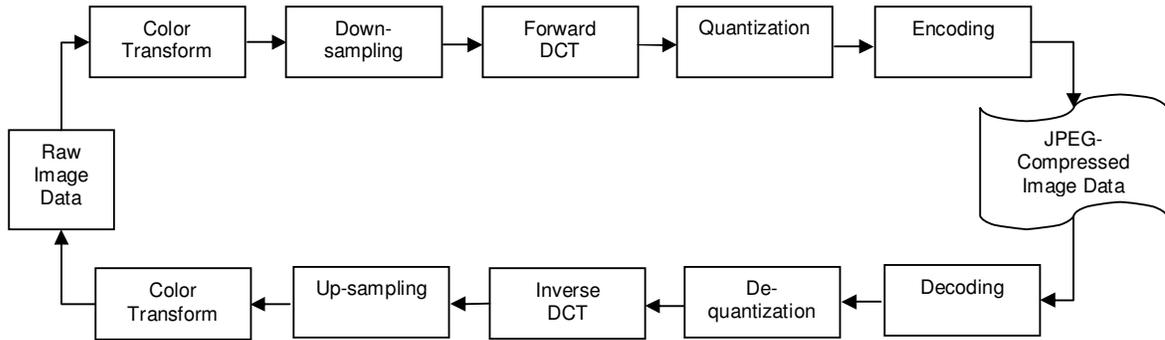
FIGURE 2.1: JPEG Compression & Decompression

2.2 Wavelet Compression

The Fourier transform is a useful tool to analyze the frequency components of the signal. However, if we take the Fourier transform over the whole time axis, we cannot tell at what instant a particular frequency rises. Short-time Fourier transform (STFT) uses a sliding window to find spectrogram, which gives the information of both time and frequency. But still another problem exists: The length of window limits the resolution in frequency. Wavelet Transform seems to be a solution to the problem above. Wavelet transforms are based on small wavelets with limited duration. The translated-version wavelets locate where we concern. Whereas the scaled version wavelets allow us to analyze the signal in different scale. It is a transform that provides the time -frequency representation simultaneously.

2.3 Decomposition Process

The image is high and low-pass filtered along the rows. The results of each filter are down- sampled by two. Each of the sub-signals is then again high and low-pass filtered, but now along the column data and the results is again down-sampled by two.
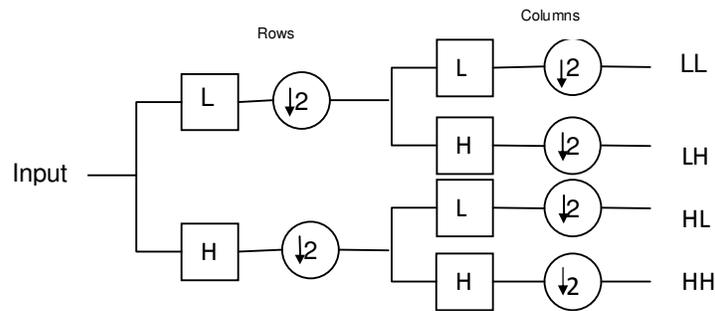
FIGURE 2.3.1: One Decomposition Step of the Two Dimensional Images

Hence, the original data is split into four sub-images each of size N/2 by N/2 and contains information from different frequency components. Fig. 2.3.2 shows the block wise representation of decomposition step.
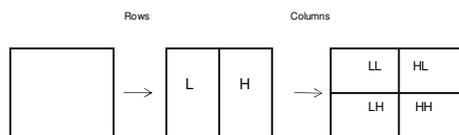
FIGURE 2.3.2: One DWT Decomposition Step

The LL subband contains a rough description of the image and hence called the approximation subband. The HH Subband contains the high-frequency components along the diagonals. The HL

and LH images result from low-pass filtering in one direction and high-pass filtering in the other direction. LH contains mostly the vertical detail information, which corresponds to horizontal edges. HL represents the horizontal detail information from the vertical edges. The subbands HL, LH and HH are called the detail subbands since they add the high-frequency detail to the approximation image.

2.4 Composition Process
Fig. 2.4 corresponds to the composition process. The four sub-images are up-sampled and then filtered with the corresponding inverse filters along the columns. The result of the last step is added together and we have the original image again, with no information loss.
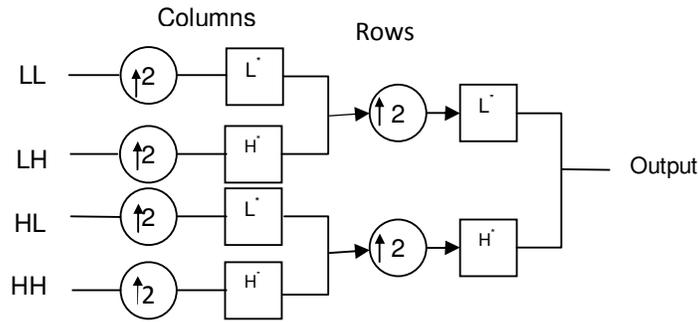
FIGURE 2.4: One Composition Step of the Four Sub Images

3. WAVELET FAMILIES
There are many members in the wavelet family, Haar wavelet is one of the oldest and simplest wavelet.
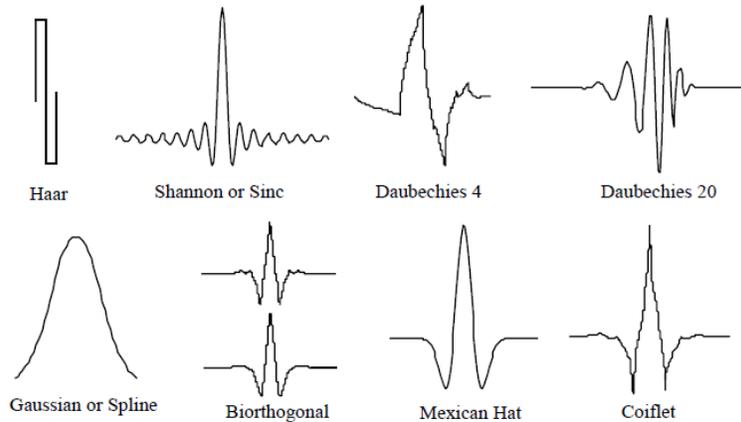
FIGURE 3: Different Types of Wavelets

Daubechies wavelets are the most popular wavelets. They represent the foundations of wavelet signal processing and are used in numerous applications.The Haar, Daubechies, Symlets and Coiflets are compactly supported orthogonal wavelets. These wavelets along with Meyer wavelets are capable of perfect reconstruction. The Meyer, Morlet and Mexican Hat wavelets are symmetric in shape. The wavelets are chosen based on their shape and their ability to analyze the signal in a particular application. Biorthogonal wavelet exhibits the property of linear phase, which is needed for signal and image reconstruction. By using two wavelets, one for decomposition (on the left side) and the other for reconstruction (on the right side) instead of the same single one, interesting properties are derived.

4. MEDICAL IMAGES
Computed tomography (CT) , is a medical imaging procedure that uses x-rays to show cross-sectional images of the body. A CT imaging system produces cross-sectional images or "slices" of areas of the body, like the slices in a loaf of bread. These cross-sectional images are used for a variety of

diagnostic and therapeutic purposes. Magnetic resonance imaging (MRI) is an imaging technique used primarily in medical settings to produce high quality images of the inside of the human body. ECG (electrocardiogram) is a test that measures the electrical activity of the heart. The heart is a muscular organ that beats in rhythm to pump the blood through the body. The signals that make the heart's muscle fibres contract come from the sinoatrial node, which is the natural pacemaker of the heart. In an ECG test, the electrical impulses made while the heart is beating are recorded and usually shown on a piece of paper. Mammography can be used for diagnosis or for screening asymptomatic patients. Mammography is a highly effective imaging method for detecting, diagnosing, and managing a variety of breast diseases, especially cancer. It is an application where an emphasis on patient dose management and risk reduction is required. This is because of a combination of two factors. First, breast tissue has a relatively high sensitivity to any adverse effects of radiation, and second, mammography requires a higher exposure than other radiographic procedures to produce the required image quality. Retinal (eye fundus) images are widely used for diagnostic purposes by ophthalmologists. The normal features of eye fundus images include the optic disc, fovea and blood vessels. Ultrasound imaging is a common diagnostic medical procedure that uses high-frequency sound waves to produce dynamic images (sonograms) of organs, tissues, or blood flow inside the body.

5. FIDELITY CRITERIA

It is natural to raise the question of how much an image can be compressed and still preserve sufficient information for a given clinical application. This section discusses some parameters used to measure the trade-off between image quality and compression ratio. Compression ratio is defined as the nominal bit depth of the original image in bits per pixel (bpp) divided by the bpp necessary to store the compressed image. For each compressed and reconstructed image, an error image was calculated. From the error data, maximum absolute error (MAE), mean square error (MSE), root mean square error (RMSE), signal to noise ratio (SNR), and peak signal to noise ratio (PSNR) were calculated.

The maximum absolute error (MAE) is calculated as [21]

$$\text{MSE} = \max \left| f(x,y) - f(x,y)^* \right| \tag{5.1}$$

Where $f(x,y)$ is the original image data and $f^*(x,y)$ is the compressed image value. The formulae for calculating image matrices are:

$$\text{MSE} = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} \left[f(x,y) - f(x,y)^* \right] \tag{5.2}$$

$$\text{RMSE} = \sqrt{\text{MSE}} \tag{5.3}$$

$$\text{SNR} = 10 \log \left\{ \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} f(x,y)^2}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} \left[f(x,y) - f(x,y)^* \right]} \right\} \tag{5.4}$$

$$\text{PSNR} = 20 \log \left(\frac{255^2}{\text{RMSE}} \right) \tag{5.5}$$

Structural Similarity Index Measurement (SSIM):

Let $x, y \in R^n$, where $n > 2$. We define the following empirical quantities: the sample mean

$$\mu_x \qquad \left(\frac{1}{n} \right) \sum_{i=0}^{n-1} x_i \tag{5.6}$$

The sample variance

$$\sigma_x^2 \qquad \left(\frac{1}{n-1} \right) (x-\mu_x)^T (x-\mu_x) = \left(\frac{x^T x}{(n-1)} \right) - \left(\frac{n\mu_x^2}{(n-1)} \right) \tag{5.7}$$

and the sample cross-variance

$$\sigma_{xy} = \sigma_{yx} \quad \left(\frac{1}{n-1}\right)(x - \mu_x)^T (y - \mu_x) = \left(\frac{x^T y}{n-1}\right) - \left(\frac{n\mu_x \mu_y}{n-1}\right) \qquad (5.8)$$

We define μ_x and σ_y^2 similarly. The SSIM index is defined as,

$$SSIM(x,y) \quad \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \qquad (5.9)$$

Where $c_i > 0, i=1, 2$. The SSIM index ranges between -1 and 1, where positive values closed to 1 indicates a small perceptual distortion. We can define a distortion "measure" as one minus the SSIM index, that is,

$$d(x,y) \quad 1 - \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \qquad (5.10)$$

which ranges between 0 and 2 where a value closed to 0 indicates a small distortion. The SSIM index is locally applied to $N \times N$ blocks of the image. Then, all block indexes are averaged to yield the SSIM index of the entire image. We treat each block as an n-dimensional vector where $n = N^2$

$$\text{Compression ratio}, CR = \frac{\text{number of coded bits}}{n \times m} \qquad \text{where, n, m is the image size.}$$

$$\text{percentage compression} = \frac{\text{size of original image - size of compressed image}}{\text{size of original image}} \times 100 \quad (5.11)$$

6. PROPOSED METHOD

In this proposed method we have analyzed the different medical images with different wavelet transforms at constant PSNR and computed the percentage compression and SSIM.
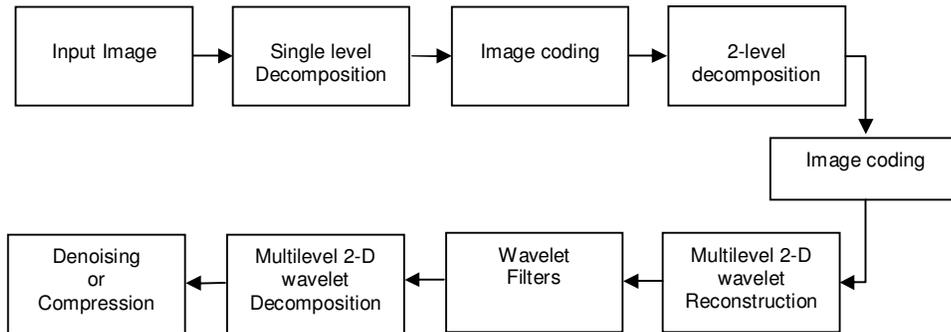
FIGURE 6: Proposed Algorithm

7. SIMULATION & RESULTS
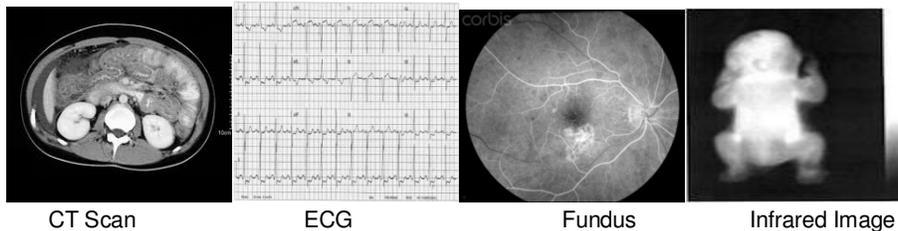
CT Scan   ECG   Fundus   Infrared Image
FIGURE 7.1.1: Original images

| Mammography | MRI | US Image | X-Ray |
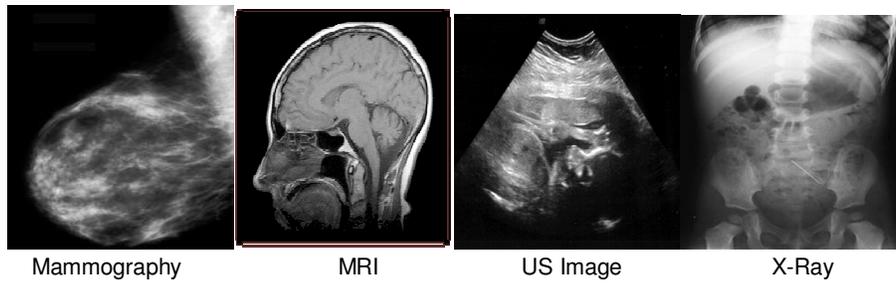
FIGURE 7.1.2: Original images

FIGURE 7.2: Compressed Images after Haar Transform at 2-Level Decomposition

FIGURE 7.3: Compressed Images after Daubechies Transform at 2-Level Decomposition

FIGURE 7.4.1: Compressed Images after Coiflets Transform at 2-Level Decomposition
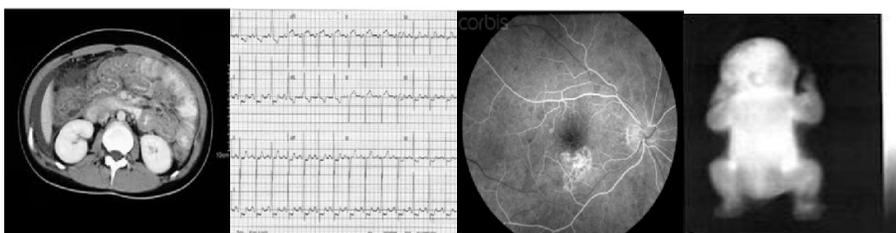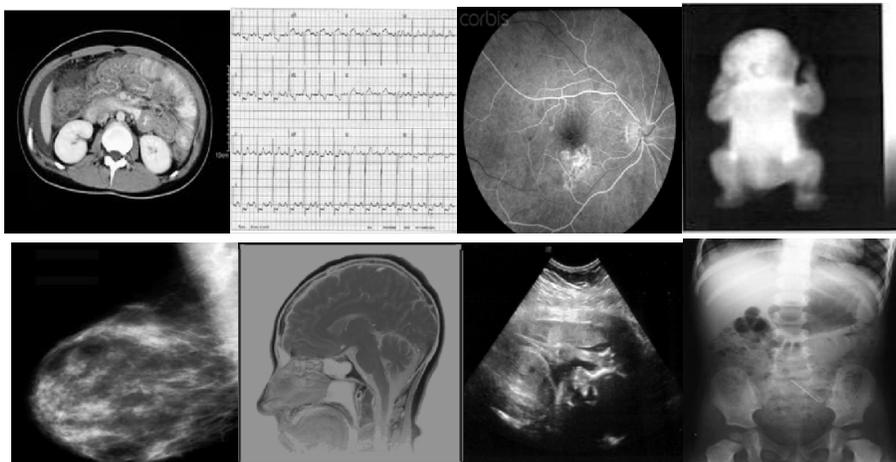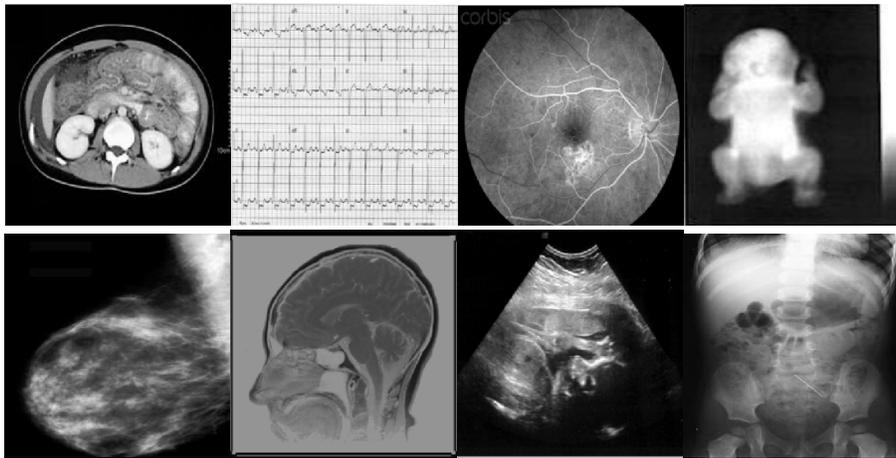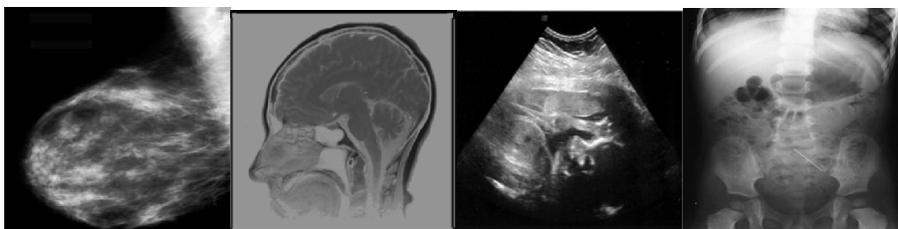
FIGURE 7.4.2: Compressed Images after Coiflets Transform at 2-Level Decomposition
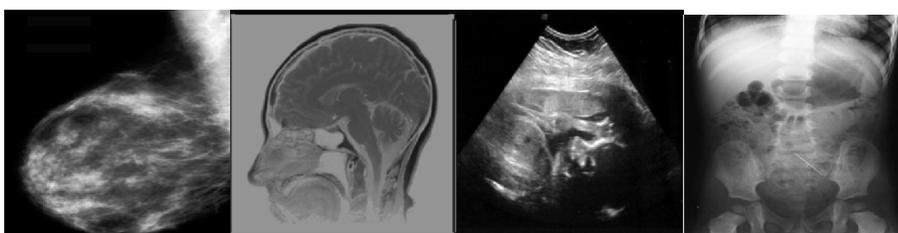
FIGURE 7.5: Compressed Images after Biorthogonal Transform at 2-Level Decomposition

Images	Wavelet Transforms			
	HAAR	**Daubechies**	**Biorthogonal**	**Coiflets**
CT	67.5415	75.4188	78.1819	**80.3231**
MRI	77.1469	**79.6038**	76.7343	74.3275
ECG	**44.4733**	41.0012	31.3784	30.6351
Infrared	84.2682	**87.0825**	85.7940	85.5303
Mammography	75.9598	84.5384	86.0533	**86.2369**
Fundus	62.4176	**69.2187**	68.5846	67.1999
Ultra Sound	71.2311	78.5077	79.2452	**79.4678**
X-Ray	78.4210	86.1492	**87.0921**	86.0198

TABLE 7.1: Percentage Compression for Different Medical Images with Wavelet Transforms

Krishna Kumar, Basant Kumar & Rachna Shah
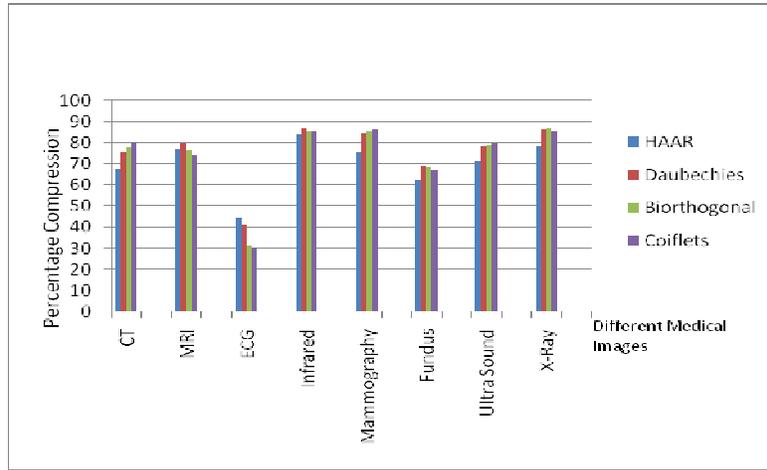
FIGURE 7.6: Percentage Compression for Different Medical Images with Wavelet Transforms
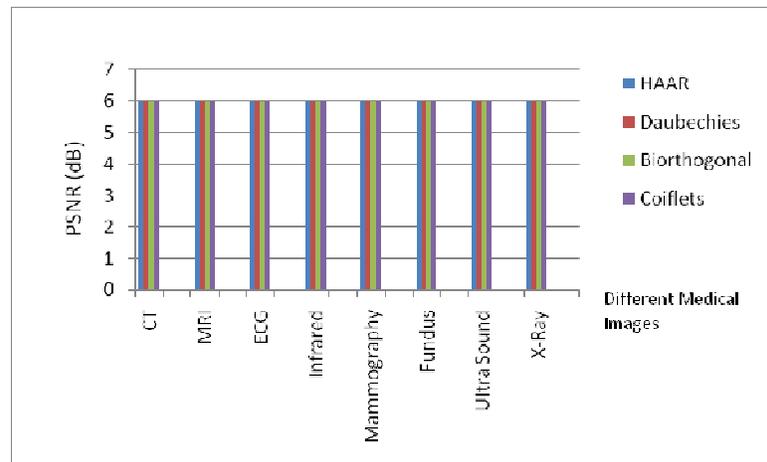
FIGURE 7.7: PSNR (dB) for Different Medical Images with Wavelet Transforms
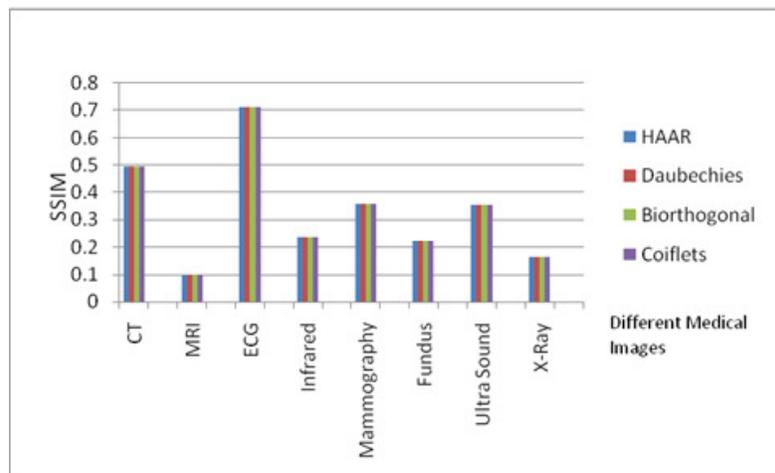
FIGURE 7.8: SSIM for Different Medical Images with Wavelet Transforms

8. CONCLUSION
In this paper we have analyzed that the Coiflets transform gives a higher percentage of compression for CT, US and Mammography images, Daubechies transform gives a higher percentage of compression for MRI, Fundus and Infrared images, Haar transform gives a higher percentage of compression for ECG images and Biorthogonal transform gives a higher percentage of compression for X-ray images at constant PSNR.

REFERENCES
[1]   Short liffe EH, Perreault LE, editors. "Medical Informatics: Computer Applications in Health Care and Biomedicine". New York: Springer, 2001.

[2]   Unser, M. and Aldroubi, A., "A review of wavelets in biomedical applications". Proc. IEEE, No. 5 1996.

[3]   Cosman, P. C., Gray, R. M., and Vetterlui, M., "Vector quantization of image subbands: A survey," IEEE Trans. Image Process. 5, No. 2, 1996.

[4]   Andrew, R. K., Stewart, B., Langer, S., and Stegbauer, K. C., "Wavelet Compression of Ultrasound Video Streams for Teleradiology". IEEE Press, New York, pp. 15–19, 1998.

[5]   Munteanu, B. A., Cristea, P., and Alexopoulos, D., "A New Quantization Algorithm for a Wavelet Compression Scheme of Coronary Angiograms". IEEE Press, New York, pp. 569–572, 1996.

[6]   Vlahakis, V. and Kitny, R. T., "Wavelet-Based Inhomogeneous, Near-Lossless Compression of Ultrasound Images of the Heart". IEEE Press, New York, pp. 549–552, 1997.

[7]   A. Said andW. A. Peralman, "An image multiresolution representation for lossless and lossy image compression,"IEEE Trans. on Image Processing 5, pp. 1303-1310, Sept. 1996.

[8]   J. Luo, X.Wang, C.W.Chen, and K. J.Parker, "Volumetric medical image compression with Three- dimensional wavelet transform and octave zerotree coding," Proceedings SPIE,1996.

[9]   A. Zandi, J. D.Allen, E. L.Schwartz, and M. Boliek, "Compression with Reversible Embedded Wavelet",RICOH California Research Center Report, 1997.

[10]   A. Bilgin andM.W.Marcellin, "Efficient lossless coding of medical image volumes using reversible integer wavelet transforms in Image Processing", Proc. of Data Compression Conference , March 1998.

[11]   Z. Xiong, X. Wu, and D. Y.Yun, "Progressive coding of medical volumetric data using 3-D integer wavelet packet transform," in Image Processing, IEEE Workshop on Multimedia Signal Process- ing , pp. 553-558,Dec. 1998.

[12]   M.Vetterli and J.Kovacevic, "Wavelets and Subband Coding", Prentice Hall, Inc, 1995.

[13]   A. Said and W. A. Pearlman, "Reversible image compression via multiresolution representation and predictive coding," in Visual Communications and Image Processing '93, Proc. SPIE, pp.664- 674, Nov.1993.

[14]   Wang, J. and Huwang, H. H., "Medical image compression by using three dimensional wavelet Transformation". IEEE Trans. Medical Imaging 15, No. 4 ,1996.

Krishna Kumar, Basant Kumar & Rachna Shah

[15]   Winger, L. L. and Ventsanopoulism, A. N., "In Birthogonal Modified Coiflet Filter for Image Comp-ression". IEEE Press, New York, pp. 2681–2684, 1998.

[16]   Manuca, "Medical image compression with set partitioning in hierarchical trees". In Proceedings of the 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Amsterdam, pp. 1224–1225, 1996.

[17]   J. I. Koo, H. S. Lee, and Y. Kim, "Applications of 2-D and 3-D compression algorithms to Ultrasound images," SPIE Image Capture, Formatting, and Display, vol. 1653, pp. 434-439, 1992.

[18]   H. Lee, Y. Kim, A. H. Rowberg, and E. A. Riskin, "Statistical distributions of DCT coefficients and their application to an interframe compression algorithm for 3d medical images," IEEE Trans. Med. Imag., vol. 12, no. 3, pp. 478-485, Sept. 1993.

[19]   K. K. Chan, C. C. Lau, S. L. Lou, A. Hayrapetian, B. K. T. Ho, and H. K. Huang, "Three Dimensional transform compression of images from dynamic studies," SPIE med. Imag. IV: Image Capture and Display, vol. 1232, pp. 322-326, 1990.

[20]   Bradley J. Erickson,Armando Manduca, "Wavelet compression of medical images ," Journal of Radiology, vol. 206, pp. 599-607, 1998.

[21] K Stephen & J.D.Thomson, "Performance analysis of a new semiorthogonal spline wavelet Compression algorithm for medical images,"Med. Phy., vol. 27, pp. 276-288, 2000.

[22]   Brammer MJ. "Multidimensional wavelet analysis of functional magnetic resonance images". Hum Brain Mapp. 6(5-6):378-82, 1998.

[23]   DeVore RA, Jawerth B, Lucier BJ. "Image compression through wavelet transforms coding". IEEE Trans. on Information Theory, 38:719-46, 1992.

[24]   A. Munteanu et al., "Performance evaluation of the wavelet based techniques used in the lossless and lossy compression of medical and preprint images", IRIS, TR-0046, 1997.

[25]   N.J. Jayant, P. Noll, "Digital Coding of Waveforms", Prentice-Hall, Englewood Cliffs, NJ, 1984.

[26]   J.S. Taur, and C.W. Tao, "Medical Image Compression using Principal Component Analysis," International Conference on Image Processing, Volume: 1, pp: 903 -906, 1996.

Performance Analysis and Optimization of Nonlinear Image Restoration Techniques in Spatial Domain

Mr. Anil L. Wanare   *a.wanare@rediffmail.com*
Department of Electronics & TC Engg.
BDCOE, RTMN University, Nagpur, India

Dr. Dilip D. Shah   *dilip.d.shahl@gmail.com*
Principal, G.H. Raisoni College of Engg. & Management
ACM of university of Pune, India

Abstract

This paper is concerned with critical performance analysis of spatial nonlinear restoration techniques for continuous tone images from various fields (Medical images, Natural images, and others images).The performance of the nonlinear restoration methods is provided with possible combination of various additive noises and images from diversified fields. Efficiency of nonlinear restoration techniques according to difference distortion and correlation distortion metrics is computed. Tests performed on monochrome images, with various synthetic and real-life degradations, without and with noise, in single frame scenarios, showed good results, both in subjective terms and in terms of the increase of signal to noise ratio (ISNR) measure. The comparison of the present approach with previous individual methods in terms of mean square error, peak signal-to-noise ratio, and normalized absolute error is also provided. In comparisons with other state of art methods, our approach yields better to optimization, and shows to be applicable to a much wider range of noises. We discuss how experimental results are useful to guide to select the effective combination. Promising performance analyzed through computer simulation and compared to give critical analysis.

Keywords: Nonlinear Image Restoration, Correlation Distortion Metrics, Median With Weight in Spatial Domain, Additive Noise

1. INTRODUCTION

The need for efficient image restoration techniques has grown with the massive production of digital images from various fields often taken in noisy conditions. No matter how good image sensors are, an image restoration is always desirable to extend their various types of transmission media. So it is still exigent problem for researchers. Digital image is generally encoded as a matrix of gray level in continuous range called continuous tone image (CTI). The two main limitation in image accuracy are categorized as blur and noise, blur is intrinsic to image acquisition system [1] and second main image perturbation is the different type of noises. Nonlinear restoration techniques deal with those images that have been recorded in the presence of one or more sources of degradation. Spatial domain is based conditionally on the values of the picture element in neighborhood under consideration and employ a low pass filtering on groups of picture elements the higher region of frequency spectrum [2]. Variety of nonlinear relaxed median and considering weight rank selection have been implemented in MATLAB 7.2.0 to see the suitable combination according to the noise and nonlinear restoration technique, as well as to find efficiency of nonlinear filtering by using various quality metrics. The performance of an image nonlinear restoration method depends on its ability to detect the presence of noisy picture element in the digital image. An interesting method for restoring of single type of image was proposed in[2] [3]. This method appears not to pose any strong restrictions on the degradation. In the cited paper, several experimental results on synthetic noises are shown, but little information is

provided about them. From the information that is given, it appears that the degradation techniques that were used in the experiments either circularly symmetric or corresponded to straight line motion blurs. There seems to be no reason for the method not to be able to successfully deal with other kinds of noises, however the PSF, Gaussian, Speckle, Salt & pepper, and Poission noises are shown in this paper appear to have different density. The improvements in increase signal to noise ratio (ISNR) seem to be between particular ranges in dB for

specific standard deviation. The experimental result presented in section V show that, with much at particular density, our scheme at normally yielded to optimum selection according to larger improvement in nonlinear restoration methods.

In all cases, one has access to more than one degraded image from the same noise a fact which can be used to the ill-posedness of the problem. The scheme also assumed that the quality of the original image, before the degraded, as happens in most images from various fields.To the author's acknowledge, this is first scheme to be proposed, which is able to yield result of optimum selection in such a wide range of situations. The performance and robustness of the nonlinear restoration techniques were tested in various experiments, with synthetic and real life degradation without and with density of noises on the restoring filters, using monochrome images, and under the single frame. The quality of the results was evaluated both visually and in terms of ISNR, normalised mean square error (NMSE), Structural content (SC), absolute difference (AD),maximum difference (MD), normalised cross correlation i.e. correlation quality metric. Detailed comparisons of median, weighted median filtering methods (nonlinear restoration) with MSE, NAE, NMSE, PSNR, AD, MD, difference distortion metrics were evaluated, and show that the proposed scheme yields significantly better for optimum selection of restoration technique.

The remainder of the paper is organised as follows: Section II provides a background review on the concepts involved in nonlinear restoration and continuous tone image (CTI), discrete tone image (DTI), modelling different types of noise in brief. The median filter and weighted median filter implementation in section III. In section IV, results obtain from the nonlinear restoration techniques (distortion metrics, correlation metrics and histograms according to density of noise). In section V; we show comparative results according to combination of CTI, bad CTI and noise to non linear restoration technique. State of art and concluding remarks and future research directions are made in section VI.

2. BACKGRONUD REVIEW

In this section, we briefly review previous work on image restoration, continuous tone image model and noise modeling. Image restoration techniques differ in the choice of image prior model, and many noise estimation techniques assume synthetic degradations.

2.1 Continuous Tone Image and Noise Models

Image sequence $I(r,c,t)$, where t denotes time and (r,c) denotes a spatial location in image domain D. Let (r, c) denote displacement vector of point $(r,c) \in D$ from time t to$(t+1)$, that is $I(r,c,t) = I(r+p,c+q,t+1)$, where image intensity at an object point is assumed to be constant along its motion trajectory over time. For notational simplicity, it is noted that in above equation, the motion components p and q are simplicity function of both (r,c) and 't' [4].Contaminated continuous tone image model considered for analysis as equation, $g(r,c) = \sum_{j=0}^{n-1}\sum_{i=0}^{n-1} f(i,j) h(r-i,c-j) + n(r,c)$, for $r = 0,1,2,....M-1$ and $c = 0,1,2,....N-1$. In matrix vector notation representation is $g = hf + n$. g, f and n are MN dimensional vectors and h is $(MNXMN)$ block circulant matrix represents degradation process embedded in continuous tone image formation process, g vector from the given matrix $\{g(r,c)\}$ is $g(r,c) \in \{0,1,2,....L-1\}$. Many two

tone image processing algorithms are considering, only entire rectangular domain of the continuous tone image [5].

We have considered spatial degradation models for analysis are commonly i) Atmospheric turbulence ii) Motion blur iii) Defocused system. Space-invariant point spread function for mentioned types are described in detail [6][11]. All noises occur in spatial domain modeled as Gaussian, Speckle, Poission and Salt & Pepper. According to application Gaussian

noise is distributed over signal while transmission by $f(g_t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(g-m)^2/2\sigma^2}$, which has bell shaped PDF [8]. Salt & Pepper is an impulse noise, it is generally caused by malfunctioning in picture element in common sensors, faulty memory location or timing problem in quantization process, transmission line is not installed properly [8][9]. Speckle noise occurs in almost all coherent imaging system such as laser acoustic and synthetic

aperture radar imagery. Distribution of speckle noise $f(g_s) = \frac{g^{\alpha-1}}{(\alpha-1)!a^\alpha} e^{\frac{g}{a}}, a^2$ is density of speckle noise and α, g are the picture elements. The Poisson distribution is a one parameter of discrete distribution that takes non-negative integer values. The parameters, both the mean and the variance of the distribution. Thus, as the PDF of Poisson noise is

$f\left(\frac{x}{\gamma}\right) = \frac{\gamma^x}{x!} e^{-\gamma} I_{(0,1,2,\ldots)}(x)$ [10]. CTI an image, Such as a photograph, where gray levels in the image are continuous not discrete and it contains gradient tone, it allows 256 density levels per color [12].

3 SPATIAL NONLINEAR RESTORATION TECHNIQUES

Median filter is nonlinear filter which preserves edges while effectively removing the noise. Median operations are performed by row sorting, Colum sorting and diagonal sorting in matrix [8] [10] [18]. General median filter often exhibit blurring for large window sizes, or insufficient noise suppression for small window sizes.

3.1 Median and With Weights Filter

In previous work, a variety of restoration techniques have been developed in the image processing and computer vision communities. Although seemingly very different, they all share same synthetic noises but only on single type of image. We categorize existing image restoration work in spatial domain by heir single type of image prior and the corresponding representation of type of image statistics.

Objective of the nonlinear restoration technique [9] [10] is to improve the initial continuous tone image means reduce the noise, to increase the contrast, to highlight boundaries and mentioned parameters estimated. Nonlinear filter is robust whose output is not linear to function of input. One of the most commonly used nonlinear filters is the median in spatial domain and major advantage to eliminate the effect of input noise values with extremely large magnitude [15].Median (m) value of a set of numbers is the midpoint value in that set, 'm' is highest likelihood leads to a loss of details; however it is quite distinct from that of 'm'. We are applying 3x3 median filter to all considered noises, which requires an extended length L=15. Spatial median filter (SMF) and vector median filter (VMF) are similar although SMF unbiased [16] so it called smoothing algorithm.

Median with weights which contain positive integer weight expressed for discrete time continuous valued input vector $x = [x_1, x_2, x_3, \ldots, x_N]$, and output Y of SMF of width N with corresponding integer weights are $w = [w_1, w_2, w_3, \ldots, w_N]$, actual response $Y = med[x_1 * w_1, x_2 * w_2, x_3 * w_3 \ldots \ldots x_N * w_N]$. The '$m$' is chosen from the sequence of the products of the sample and their corresponding weights. Positive non-integer weights expressed as weighted median of 'x' is the value of β minimizing the expression $L(\beta) = \sum_{i=1}^{N} w_i |x_i - \beta|$. The picture element in particular window size of an

image are assigned weights according to weight matrix, sorting image vector and center value can be considered as the median [16].

4. NOISE ESTIMATION BY QUALITY MEASURES

Image dependent noise can be estimated from multiple frames or single image. Estimation from multiple images is an overconstrained problems and mentioned [17]. Estimation from single frame, however is an underconstrained problem and further assumption have to be made for degradation. In the image restoration literature degradation often assumed to be an white Gaussian noise. A widely used estimation techniques are based on the mean absolute deviation [18].In [19] the quantitative measures are estimated for each intensity in spatial domain. A.stefano and P. whites system was based on training samples in other domain only for natural images [20]. Generalization expectation maximization restoration techniques in any domain developed and estimate the spectral features.

The scheme for restoration followed by synthetic degradation to give the optimum solution for restoration of degraded images from diversified field. Signal-dependent degradations is calculated from the smooth region of the image by segmenting the image gradient with an adaptive threshold in different domain for specific type of image [21]. By comparing our system provide optimization up to some extend for synthetic and real life degradations in branded camera used. And scheme provides a principled way for selecting nonlinear restoration techniques for estimating the quality of degraded images under various density of noise from diversified field.

4.1 Difference Distortion Metrics

The subjective measures that we used for evaluating the quality of result of restoration tests are mean square error and normalized mean square error, assume that x_0 is an original image, y degraded (with noise) version of that image and x is a restored image, obtain from y .It can be computed by equations (1) .

$$NMSE = \frac{1}{MN}\Sigma_{MN}(x_0 - x)^2 / \Sigma_{MN} x_0^2 \ldots\ldots (1)$$

Maximum Difference, Absolute Difference, Structural Content, difference distortion metrics can be calculated twice one, between original and noisy second between original and restored image as following equations (2) respectively as.

$$AD_{1,2=} \Sigma_{i=0}^{N-1}\Sigma_{j=0}^{M-1}(x_0 - y)/MN \ldots\ldots (2)$$

The performance evaluation of the restored image quality is measured objectively using maximum difference (MD), normalized absolute error (NAE), and normalized mean square error (NMSE). Moreover, observers do the subjective performance assessment. From this experimental evaluation, it can be concluded that objective assessment alone is not suitable objective scale to evaluate the quality of the restored image. Therefore, subjective assessment is very important to take into account visual quality. Normalized cross correlation has been computed in the spatial domain for feature matching of restored image.

4.2 Correlation Distortion Metrics

There are two measures used for evaluating the correlation quality of the result of restoration are Normalized Cross Correlation (ncc) and Normalized Absolute Error (nae). we start by defining the "signal" as image x_0 , y degraded (with noise) version of that image and 'x' is a restored image, obtain from 'y' . It can be computed by equations (3) as

$$ncc = \sum_{MN}(x_0 - y)/\sum_{MN}x_0^2 \dots \dots \dots (3)$$

However, the computation of meaningful increase in signal to noise ratio ($iSNR$) in restoring situation raises some special issues that we now address. We start by recalling basic concept of $iSNR$, the noise of 'y' as $(y - x_0)$, and the noise of 'x' as $(x - x_0)$. $iSNR$ of the restored image 'x' relative to the degraded image y is, then, the difference between the SNR of 'x' and SNR of 'y'. It can be computed, in decibels, as

$$iSNR = 10 \, log_{10}\frac{\sum_i(y^i - x_0^i)^2}{\sum_i(x^i - x_0^i)^2} \dots \dots \dots \dots (4)$$

Where the superscript i indexes the images' pixels, and the sums run through all pixels. The special issues arises in the computation of this measure in restoration situation, are due to the following. The restoration problem is strongly ill-posed. This means that non-regularized solution have a large degradability. There are four kinds of degradation that occurs are mentioned in 2.1. This degradation should be taken into account by the quality measure.

Analysis of restoration techniques based on gray level histogram contains two regions of interest: object and background [12]. As a result of these restoration techniques, the gray level histogram may change according to noise added in the continuous tone image.Most importantly we have learnt about histograms. Histograms tell us how the values of individual pixels in an image are "distributed". An image may have the same histogram as another image. Histogram similarity can be computed as $hc = \sum_{c=0}^{255}\left|f_1(c) - f_1'\right|$, where,$f_1(c)$ is the relative frequency of level c in 256 levels continuous tone image [13]. Contrast enhancement plays a crucial role in image restoration application, such as digital photography, medical image analysis, remote sensing, and scientific visualization. There are several reasons for an image to have poor contrast, addition of synthetic noises, the poor quality of the used imaging device, faulty communication system, lack of expertise of the operator, and the adverse external condition at the time of acquisition. Several contrast analysis (Histogram) technique also have been adapted to restored, degraded, and original image. Other metric is called structural content ($SC_{1,2}$), it can also be computed by standard equation [10]. We have expressed that Autocorrelation of original image, and cross correlation of original x_0 and noisy y image and performing corresponding integration to find out single quantitative measure. It is shown in the next section of the paper.

5. EXPERIMENTAL RESULTS

The main experiment has been performed with on both synthetic and real life degraded gray scale images to test the nonlinear restoration techniques. Each image was degraded with Gaussian, Salt & Pepper, Speckle, Poission noises and PSF. The PSF is nonuniform-intensity circle, and simulates an out-of-focus degradation.

(a)　　　(b)　　　(c)　　　(d)　　　(e)　　　(f)　　　(g)

(h)   **(i)**   **(j)**   **(k)**   **(l)**   **(m)**

(n)   **(o)**   **(p)**
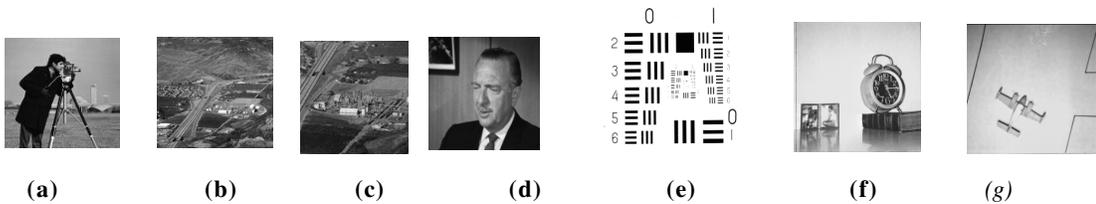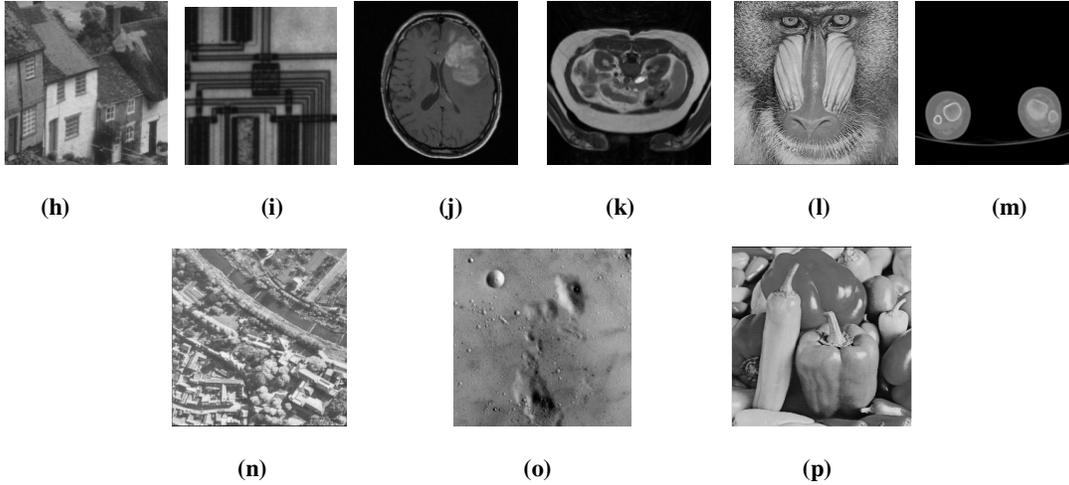
FIGURE 1: Set of typical gray scale images used in experiment (a) "cameraman" (b) "water" (c) "chemical plant" (d) "man" (e) " resolution chart" (f) "clock" (g) "plane". (h) "pic house" (i) "circuit"   (j) "brain" (k) "liver" (l) "baboon" (m) "apperts" (n) "arial" (o) "planet" (p) "pepper" etc.

All images from various fields as Calgary corpus and some natural images from the Berkeley image segmentation database [19] are applied for experimentation. We tested the nonlinear restoration methods on different types of continuous tone images. We also performed comparisons with to each other on same data. Median and weighted median filters implementing in Matlab7.2.0. In the spatial domain, the PSF describes the degree to which an optical system blurs (spreads) a point of light [20] in weighted median filter. In this section, we first describe the experiment, which was intended at showing that effectively deals with a large variety of images and of noises.

Specially goals are: i) to examine the correlation quality between numerical results with ground truth data image (corrupted by additive noise) and results from experiments with original image ii) to quantify the performance in percentage (%) according to level of noise. And sequence of wide variety of noise density 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, and 0.10. Details parameters were computed on Intel core 2 duo system at 2.8GHz, programmed in Matlab for monochrome continuous tone 33 images of size 256x256 as shown in fig.1., about average time 30 seconds after adding degradation. Experiments are performed to observe the effectiveness of the nonlinear restoration techniques. The qualities of the various field images are compared in terms of visual quality and quality correlation parameters.

Table1 and table 2 gives a summary of the result, in terms of improvement in percentage according to MSE and NAE to different four types of synthetic degradation. Improvement is computed by, $|(MSEI - MSEII)/MSEI| X 100$, where MSE-I is error after adding noise to original image and MSE-II is error calculated after restored an image in spatial domain.

For the comparison, we used four types of synthetic degradations. The synthetic degraded images were obtained from the gray scale images with noise which are mentioned in section 2 of this paper. Noises #1, #2, #3, #4 are Gaussian, salt & pepper, speckle, poission respectively and, therefore, is within the family of images for which the median filtering (nonlinear restoration) method is appropriate.

Types of images from various fields	Noises #1,#2, #3, #4 are Gaussian, salt & pepper, speckle, Poission respectively				Types of images from various fields	Noises #1,#2, #3, #4 are Gaussian, salt & pepper, speckle, Poission respectively			
	#1	#2	#3	#4		#1	#2	#3	#4
Camerama	58.17	85.80	52.53	19.46	Planet	74.41	**93.69**	65.51	41.40
Water	59.73	**92.94**	53.60	02.35	house	68.70	**91.20**	57.11	6.96
Chem.Plant	61.13	86.44	44.89	29.28	Pepper	76.77	**97.32**	**67.86**	61.34
Man	**79.78**	**98.23**	**70.22**	**74.71**	Liver	**77.88**	98.88	67.63	59.64
Reso. chart	25.94	73.96	36.66	69.47	Circuit	**77.47**	98.64	68.26	62.66
Clock	66.05	**92.00**	59.63	42.47	Brain	68.81	**95.35**	28.34	36.66
Plane	77.20	**97.42**	66.67	71.05	Baboon	05.59	36.41	01.20	**77.96**
Arial	59.5	86.29	54.73	6.75	Apperts	81.48	**98.70**	67.77	38.64

TABLE 1: Comparison of the result obtained after restored image on median filter. Each entry gives the improvement in % to four types of noise according to MSE for sixteen images, under the indicated conditions. The best performance for each case is shown in bold, Improvement in percentage according to mean square error of median filtering to different four types of synthetic degradation

Types of images from various fields	Noises #1,#2, #3, #4 are Gaussian, salt & pepper, speckle, Poission respectively				Types of images from various fields	Noises #1,#2, #3, #4 are Gaussian, salt & pepper, speckle, Poission respectively			
	#1	#2	#3	#4		#1	#2	#3	#4
Camerama	**43.82**	40.00	41.17	20.28	Planet	**53.33**	16.66	47.05	25.71
Water	47.82	28.57	**86.99**	11.90	house	47.36	**66.67**	41.17	11.84
Chem.Plant	**40.00**	11.11	35.29	11.76	Pepper	56.25	**70.00**	47.05	42.85
Man	59.09	**84.93**	52.94	62.50	Liver	53.33	**75.00**	47.05	44.42
Reso. chart	20.83	**50.00**	25.00	03.70	Circuit	53.84	5.00	47.52	39.08
Clock	47.57	**57.14**	35.71	40.00	Brain	45.00	**46.66**	23.52	02.77
Plane	53.92	**75.71**	50.00	53.57	Baboon	08.48	**55.58**	11.76	43.90
Arial	**39.16**	20.40	37.50	06.06	Apperts	**57.81**	93.90	47.05	38.96

TABLE 2: Comparison of the result obtained after restored image on median filter. Each entry gives the improvement in % to four types of noise according to NAE for sixteen images, under the indicated conditions. The best performance for each case is shown in bold, Improvement in percentage according to normalized absolute error (NAE) of median filtering to different four types of synthetic degradation

The result of the same experiment with Gaussian noise (zero mean with standard deviation varies from (0.01- to 0.10) are shown in fig.3. Similar performance increases by using nonzero values of mean can be observed, with again estimating
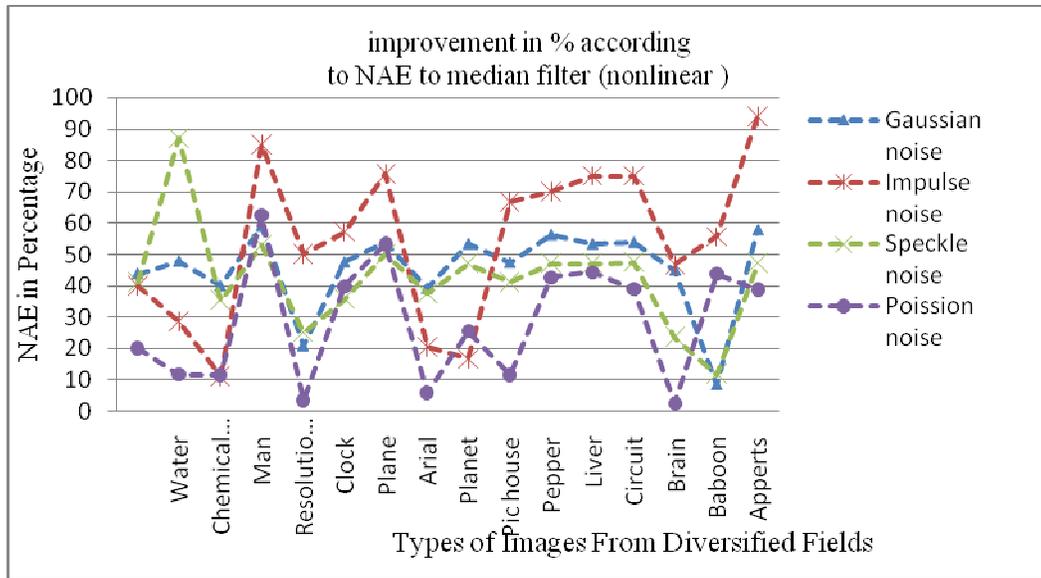
FIGURE 2: Number of images from diversified fields versus improvement in percentage according to normalized absolute error for the different four degradations; Gaussian, salt & pepper, speckle, poission noise
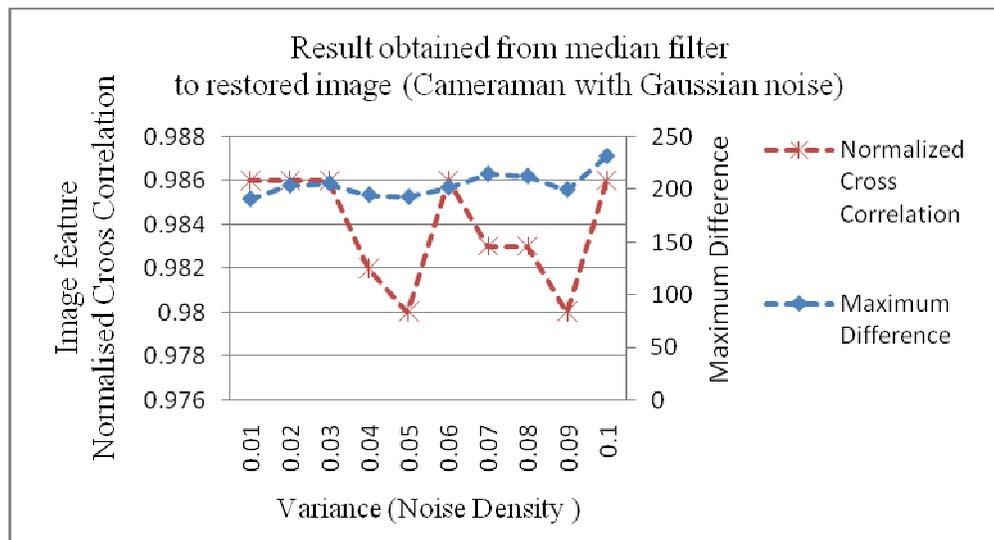
FIGURE 3: Noise density (standard deviation) versus normalized cross correlation and maximum difference of restored cameraman image with Gaussian noise

Weighted Median filter provides the consistence performance to all continuous tone images with Gaussian noise except for medical images which are having more Black back ground like brain and liver. Resolution chart and plane having remarkable result as shown in Table 3. Image of Chemical plant contains same things although this is from natural and Brain from medical field (x-ray), so visually performance of median filter is not up to the mark according, to obtained numerical results to Gaussian noise. It is effective for these four images from object oriented and some natural fields; otherwise we can apply to remaining images, if Gaussian noise is there, or else, we can't apply said restoration technique.

Images from diversified fields	MSEI noisy image /MSEII Restored image	Improvement according to MSE in %	Images from diversified fields	MSEI noisy image /MSEII Restored image	Improvement according to MSE in %
Cameraman	34/22	35.29	Water	30/20	33.36
Chem. Plant	34/24	29.41	Man	34/13	45.83
Resol. Chart	85/55	54.55	Clock	35/17	51.42
Plannet	29/12	58.62	Arial	41/28	31.70
Plane	28/16	42.85	Pic house	31/21	32.25
Pepper	30/23	23.34	Liver	20/11	45.00
Circiut	27/17	37.03	Brain	18/11	38.88
Baboon	43/36	16.27	Apperts	05/03	40.00

TABLE 3: Comparison of Improvement in parentage of diversified field images with MSE I (Degraded by Gaussian noise at noise density = 0.01) and MSEII (Restored images denoised by weighted median filter

Table 4 shows the ISNR values obtained with the synthetic degradation. We can see that, with nonlinear restoration method is clearly providing significant improvement in quality result. Nonlinear restoration method only yielded a significant improvement in image quality for the Gaussian noise with zero mean and standard deviation 0.01, as expected to natural images. When we used mentioned images with salt & pepper noise, which are most appropriate for aerial images. Nonlinear restoration method only surpassed in case of speckle noise to only planet and X-ray medical images. ISNR values attained in the tests described so far in some cases can be considered rather good, taking into account that the restoration methods under nonlinear techniques. In fact, these ISNR values are relatively close to the values attained by state of art restoration methods.

Various Types of images from diversified fields	ISNR values in dB with PSF#5, salt & pepper #6, Gaussian #7, speckle #8, respectively				Various Types of images from diversified fields	ISNR values in dB with PSF#5, salt & pepper #6, Gaussian #7, speckle #8, respectively			
	#5	#6	#7	#8		#5	#6	#7	#8
Cameraman	**16.30**	15.13	15.32	13.73	Planet	9.65	9.96	9.55	**10.08**
Water	**16.74**	16.57	16.60	16.67	house	9.62	12.60	9.72	**12.78**
Chem.Plant	14.86	12.55	12.83	**14.11**	Pepper	**13.08**	11.63	12.85	11.73
Man	32.25	**111.1**	34.44	43.13	Liver	4.99	6.13	**10.93**	5.80
Reso. chart	**8.44**	8.31	4.67	8.34	Circuit	23.7	21.12	5.71	**24.49**
Clock	5.79	**5.84**	5.67	5.52	Brain	1.14	0.93	**25.53**	1.83
Plane	5.47	5.61	**5.79**	5.64	Baboon	**9.80**	9.66	1.33	9.64
Arial	8.65	9.50	**9.55**	9.20	Apperts	36.08	**42.33**	8.89	32.88

TABLE 4 : ISNR values (in dB) obtained for nonlinear restoration method with PSF, Gaussian, Salt & Pepper, and Speckle noises to given sequence of images from diversified fields, the best result are shown in bold

Comparison of result obtained with and without Gaussian noise on the median filter. Each entry gives summation of Autocorrelation and cross correlation of original image with and restored image version of tested images. The correlation between two images is a standard approach to feature detection as well as a component of more sophisticated techniques of original, with degraded image and restored image. Required steps are shown in fig.5.
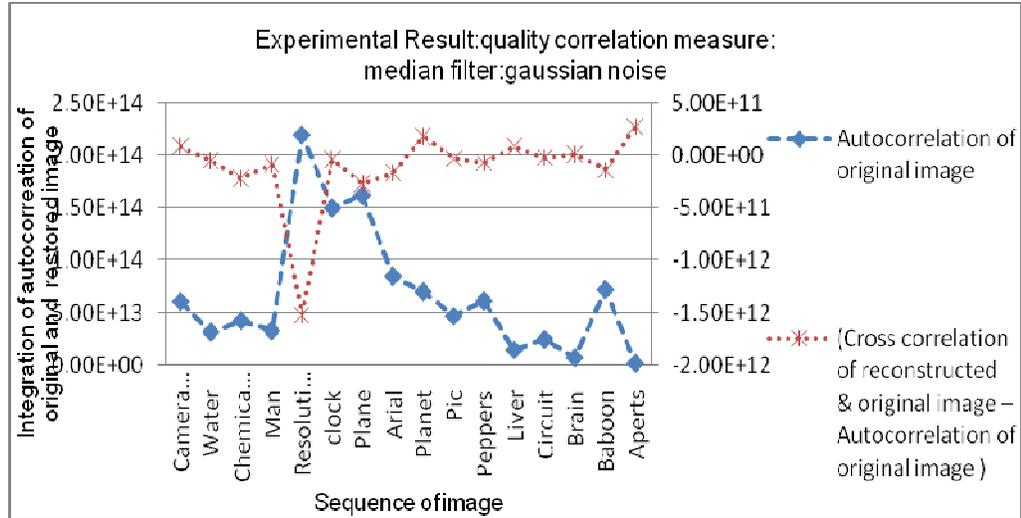
FIGURE 4: Comparison of various images from diversified fields with autocorrelation and cross correlation of the original image, degraded by Gaussian noise image with constant density, and restored image by median filtering technique.
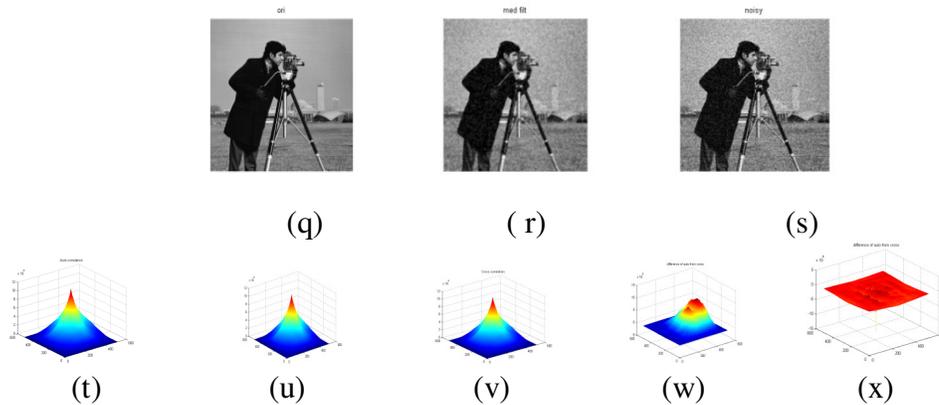
FIGURE 5: Steps of Performing and evaluating the correlation quality of noisy image and Restored image, (q) "original image cameraman" (r) "image degraded by Gaussian noise" (s) "restored image" (t) " Autocorrelation of original image" (u) " Cross correlation of original & noisy image" (v) "Cross correlation of restored & original image" (w) "difference between u and t; (u-t) " (x) "difference between v and t: (v-t)"

Besides testing the proposed scheme on synthetic degradations, we also applied it to real-life degraded photos. We used four different gray scale images shown in fig.6. The corresponding grayscale images were also restored by nonlinear filtering techniques. We addressed two kinds of real-life degradations: the images in fig.6.were purposely taken with the camera wrongly focused in foggy area, while original version in fig.6. The camera was rotated in vertical direction while the images being taken in foggy forest area to produce a particular noise.

(a)

(b)

(c)

(d)
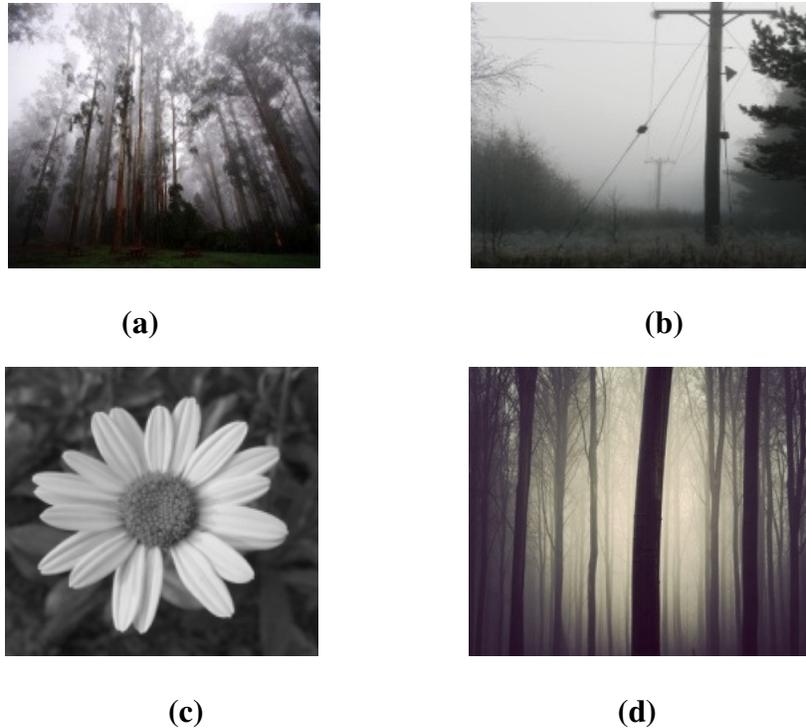
FIGURE 6: Result with an actual degraded images from nonlinear restoration techniques with ISNR in dB (a) "forest ISNR= 4.07" (b) "foggy Telephone pole ISNR= 5.99" (c) "flower ISNR=3.93" (d) "foggy forest ISNR=2.23"

The images were taken with cannon camera, and were coded in JPEG format (this JPEG format convert in to tif) to process. The noise that was present in the image was quite significant. All restored images were significantly sharper and had more visible details than the noisy ones, even though they had some what a "patchy" look, corresponding to some what a piece wise constant character. As had happened with the synthetic degradations, the restoration was slightly better than noisy (real life degraded) images. The results obtained with these images were of lower visual quality than those obtained with the synthetic degradations. Two of the reasons for this probably were as follows. The noise that present in the images probably did not exactly follow the restoration method of; one of the main reason may have been the presence of nonlinearities in the image acquisition. It is known that image sensors may not be perfectly linear, due to the presence of anti-blooming circuits, for example. Furthermore, in the case of the canon camera, for which we did not have access to actual data of image, we suspect that the camera also performed some nonlinear operations like restoration, sharpening, and gamma compensation. The noise produced by charge couple device and climate condition is not synthetic Gaussian and its intensity independent from the image intensity, so that restoration technique not showing the response as to synthetic degradations.

Type of images from diversified fields	PSNR (original and noisy)/(original & restored image)	Normalized cross correlation(NCC) (original & noisy)/(original & restored image)	Absolute Difference(AD) (original & noisy)/(original & restored image)	Maximum Difference (MD) (original & noisy)/(original & restored image)	Structural content ,original / Structural content, restored
Water	20.12/25.31	1.00/.98	0.44/0.4	107/124	0.92/1
Chemical plant	20.06/24.18	0.99/.97	0.009/0.47	112/12	0.94/1.01
Man	20.35/27.29	1.00/0.99	0.97/0.03	98/73	0.94/0.99
Reso.char	22.77/21.76	0.96/0.98	8.05/1.79	108/255	1.07/1.03
Clock	20.57/25.32	0.94/0.99	0.81/0.05	111/197	0.99/1
Plane	20.33/26.75	0.99/0.99	0.35/0.11	120/203	0.98/0.99
Arial	20.13/24.05	0.99/0.98	0.22/0.52	114/130	0.97/1.01
Planet	20.03/25.94	0.99/0.99	0.02/0.34	108/149	0.96/1
Pic	20.07/25.11	1/0.99	0.25/0.06	108/126	0.95/1
Pepper	20.11/26.43	0.99/0.99	0.27/0.22	111/189	0.96/1
Liver	20.96/27.51	1.00/0.99	2.92/0.45	89/64	0.91/0.99
Circuit	20.38/26.85	1.0/0.99	0.89/0.10	97/153	0.92/0.99
Brain	21.25/26.22	1.0/0.96	3.99/0.76	93/142	0.86/1.02
Baboon	20.03/19.80	1.0/0.96	0.15/0.31	101/192	0.96/1.027
Appert	22.34/29.62	1.0/0.99	8.48/3.31	99/62	0.85/0.98

TABLE 5: Showed Differerent Distortion metrics & correlation distortion metrics computed of Original, Degraded and Restored Image: Filter; Median, Noise: Gaussian at Constant Noise Density (Standard Deviation and Mean)

6. DISCUSSION

We stress that, although our numerical results of nonlinear restoration techniques with different noises, some parameters are crucial: MSE, NMSE, and PSNR. In fact experiments have shown that the same sequence of 'σ' and numerical parameters yield considerable results for a wide range of images from diversified fields. This being said, we should note that, by considering quantitative measures, somewhat better optimum selection can be obtained without wasting time to perform experiment separately. In several practical applications, it may be quite possible to select restoration techniques to particular field images. For example, EIA pattern image with Gaussian noise, non linear restoration technique is not suitable to restore the image of EIA pattern.

The choice of restoration techniques for the diversified field images degraded by different types of noise with density is crucial for selection. The choice of restoration technique for specific noise and image has been provided for researchers, it is nothing but the optimization solution up to some extent to select the nonlinear restoration technique.

Weighted median filter with Gaussian operator is providing improved result to some images, we performed experiment with different operators with PSF. Result obtained from the same filter with different operators are less improved than with Gaussian operators, it is shown in table 5 in percentage. We can see that, with nonlinear restoration (weighted median filter) to Gaussian noise clearly surpassed the other combination to same sequence of images from various fields except for baboon image.

7. CONCLUSION

We have presented a critical performance of restoration methods for noisy images from diversified fields. The methods are handled with synthetic degradation to compute the

numerical results. We have computed the quantitative measures using correlation techniques and also adapted MSE, PSNR, NMSE measures to the evaluation of restoration performance of nonlinear methods. According to restoration quality of nonlinear techniques were visually and quantitatively compared and provided optimum solution for the selection to particular image from specific field.

Experimental performance showed on a variety of images, only on gray scale, with a variety of synthetic degradation, without and with noise, generally occurs in real life degradation, and in single image at a still situation. We compared performance of nonlinear restoration methods, median filter is not efficient for resolution chart and baboon (animal face) image if encompass the Gaussian noise, efficient to medical image 'Appert'. When Pepper & Salt noise occurs in animal facing images, median filter is not suitable. We should note that, by finding distortion measure parameters, somewhat better result obtained than once that we have shown in first combination, however efficiency for natural images more than 90% of nonlinear restoration method. Median filtering technique, Performance estimated based on correlation concept used to determine the improvement of nonlinear restoration methods, degraded by synthetic noise.

So far, whenever the degradation image has noise, the particular restoration technique has to be choosing, by selecting the nonlinear restoration technique which yields the best compromise between type of image from particular field and noise detail. An automatic selection criterion will obviously be useful. This is the direction in which further research will be done.

8. ACKNOWLEDGEMENT

9. REFERENCES

[1]   T.Weissman, E.Ordentlich, G.Seroussi, S.Verdi, and M.J.Weinberger, "Universal discrete   denoising: known channel," IEEE trans. Inform. Theory, Jan. 2005.

[2]   T.Weissman, E.Ordentlich, G.Seroussi, S.Verdi, and M.J.Weinberger, "A discrete universal denoiser and its application to binary image," in proc. Of ICIP'03, Barcelona, Sept. 2003.

[3]   A.Buades, B.Coll, and J.M.Morel, "A new of image denoising algorithms, with a new one," SIAM of multiscale modeling and simulation (MMS)

[4]   H.Takeda, S.Farsiu, and P.Milanfar, "Kernel regression for image processing and reconstruction," IEEE Trans. Image Processing., Vol.16, no. 02, pp. 349-366, Feb. 2007.

[5]   Ignacio Ramirez, M. Weinberger, "The dude frame work for continuous tone image denoising, " IEEE Trans.2005.

[6]   R.H.Chan, C.Ho, and M.Nikolova, "salt & pepper noise removal by noising detector and detail persening   regulization," IEEE Transon Image Processing, available at http:/www.math.cunk.edu.hk /impulse.

[7]   L.Yin, et.al. "Weighted median Filter: Atutorial," IEEE Trans on circuit and system II, Vol.43, Issue03, pp.157-192, March 1996.

[8]   A.Buades, B.Cell, and J.Morel, " A non-local algorithm for image denoising," in Proc. IEEE Conf. Computer Vision and Pattern Recognition, Washington, DC, Oct. 2005, Vol. 02, pp. 60-65.

[9]   I.Pitas, A.N.Venetsanopoulos, "Nonlinear filters principle and application," Kluwrer Academic Publishers, Dodresht, 1990.

[10] J. Astola and P. Kuosmanen, Fundamental of Nonlinear Filtering, Boca Raton, CRC Press, 1997.

[11] A.Asensio, A.Lopez Ariste, "Image Reconstruction with Analytical Point Spread Function," Astronomy & Astrophysics Manuscript, pp. 1-10, April 20, 2010.

[12] Y.E. Ioannidis, "The History of Histograms (Abridged)," *Proc. Int'l Conf. Very Large Data Bases,* 2003.

[13] S. Guha, N. Koudas, and K. Shim, "Approximation and Streaming Algorithms for Histogram Construction Problems," *ACM Trans. Database Systems,* vol. 31, no. 1, pp. 396-438, 2006.

[15] T. Loupos, W. N. McDicken and P. L. Allan, "An adaptative weighted median filter for speckle suppression in medical ultrasonic images," IEEE Trans. Circuits Syst., vol. 36, Jan. 1989.

[16]   L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: A tutorial," IEEE Trans. Circuits and Syst. II: Analog and Digital Signal Processing, vol. 43, pp. 157-192, Mar. 1996.

[17] M.D.grassberg and S.K. nayar, "modeling the space response function of camera," IEEE trans., PAMI, vol. 26, Oct. 2004.

[18]   D.Donoho, " denoising by soft thresholding ," IEEE, Trans IT, vol. 3, 1995.

[19]   D.Martin, C. Fowlkes, D. Tal and J. Malik, " A database of natural images and its applications to evaluating segmentation algorithms and measuring ecological statistics", proc. IEEE conf. Computer Vision, Volume 2, pp., 416-423, July. 2001.

[20]   A. stefano, P. white, " training method for image noise level estimation on wavelet componenet," JASP, vol. 16, 2004.

[21] J. portilla, " blind denoising through noise covarience estimation using gaussian scale mixture in wavelet domain," IEEE, conf Image processing, pp1217-1220, 2004.

[17] S-J. Ko and Y. H. Lee, "Center-weighted median filters and their applications to image enhancement," IEEE Trans. Circuits and Syst., vol. 38, pp. 984-993, Sept. 1991.

[18] T. Chen, K.-K. Ma and L. -H. Chen, "Tri-state median filter for image denoising," IEEE Trans. Image Processing, vol. 8, pp. 1834-1838, Dec. 1999.

[19]   E. Abreu, M. Lightstone, S. K. Mitra and K. Arakawa, "A new efficient approach for the removal of impulse noise from highly corrupted images," IEEE Trans. Image Processing, vol. 5, no. 6, pp. 1012-1025, Jun. 1996.

Mr. Anil L. Wanare & Dr.Dilip D. Shah

[20]   E. Abreu, "Signal-dependent rank-ordered mean filters," Nonlinear Image Processing, S. K. Mitra and G. L. Sicuranza (Eds.), Academic Press, 2000.

[21]   Y.L.You and M.Kaveh, "Blind image restoration by anisotropic regularization," IEEE Trans. Image Processing, vol. 8, no. 3, pp. 396-407, Mar. 1999.

Image-Based Multi-Sensor Data Representation and Fusion Via 2D Non-Linear Convolution

Aaron R. Rababaah   *arrababaah@umes.edu*
Math & Computer Science
University of Maryland Eastern Shore
Princess Anne, 21853, USA

Abstract

Sensor data fusion is the process of combining data collected from multi sensors of homogeneous or heterogeneous modalities to perform inferences that may not be possible using a single sensor. This process encompasses several stages to arrive at a sound reliable decision making end result. These stages include: senor-signal preprocessing, sub-object refinement, object refinement, situation refinement, threat refinement and process refinement. Every stage draws from different domains to achieve its requirements and goals. Popular methods for sensor data fusion include: ad-hock and heuristic-based, classical hypothesis-based, Bayesian inference, fuzzy inference, neural networks, etc. in this work, we introduce a new data fusion model that contributes to the area of multi-senor/source data fusion. The new fusion model relies on image processing theory to map stimuli from sensors onto an energy map and uses non-linear convolution to combine the energy responses on the map onto a single fused response map. This response map is then fed into a process of transformations to extract an inference that estimates the output state response as a normalized amplitude level. This new data fusion model is helpful to identify sever events in the monitored environment. An efficiency comparison with similar fuzzy-logic fusion model revealed that our proposed model is superior in time complexity as validated theoretically and experimentally.

Keywords: Multi-senor Data Fusion, Situation Assessment, Image-based Fusion, Data Fusion Via Non-linear Convolution.

1. INTRODUCTION

Data fusion process (DFP) consist of typical stages including: environment sensing, event detection, event characterization, and fusion of events of interest which is the scope of this work. Each of the listed above DFP stages has a spectrum of alternate models and techniques that fit a particular solution. Reported techniques for these stages are discussed as follows: 1) Event Characterization - The objectives of this phase in the DFP are mainly: define a set of events of interest (EOIs) that corresponds to anomalous behaviors, select a set of features that effectively and efficiently characterize these EOIs, and classify the EOIs into designated classes of events such that, each of which has its intensity according to the situation context. The reported techniques for this stage include: context-based reasoning [1, 2], agent collaboration [3], analysis of trajectories [4, 5, 6], centroid-to-contour distance [7, 8], K-means clustering [9], signal amplitude histogram, zero-crossing, linear predictive coding, FFT, Gaussian mixture modeling, and Bayesian inference networks [10-15]. 2) Fusion of Events of Interest (EOIs) - In DFP applications, the current situation is described by a predefined set of EOIs. Furthermore, these EOIs are estimated through their corresponding target/phenomena' evolving states. The focus of this paper is on multi EOI fusion for anomalous situation assessment. Fusion of EOIs in DFP involves combining data/information locally-processed by multiple sources. 3) Situation assessment: this stage is concerned with mapping the output state of the fused EOIs onto a situation scale/map that reflects the current intensity of the monitored process. In addition to the basic motivations of Multi Sensor Data Fusion (MSDF) such as: improving operational performance, extending spatial and temporal coverage, increasing confidence in inferences and decisions making, reducing ambiguity, enhancing EOIs detection, improving system reliability,

and increasing dimensionality [16], the main motivations of EOIs fusion in multi-source systems include:

- Sources partially perceive the environment from different perspectives. Therefore, this motivates considering all perspectives of the different sources as a bio-inspired behavior in situation assessment.
- Sources fields of coverage (FOC) may be non-overlapping, partially overlapping or significantly-to-fully overlapping. This in turn, enhances the redundancy of the system and calls for collaborative-oriented processing.
- The characterization features or states might not be available from all sources over the entire space-time domain. Therefore, MSDF framework is needed to make the best of the given perceptions of the situation.
- Every source has it is own inherent variability and environment-related noises, which effects the quality, reliability, and confidence of the data/information drawn from that source. Therefore the contributions from a particular source should be tagged by these metrics of performance to determine its weight in the inference making across the fusion process.

Reported techniques of EOIs fusion include: Kalman filter was used in [17] to fuse the localization of events from two different modalities, i.e., camera and laser ranger. Fusion model for 3D track estimation from two partially-overlapping cameras was introduced in [18]. Object features were computed independently from each camera then fused for robust attribute estimation and enhanced tracking accuracy. The hypothesis/track association data fusion concept was utilized in [19, 20, 8, 21, 22] to assign detected targets to existing tracks in the historical databases. A fusion model was presented in [23] that is capable of fusing information collected from two modalities: digital camera and IR camera. They used the quality of detected regions by different sensors as a weight factor to tag their contributions. Traffic surveillance system was introduced in [1] utilizing seismic sensors, CCD, infrared, and omni-directional cameras. The system combines all events detected by these multi-modalities and infers a traffic situation. The tasks of the sensor network were managed by event or context dependent scheme in [24]. Based on the situation awareness and understanding, accuracy, high transmission rate, focus of attention can be triggered for efficient utilization of surveillance network resources. A fusion-based solution for target occlusion was introduced by [19, 20, 22]. The concept is to use overlapping FOC sensors for multi-perspective reasoning of the occluded targets, which helps higher processing levels to enhance the situation awareness. Correlation-based tracking and Kalman filter were used as means of object tracking. Multiple hypothesis generation and validation paradigm using Bayesian rule is used for state estimation. Trajectory segments are collected and fused to characterize events [25]. A multi-camera target tracking presented in [26] where, the system was based on the Joint Directors Lab (JDL) fusion model. They implemented a three-level fusion process of report to track association, i.e., signal level, object level, and event level fusion based on features including: kinematics, objects attributes and event dynamics.

The proposed fusion model is based on a human-inspired concept of brain energy mapping model for situation representation, fusion and decision-making. We will call our fusion model IBMS. Human biological senses (sight, hearing, touch, smell and taste) are connected to different regions of the brain as shown in Figure 1 [27]. Humans collect sensory data from the environment via these biological sensors and map this data as energy stimuli onto designated regions of the brain. Every sensory modality has its logical importance factor and consciousness that are acquired and learned throughout the human life. After mapping these energies from different sensing sources, the brain builds a map in the hyper-dimensional space of the current situation and by using its complex inference engine, it maps this situational awareness onto the decision making space of actions. This concept and analogy will be further developed in the following sections.
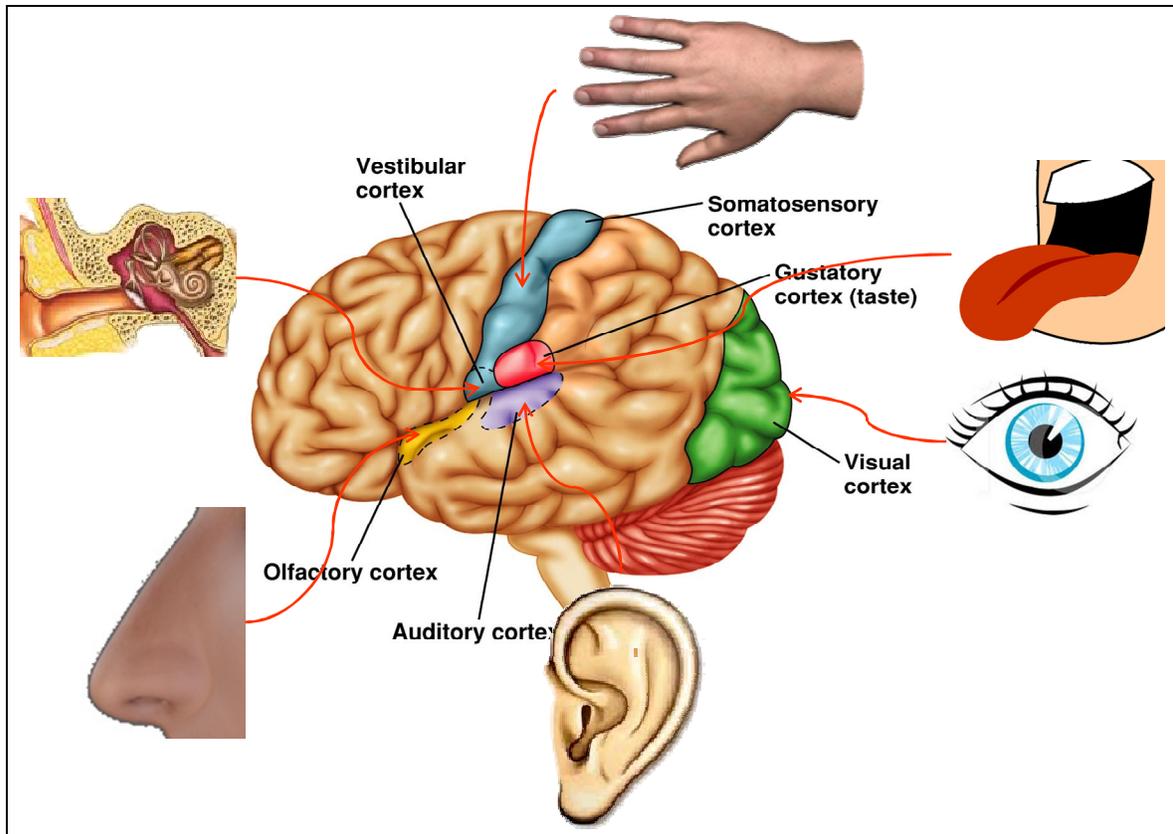
FIGURE 1: Human Brain Sensory Maps. This figure is to show the analogy between the human sensing, characterization and fusion of events from multi sources of information to the proposed fusion model IBMS. The human brain maps sensory data onto designated cortexes before fusing and making sense of it. This concept is adopted in the IBMS model which designates certain volumes to collected data by sensors and then applies image processing theory to extract an useful inferences out of this fused data.

IBMS fusion model is based on feature energy-mapping and image nonlinear convolution for multi-feature fusion. The principal of IBMS is to effectively and efficiently represent event-features by mapping their energy assessment-levels onto a multi-feature energy map. Each feature is projected onto an axis with an exponential energy propagation function that is used to merge/fuse adjacent feature energies. The position of the feature-energy along its axis is determined by the importance factor of the feature compared to the rest of the set of features for a particular sensing modality. The separation angle between axes signifies the dependencies of the adjacent features, where, dependent features are grouped in the same sector. The above mentioned functions, i.e., (feature importance factor, and separation angle, and energy propagation function) represent the key parameters for the IBMS model. These parameters will have to be optimized for any specific application domain. The IBMS draws from the theory of image processing to perform the multi-feature fusion. The IBMS data and information fusion process consists of five main stages: 1) feature modeling, 2) feature mapping, 3) multi-feature homogeneous/heterogeneous fusion, 4) Multi-agent data fusion and 5) output space state estimation. This process is illustrated in Figure 2. In the later sections, all of these stages are presented, discussed, and the mathematical modeling of each stage will be established

2. THEORETICAL DEVELOPMENT

This section will present the theoretical formulation of the fusion process of the IBMS model. For each stage in the fusion process we will present the mathematical models established to ultimately transform the input data (state/feature vector) into situation assessment (output state).
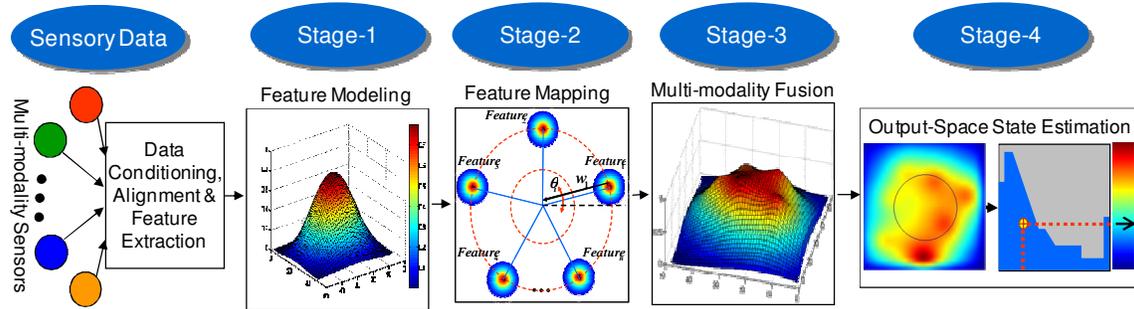
FIGURE 2: IBMS Process with the different stages identified. Each of these stages are described in details in this section. See sections: 2.1 ~ 2.4.

2.1 Feature Modeling

Before this stage, there are two pre-processing stages that take place outside the IBMS model, these are: sensory data collection, where, an array of sensors is deployed in a system sense the required features and relay it to the second pre-processing stage, the data condition, alignment and feature extraction. Where, the data is filtered and aligned (registered, normalized) in terms of space and time, and features of interest are computed. After, these two pre-processing stages, the first stage of the IBMS model is performed. Typical outcome of the characterization of target phenomena of interest are extracted features that represent the signature of the state of that phenomena. Feature modeling transforms the computed features into a systematic intensity scale of [0,1]. Features can be of two types; qualitative and quantitative features. Qualitative features are expressed in natural language with discrete feature intensity scale. An example of qualitative features is human posture. The feature of human posture can be modeled as a discrete scale of postures as: standing, bending, squatting, crawling, etc. These different postures are assigned proportional intensities ranging from 0 to 1 based on different criteria such as: belief, statistical history, expert input, heuristics, etc. On the other hand, quantitative features are continuous attributes that can assume any real value. To achieve the feature alignment among qualitative and quantitative features, the quantitative features are normalized to the range of [0,1]. Having the input of the features of interest defined and categorized as presented above, the next stage is to model each feature using conventional probability density functions (PDFs) in three dimensional space such as: binomial, exponential, geometric, Gaussian, etc.
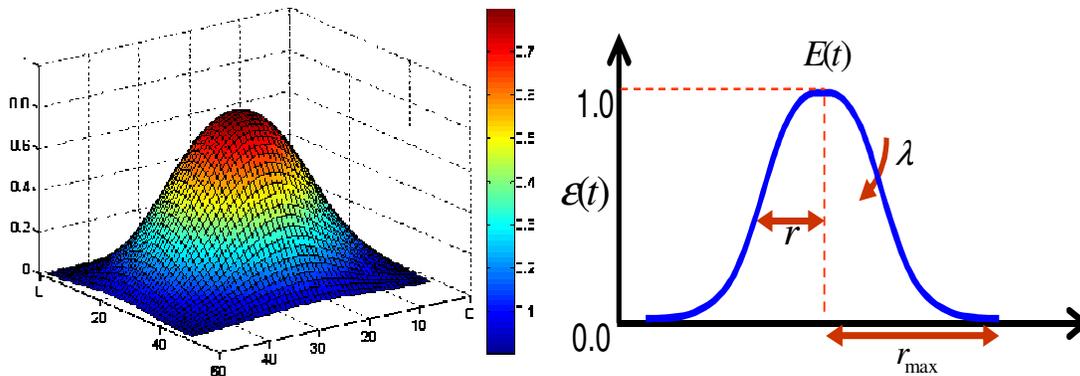
FIGURE 3: Energy Map of a typical feature, Left: 3D version, Right: 2D version

$$\varepsilon(t, x, y) = E(t)e^{-\lambda\sqrt{x^2+y^2}} \qquad (1)$$

Where,

λ = feature fusion factor

t = time

x,y = spatial coordinates of the energy map

ε = propagated feature energy

E = the max feature energy

IBMS models, normalizes, and maps these intensity scales to a 3D exponential energy map with the highest peak of one as shown in Figure 3. The controlling parameters of this energy map are illustrated in Figure 3 (right). Note that Figure 3 (right) is a 2D profile projection of the 3D energy map as shown in Figure 3 (left). This 2D representation provides a better clarity in identifying the different parameters of the map expressed in equation (1). Although, there are other choices for energy mapping functions such as: Gaussian distribution, sigmoid function, polynomial function, etc. the exponential mapping function was used due to the fact that the theory of energy decay follows an exponential decay function such as: mechanical vibration decay, heat dissipation, capacitance electrical potential release, etc. Therefore, it was decided that an exponential modeling of the feature energies would be more appropriate since the proposed IBMS fusion model uses an energy fusion concept.

2.2 Feature Mapping

The modeled feature as PDFs in stage-1 are mapped onto an energy fusion map, where, each PDF is positioned in an orbit around the fusion center. The radius of that orbit is proportional to the importance factor ranging between [0,1]. This importance factor is decided upon by the user according to the context of the application. For example, in security systems, a gunshot signal may have a higher importance factor than crawling human posture video signal. The significance of importance factor is that, it determines the contribution of each feature to the fusion energy surface. This parameter is controlled by sliding the PDF along the radial axis passing between the fusion center and the center of the PDF. The closer the PDF to the center, the more importance it has, and vice versa. The mapped PDFs are grouped via controlling the separation angle between them. In case there exist dependent group(s) of features, they can be clustered by reducing the angle between them according to the inverse of the pair-wise degree of dependency. The elements of the IBMS fusion model are depicted in Figure 4. These elements are explained as follows:

- **Energy Map**: a 2D energy map which hosts the modeling, mapping, and fusion operations of the IBMS model.
- **Mapped Features**: target attributes modeled using equation (1) and mapped onto the energy map. The number of the mapped features determines the dimensionality of the feature space and the order of IBMS model. N is the number of mapped features.
- **Feature Weight** (w_i): the importance factor of a given feature. It varies in [0,1], where 1 signifies the highest importance factor that gives the feature high contribution to the fused energy within the fusion boundary.
- **Feature Axis**: the radial direction at which a feature is positioned on the energy map. This axis controls the variation of the feature weight by sliding the feature position inward (toward the fusion center) or outward (a way from the fusion center)
- **Feature Angle** (θ): the angle between the feature axis and the x-axis of the energy map. This parameter is significantly important in making the multi-agent-fusion seamless. That is because, by enforcing a fixed angle for each different mapped feature makes the feature alignment inherent between agents. Hence, making the aggregation of multiple agent IBMSs transparent and readily achieved.

- **Fusion Boundary**: a circular boundary that represents the spatial threshold for energies to contribute in the IBMS process. Therefore, only, energies enclosed within this boundary will take part in the energy fusion process.
- **Map Center**: the center of the fusion boundary which is also the same as the center of the energy map.

After the parameters of the feature energy function (as shown in Figure 3) have been computed for each target feature, the second step is to map the target features onto the energy map. The schematic representation of the IBMS shown in Figure 4 presents initial state of a five-dimensional energy space, where each dimension represents a feature energy. Each feature energy has an importance factor that varies from 0 to 1, where 1 signifies the highest order of feature energy contribution to overall fused energy map.

To set the importance factor for a particular feature, the energy model of the feature need to be slid radially along its axis. The closer the feature to the center the more influence it would have on the fused energy surface as it will be demonstrated later in this section. Figure 4 illustrates feature-2 that has been configured to have an importance factor $w_i = 0.75$. This means that the radial distance between the fusion center and the center of the energy model of this feature is (1-0.75 = 0.25) of the radius of the energy map.
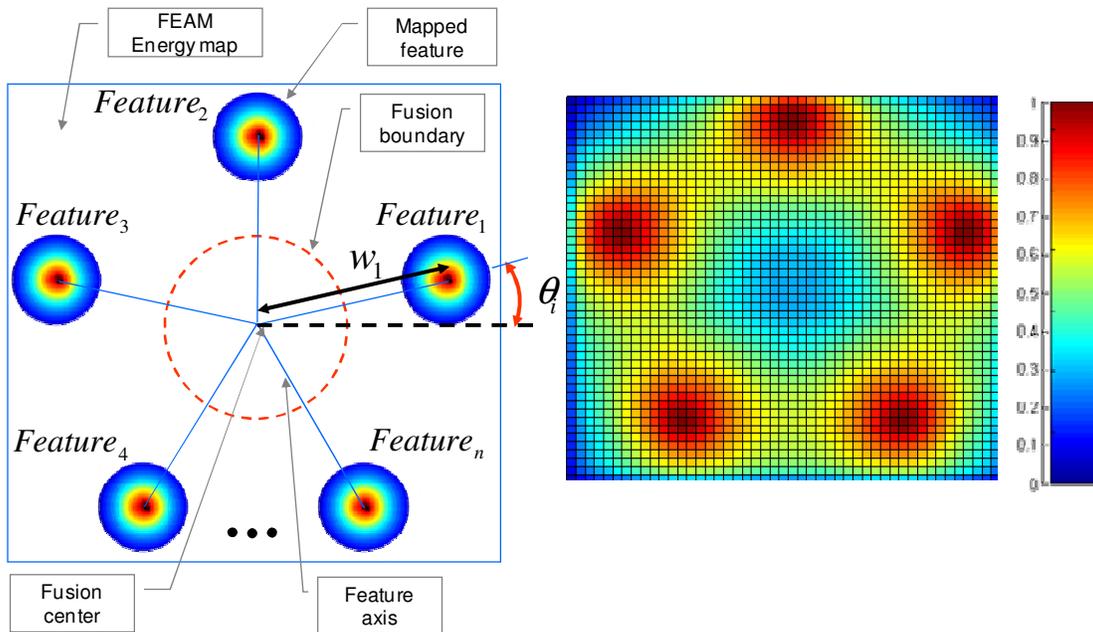
FIGURE 4: Elements of the IBMS Model: Left: schematic diagram of the IBMS energy map. Right: 2D visualization of actual IBMS map

As shown in Figure 5, the new location of the feature gives it more importance and weight to contribute to the fused energy surface within the fusion boundary. In Figures 4 and 5, schematic of energy map was shown for clarity and identifying its elements and parameters. In Figure 4, IBMS map at the initial stage is depicted. Where, five features were located on the map. The map shows at this initial stage there is no or very little interaction among the features as they are located far away from each other and from the fusion center. As the dynamics of the events evolve, the features will exhibit variation in their intensities. These intensities are then propagated using equation (1). An illustration of this operation is demonstrated in Figure 5. After the energies

of the mapped features are propagated, multi-feature fusion is performed which is presented in the next section.

2.3 Multi-Feature Fusion

After modeling, normalizing, and mapping the target features, the next stage in the IBMS fusion process is the multi-feature fusion based on the theory of image non-linear convolution. In the IBMS fusion process, the equivalent function for the non-linear convolution is given in equation (2). Equation (2) addresses the dynamic update of the energies and non-linear convolution of the feature energies, but the energies of the neighboring features need to be aggregated to generate the fused energy 3D surface. This aggregation formula is expressed as:

$$\mathcal{E}^{MF}(t, x, y) = \mathcal{E}(t, x, y) + \sum_{k=1}^{k=N} \mathcal{E}_k(t, x, y) \tag{2}$$

Where,

$\mathcal{E}^{MF}$ = Multi-feature fused energy at time (t) and coordinates (x, y)

$\mathcal{E}$ = Propagated energy from the max feature energy

$\mathcal{E}_k$ = Partial contribution of feature(k)

N = Order of IBMS (dimensionality of the feature hyper space)
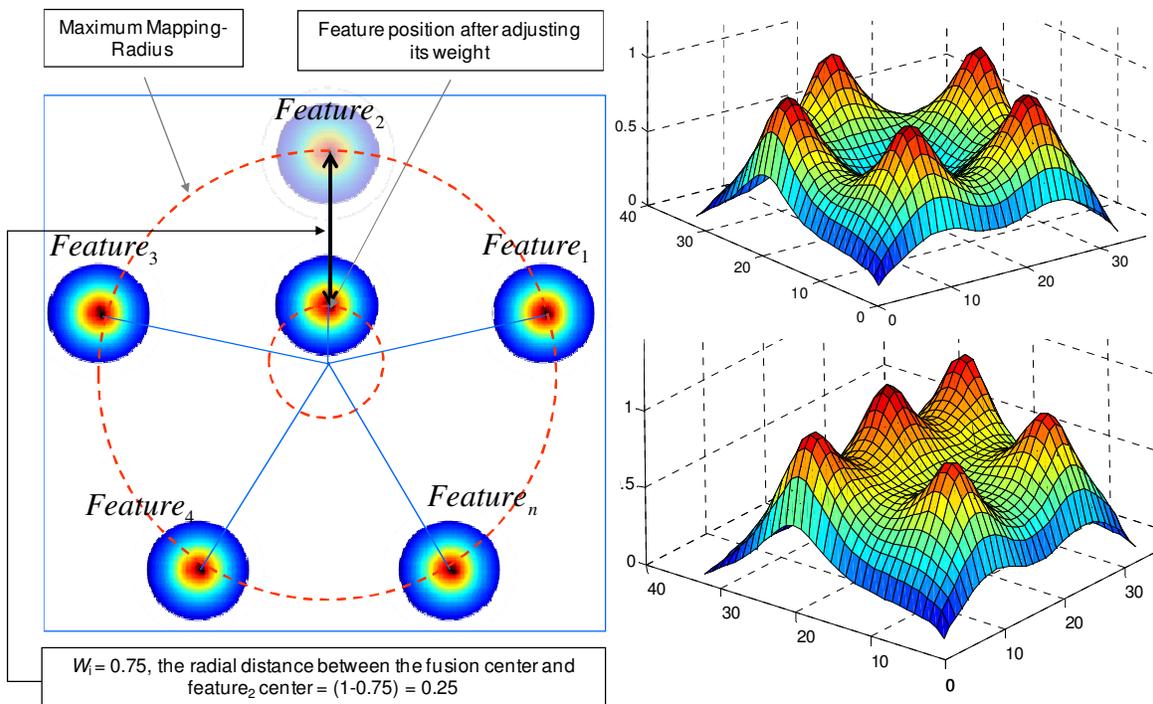
FIGURE 5: A Demonstration of Feature Importance Factor: Setting the weight (importance factor) of feature$_2$ by sliding along its axis toward and away from the fusion center. Left: the 2D schematic of the operation. Top-right: 3D visualization of the map before setting the weight. Bottom-left: 3D visualization of the map after setting the weight.

A typical demonstration of multi-feature fusion at this stage is illustrated in Figure 6. In this example, a five dimensional feature space (N=5) is considered. The feature normalized weights (importance factors) are given as {0.4,0.1,0.2,0.05,0.25}, and their current energies are assumed as {0.79,0.43,0.49,0.31,0.31} for illustration purposes.
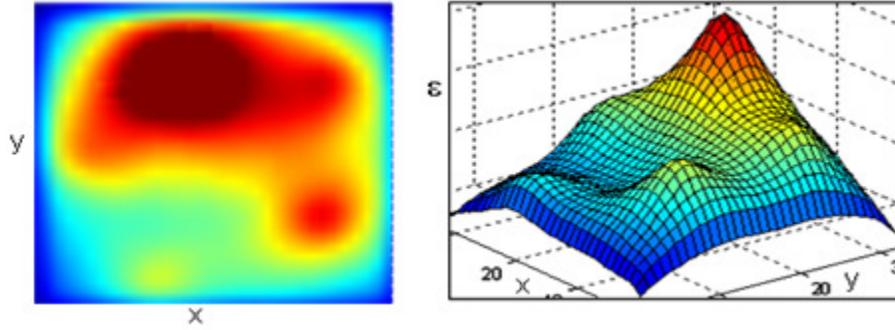
FIGURE 6: IBMS Multi-Feature Fusion: Left: 2D plan view of the fused energy surface. Right: 3D view of the fused energy surface

2.4 Output Sate Estimation

The output from stage-3 is a fused energy map that represents the fusion of the energy maps of multiple features. In order to estimate the mapping between all the input features (in case of Multi-feature fusion) and all decisions (in case of multi-agent fusion) to the output space, a function is needed to do the transformation. The IBMS fusion model implements the image histogram of the segmented fusion region. This operation is expressed mathematically as:

$$h(k) = \sum_{i=1}^{N} M_i, \quad M_i = \begin{cases} 1, & \varepsilon^{MA}(x, y) = k \\ 0, & elsewhere \end{cases} \tag{3}$$

Where,

k = the kth energy disjoint category.

$\varepsilon^{MA}(x, y)$ = the multi-agent fused energy at the coordinates x, y.

M_i = Criteria for counting the matching elements of the energy map.

N = Number of disjoint categories (energy levels), typically 256 levels.

h = The energy histogram function

Criteria count means that the count of each energy level (or range) is incremented when a data point in the energy map is found to match that particular energy level (or range). The resulting histogram may have some noisy profile. This noise depends on the resolution of the histogram which ranges between [1-256] energy levels. Filtering the histogram profile will contribute in smoothing the profile of the output state as it will be investigated in the sensitivity analysis section. To filter the histogram, the following equation is used.

$$\hat{h}(k) = S(h(k)) \tag{4}$$

Where,

h = The energy histogram

$\hat{h}$ = Approximated energy histogram

S = Statistic function {mean, median, mode, etc.}

K = K^{th} Interval of energy levels

The next step in stage-5 of the IBMS fusion engine is to compute the situation assessment as the situation intensity level (SIL). The SIL is a singleton ranging between [0,1] to reflect the output state of a given situation This is equivalent to the deffuzzification stage in the fuzzy inference systems. There are several formulas to consider for this stage. In [28] the most common defuzzification methods: Centroid, Bisector, MOM, LOM and SOM, middle, largest and smallest of maximum respectively are presented. The IBMS fusion model can be customized to fit the context of a particular application by selecting one of these methods. For this paper, the centroid

method is used in this stage based on the justification presented in [28] and is expressed in the following two formulas:

$$\bar{x} = \frac{\sum_{i=1}^{M} \bar{x}_i A_i}{\sum_{i=1}^{M} A_i}, \quad \bar{y} = \frac{\sum_{i=1}^{M} \bar{y}_i A_i}{\sum_{i=1}^{M} A_i}, \quad \varepsilon^O = \bar{x} \tag{5}$$

Where,

A_i = Area of shape (i) in the approximated histogram

$\bar{x}_i, \bar{y}_i$ = coordinates of area (i) centroids

$\bar{x}, \bar{y}$ = Overall centroid of the approximated histogram

ε^O = Output-space Energy

Finally, map the normalized centroid to the output space state. This operation stands for mapping the energy level of situation intensity (SIL) to decision making and actions. The situation intensity level is scaled between [0,1]. The SSL is represented by an energy bar with color spectrum (dark blue, light blue, green, yellow, orange, red, and dark red) coding the critical levels of different situations (from very low to very high). This process is illustrated in Figure 7.
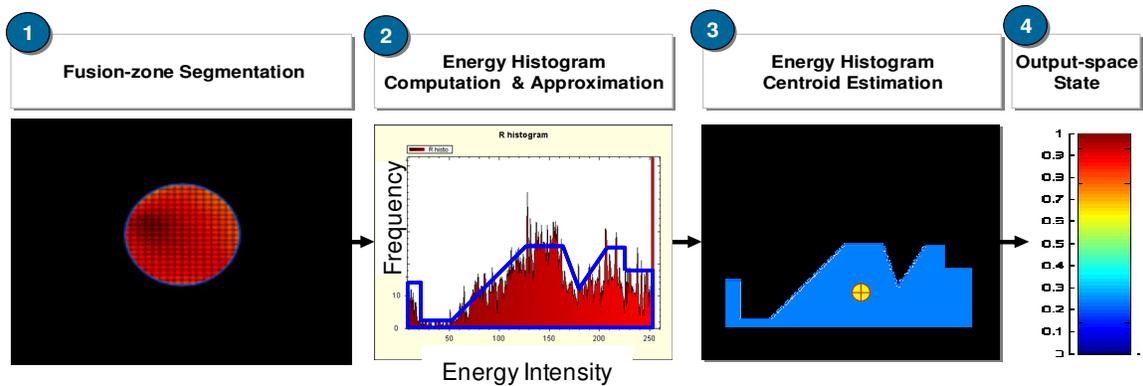
FIGURE 7: The Process of Output State Estimation: Left to right: energy histogram, histogram filtering, and histogram centroid computation

Care should be taken when interpreting the final mapped SSL, since its significance depends on the context of the application. The user of IBMS model can customize the different situation intensity intervals and their thresholds into a security concerns and decisions that fit their application.

2.5 Opportunities for Optimization

The performance of the IBMS fusion model is controlled by a number of control parameters. These control parameters allow to optimize the performance of IBMS fusion model. These parameters are: λ (feature fusion factor), r_{max}(effective fusion radius/the convulsion kernel), w_i(feature weight/importance factor), θ(angle between adjacent feature axis), w_k(agent weight/importance factor), T_r(max radius of fusion region), S(statistic function for energy histogram filtering), and IBMS energy surface smoothing. These parameters and their opportunities for optimization are discussed as follows:

λ *(Energy Propagation Steepness*: equation 1, Figure 1): this parameter controls the influence range of the feature model on adjacent regions of neighboring feature models. As shown in Figure 8, as λ is increased, the energy surface of the feature shrinks causing a lesser contribution

to the fused energy surface. On the other hand, as λ decreases, the energy surface of the feature expands outwardly causing more interaction and contribution to the fused energy surface. Therefore, an optimal setting of λ need to be found where the contributions and interactions of features can be optimized to meet the best output space state estimation
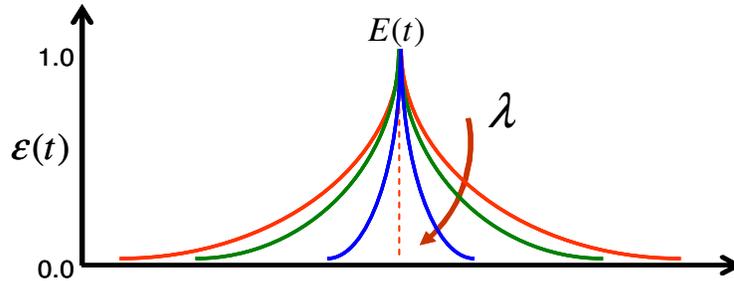
FIGURE 8: Effects of Feature Fusion Factor λ: Explains the influence range of feature energy on the adjacent features. The arrow shows the increasing direction of λ

Effective Fusion Radius (r_{max} in Figure 3): this parameter controls the efficiency of the IBMS fusion algorithm. Without this parameter, the convolution algorithm needs to keep checking on reaching a minimal threshold value to terminate the convolution process. Therefore, solving this problem by introducing a kernel size that is the size of the effective fusion radius r_{max} makes the algorithm more efficient.

Feature Weight (Importance Factor) (w_i in Figure 4): this parameter can signifies the confidence in the current estimation of a particular feature. The higher this factor is, the more contribution and influence the associated feature to this factor should be given, and vice versa. The way this is modeled in IBMS is to let the energy model of the feature slide in and out toward and away from the fusion center causing the effect of the feature on the fused energy surface dynamically changing with respect to its w_i. This phenomenon can be observed in Figure 9 Where, the weights of two features are kept unchanged, while the third one was assigned a higher w_i value. The difference between the two maps is seen to be significant, as the energy surface toward the fusion center is elevated much more in the second (right) map than one in the first (left).
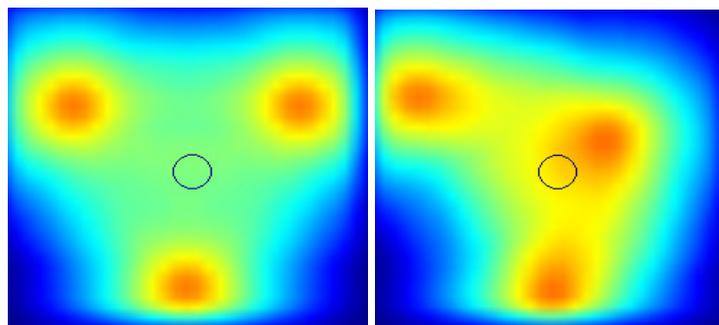
FIGURE 9: Effect of Feature Confidence Level on the Fused Energy Surface: Left: three feature each of which has a w_i of (0.25). Right: two features remained with the same w_i, while the third feature was assigned a w_i of (0.5)

Feature Alignment Angle (θ_i, in Figure 4): this parameter is used to populate and align the feature energy models on the energy map. This may have little or no effect on the fused energy surface. That is due to the fact that rotating a feature model around the fusion center will not increase or decrease its influence on the fused energy surface, it only affects the neighboring energy areas.

Whereas, as shown in the confidence level of features, sliding the energy model radially inward or outward drastically effect the IBMS fusion process. Figure 10 illustrates the impact of this parameter. In the first (left) map, one feature was rotated 45 degrees CW, where as in the second (right) map the same feature was slid toward the fusion center. The difference between the two maps is very clear, where the energy distribution in the second map was significantly altered compared to the first one.
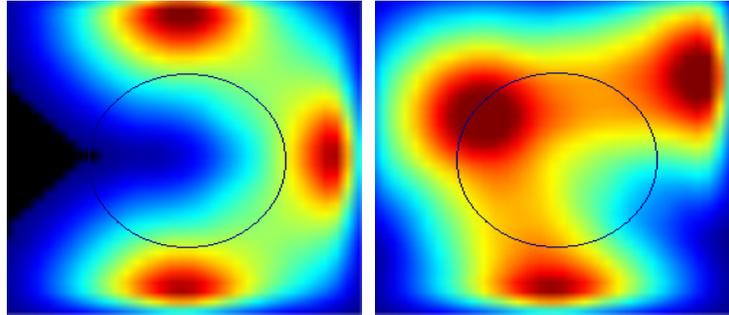
FIGURE 10: Effect of Feature Weight VS. Feature Alignment Angle: Left: one feature is rotated 45 degrees CW. Right: same feature in left is slid toward the center.

Max Radius of Fusion Region: this parameter is very important to the optimization of IBMS, since it determines the most important region of the fused energy surface. Expanding and shrinking the fusion radius can drastically change the resulting energy histogram, from which the output state is inferred using the histogram centroid as presented earlier. The effect of this parameter is demonstrated in Figure 11. In Figure 11 all parameters kept static, but the radius of fusion circle was varied once (top) at 0.2 and a second time (bottom) at 0.9 of the map size. The resulting energy histograms differed drastically as expected. That is because, varying (limiting or expanding) the cropped fusion region will produce a different energy histogram and consequently a different estimation of the output state, which in this case changed from 0.7 to 0.4 with respect to 0.2 and 0.9 fusion radius respectively.
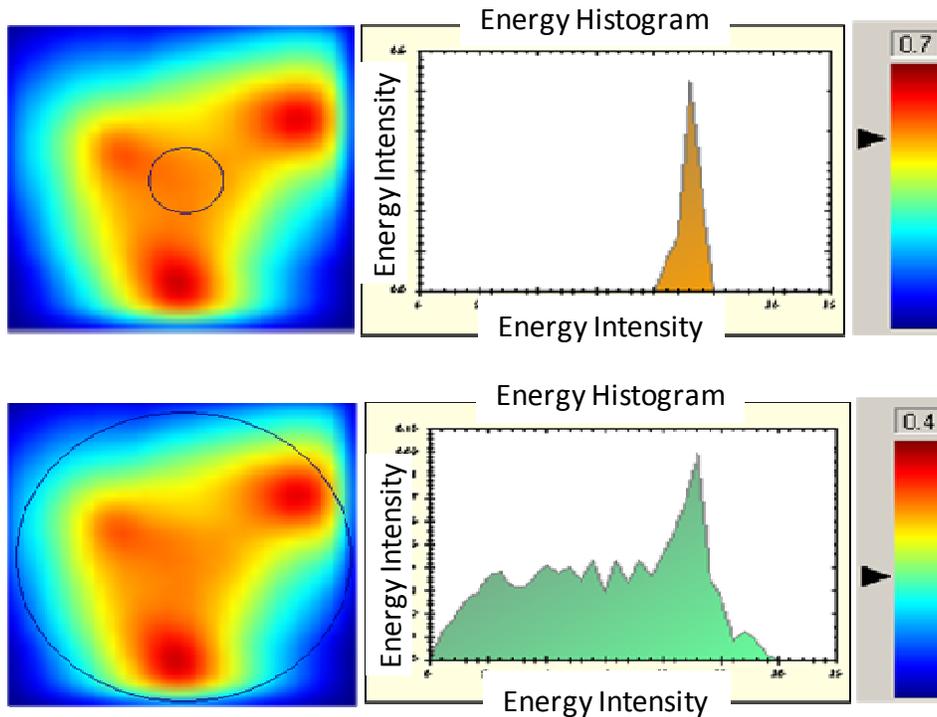
FIGURE 11: Fusion Radius Effects of the Energy Histogram:
Top: fusion radius is set at 0.2 of the map. Bottom: fusion radius is set at 0.9 of the map.
Energy Histogram Filtering: one of the most efficient ways of histogram filtering is by down-sampling it. This method not only filters the histogram in the mean sense, but also reduces the dimensionality of the histogram which makes the subsequent step in IBMS fusion process more efficient, that is the Histogram Centroid Estimation. The question becomes what is the optimal size of the histogram and what statistic to use in filtering the adjacent bins: mean, median, or accumulation. To demonstrate the effects of down-sampling on the energy histogram, Figure 12 shows two histograms for the same cropped fusion region.
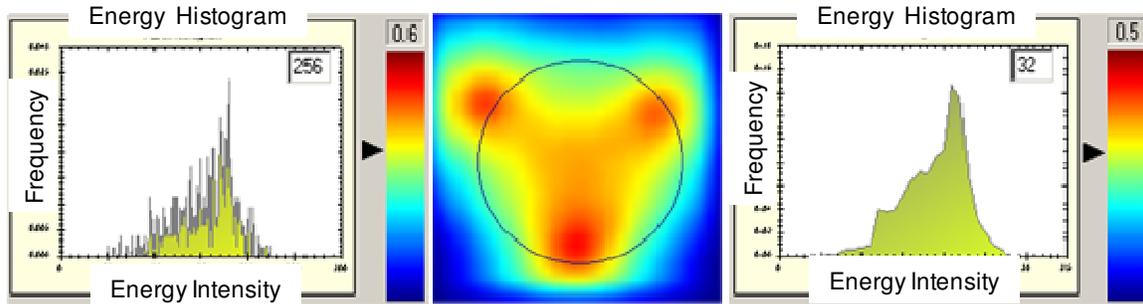
FIGURE 12: Energy Histogram Down-Sampling and Filtering: Left: Energy histogram taken at 256 bins. Middle: the energy map for which the two histograms were computed. Right: Energy histogram taken at 32 bins

As it can be observed in Figure 12, the down sampling of the energy histogram significantly smoothes the histogram but with the cost of shift it by 16% of the original value of the centroid of the histogram. For this reason an optimal size of the histogram need to be investigated.*Energy Surface Smoothing*: the results of the convolution operation in the IBMS fusion process can result in a noisy energy surface and needs to be smoothed. Using typical image processing filtering techniques including: mean, median, Gaussian filters, the surface can be integrated and smoothed out. To demonstrate this operation, Figure 13 illustrates the usage of a mean filter to smooth the energy surface.
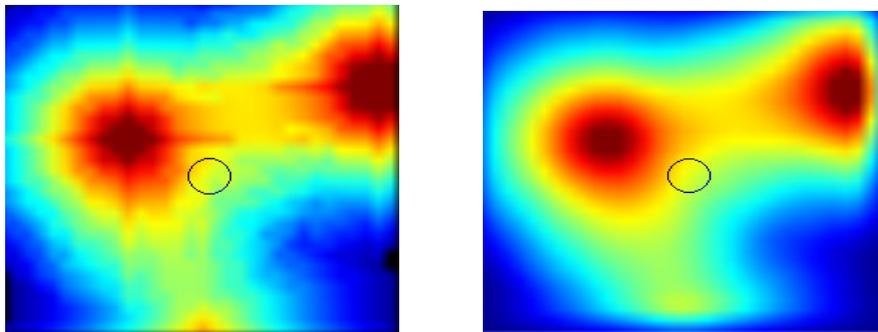
FIGURE 13: Surface Energy Smoothing Using Image Mean Filtering: Left: Raw IBMS image. Right: smoothed IBMS image

SSL Profile Filtering: as it is anticipated with any model to have some level of noise interference due to causes including: sensor noise, feature estimation errors, feature energy noise interferences, etc. IBMS handles the noisy output state estimation by using Gaussian-based zero-phase (*GZ*) signal filter [29]. The GZ filter is temporally applied on the energy level resulting from the equation (5) that is the output space energy (ε^{o}) as follows: let the filtered out space energy be $\varepsilon^{o'}$, GZ be the Gaussian zero-phase filter then, $\varepsilon^{o'}$ can be expressed as follows:

$$\varepsilon^{o'}(t)=GZ(\varepsilon^{o}(t),\pm\Delta t) \tag{6}$$

Where,

Δt: the time interval ($\pm \Delta t$) which the *GZ* is applied as in Figure 14.

In Figure 15, a demonstration of the *GZ* filter is depicted. The concept of GZ filter is to account for an interval of time before and after a particular sample point in a signal. In Figure 14, this sample point is taken as sample number 5 and the time interval is ± 3 units. Therefore the Gaussian PDF is superimposed on that interval aligning its mid-point with the sample point to be filtered. After that, $\varepsilon^{o'}$ is computed as a linear combination of all contributions of each data point within the interval (5 ± 3 units) in the example of Figure 14 according to the formula below:

$$GZ(\varepsilon^{o}(t),\Delta t) = \sum_{t-\Delta t}^{t+\Delta t} G_{pdf}(t)\varepsilon^{o}(t) \tag{7}$$

Figure 15 shows an example of applying the GZ filter on a simulated SSL signal, where it can be seen that it was very effective to eliminate noises significantly.
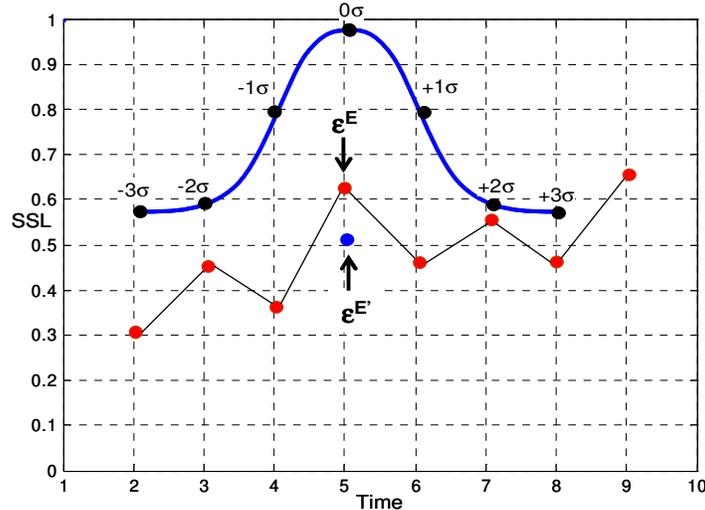
FIGURE 14: Demonstration of Gaussian-based Signal Filtering
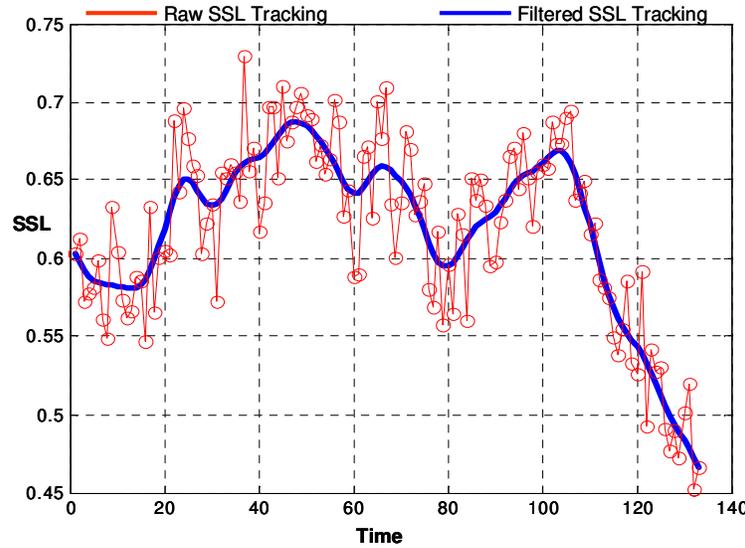
FIGURE 15: Demonstration of GZ Filter on a simulated SSL signal

2.6 Time Complexity Analysis

Time complexity of an algorithm is defined as the mathematical function that asymptotically quantifies the growth of the computational time of the algorithm as a function of the size of the input. Typically, the time complexity is denoted as *T(n)* where, n is the size of the input. The asymptotic mathematical notations for T(n) of three types: O "big O" describes the upper bound of an algorithm similar to "≤", Ω "big omega" describes the lower bound of an algorithm, similar to "≥" and Θ "big theta" describes the situation where, lower and upper bounds are the same function, similar to "=". It is very critical for any model to stand the test of time complexity due to its direct impact on design efficiency, resources allocation, process throughput and response time. Therefore, it was top priority to conduct this time complexity analysis to investigate how our proposed model stands against a well-established soft-computing model, the Fuzzy Logic model. This section will present the analysis of the time complexity of the proposed model IBMS and will be compared to a fuzzy logic fusion model. The analysis will be conducted theoretically as well experimentally. Let n signify number of fuzzy inputs; m signify number of membership functions in a fuzzy input and m is assumed to be uniform across the n inputs to simplify the analysis. Then, it can be observed that a typical fuzzy model consists of four main stages: (1) Fuzzification, which is mapping operation of the crisp inputs into their fuzzy membership functions. This operation takes constant time per each input, therefore, its time complexity is $O(1)$, hence for n inputs, the time complexity becomes $O(n)$ as shown in the Figure 16. (2) Inference, which is applying the fuzzy rules onto the fuzzified inputs. This stage depends on the number of rules in the inference engine. The worst case scenario for such an inference system is to have as many as m^n number of rules which is the inclusive case that cover all possible combinations of inputs to generate the rules for the engine, hence the time complexity of this stage is $O(m^n)$. (3) Aggregation, which also is bounded by the time complexity of the Inference stage, therefore, its time complexity is also $O(m^n)$. (4) Defuzzification is the final stage in fuzzy logic system and it only operates on one input, that is the aggregated areas of the membership functions to find the centroid of these combined areas, therefore it takes $O(1)$ time. Finally, the total theoretical time needed for the fuzzy model to perform its fusion job in the worst case scenario is given by:

$$O(n) + O(m^n) + O(m^n) + O(1) \qquad (8)$$

Which in turn reduces to O(m^n) since it is by far the greatest time complexity among all terms in (8).
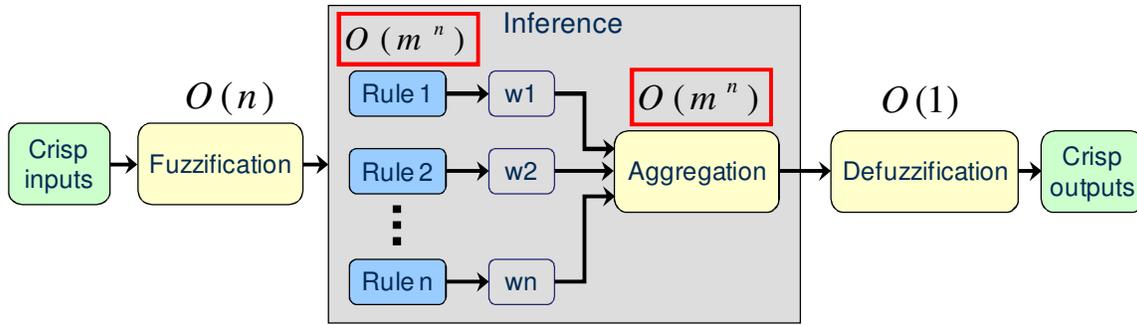
Figure 16: Fuzzy Logic Time Complexity Theoretical Analysis. n = Number of inputs, m = Number of MFs. The red box signifies the dominant term in time complexity

IBMS consists also of four main stages (Figure 17): (1) Multi-feature fusion, which operates on n inputs and takes a constant time per feature, therefore, its time complexity is $O(n)$. (2) Energy Histogram Computation and Approximation, which operates on the resulting fused image and it takes a constant time per fusion cycle, therefore, its time complexity is $O(1)$. (3) Centroid Estimation, which operates on the resulting histogram from the preceding stage and it takes a constant time per fusion cycle, hence, it is time complexity is $O(1)$. (4) Centroid to Energy Scale Mapping, which operates on the resulting centroid in the preceding stage and it takes a constant time every fusion cycle, therefore, its time complexity is $O(1)$. Finally, the total theoretical time needed to perform a IBMS cycle can be expressed in (9) as:
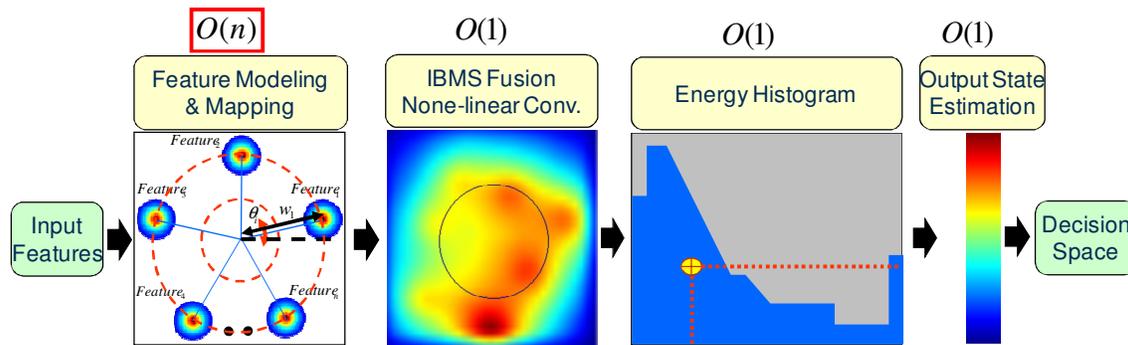
FIGURE 17: IBMS Time Complexity Theoretical Analysis. n = Number of inputs. The red box signifies the dominant term in time complexity

$$O(n)+O(1)+O(1)+O(1) \qquad (9)$$

Which reduces to $O(n)$ which is by far the greatest term in equation 9. Comparing the two results of fuzzy time complexity "$O(m^n)$" and IBMS time complexity "$O(n)$", theoretically, reveals that the IBMS model has a linear time complexity, whereas, the fuzzy model has an exponential time complexity. The Fuzzy logic model has significantly greater time complexity as can be expressed in the following asymptotic notation:

$$\underset{n\to\infty}{Lim}\left(\frac{O(n)}{O(m^n)}\right)=0 \qquad (10)$$

Which states that as n grows to huge values, IBMS time complexity compared to that of fuzzy logic is negligible.
To support this theoretical finding, a set of experiments were conducted on both models to test their time responses by varying number of input features to the models between [1 to 10] and observe their responses. These experiments are summarized Table 1 and Figure 18 respectively.

Table 1 and its graphical plotting in Figure 18 clearly proves that the fuzzy model time complexity follows an exponential curve while the IBMS time complexity follows a linear response.

TABLE 1: Experimental Time Responses Measured for IBMS vs. FUZZY

Number of Features	Time Responces (ms)		
	Fuzzy	**IBMS**	**IBMS/Fuzzy**
2	10.48	31.25	2.98
3	16.17	46.88	2.90
4	22.83	57.81	2.53
5	31.85	62.50	1.96
6	46.61	75.00	1.61
7	75.26	81.00	1.08
8	137.56	87.50	0.64
9	281.44	96.88	0.34
10	622.99	104.69	0.17
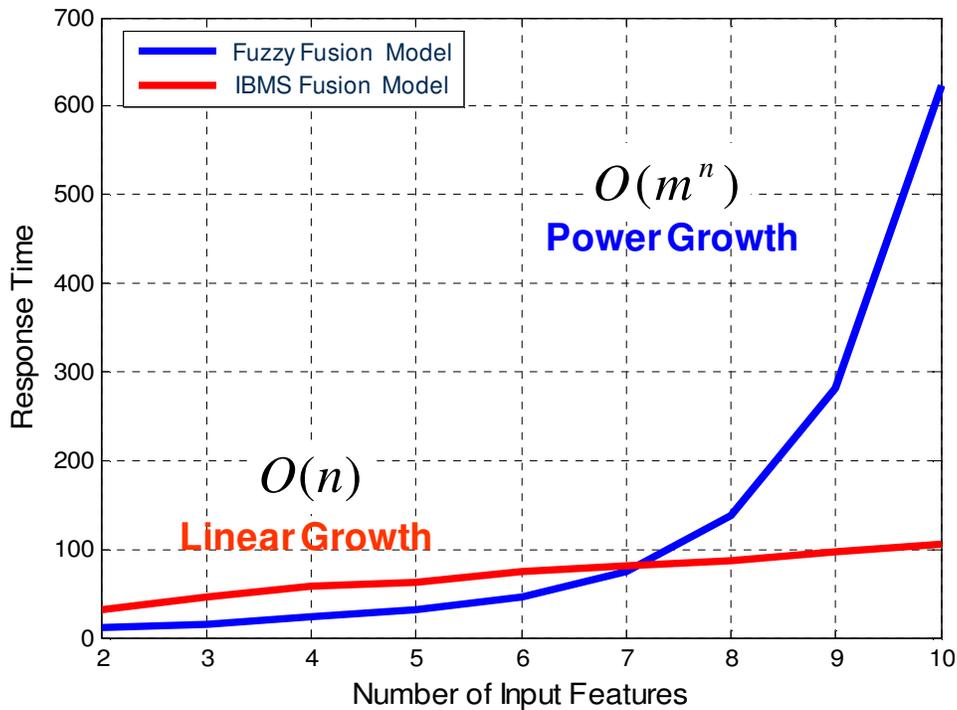
FIGURE 18: Time Response Performance of IBMS Vs. Fuzzy

3. CONCLUSIONS AND FUTURE WORK

We have presented a new an image-based fusion model for multi-sensor multi-source data fusion (IBMS). A complete and detailed theoretical development and analysis was presented for the fusion process of the proposed model. IBMS demonstrated reliability and efficiency over fuzzy logic model. It was found to be superior in time complexity both theoretically and experimentally. As a fusion engine, IBMS has potentials in data and information fusion in areas where the concept of IBMS is applicable. The concept of IBMS is based on the principles of the multi sensor

data fusion [16], where multiple sources can have different views of the same situation without necessarily the same feature directionality. This situation entails a method to combine these sources of information to come up with a unified view of the events in a particular environment and enhance the situation awareness. Potential applications of IBMS include: Application in General Surveillance Systems - The IBMS energy map dynamically changes upon the discovery of changes in any of the mapped features and the whole process of IBMS fusion accordingly responds to these changes as it was demonstrated earlier as more evidence about SSL (depending on the context of the application) IBMS intensifies its fusion surface and accordingly the SSL is elevated so a corresponding decision can be taken. In the light of this concept, some potential application include: military sites surveillance, government facility surveillance systems, boarder security, university campus security systems, law informant surveillance for public area monitoring, etc. Application of IBMS in the Fusion Process - Another potential application of the IBMS model is supporting the JDL fusion model. To relate IBMS to the JDL model, it can be observed that this concept is related to level 2 processing (situation assessment) in the MSDF process of JD. As for the hierarchy of data fusion inference the proposed model of fits two levels of inference: object behavior and situation assessment. This is due to the fact that the proposed model in this work addresses target/phenomena characterization using multi-modality framework for EOI inference and situation assessment. Application of IBMS in Resource Allocation Optimization - Optimization of resources in fusion models (JDL level-4) is very essential and vital for survival and management of the available resources. Mapping available resources to situation requirements is a complex task and needs dynamic intelligent techniques and models to approach it. IBMS model can be utilized to generate the situation map by projecting foe events on the energy map and try to allocate the resources maps to better allocate what resource(s) to what location(s) and events currently occurring in spatial and temporal space of IBMS model.

4. REFERENCES

[01] Trivedi Mohan M., Gandhi Tarak L., and Huang Kohsia S., "Distributed Interactive. Video Arrays for Event Capture and Enhanced Situational Awareness," in the 2005 IEEE Intelligent Systems.

[02] Koller D., Weber J., Huang T., Malik J., Ogasawara G., Rao B., and Russell S., " Toward Robust Automatic Traffic Scene Analysis in Real-Time," IEEE Computer Science Division University of California, Berkeley, CA, 1994. http://dx.doi.org/10.1109/ICPR.1994.576243

[03] Kogut Greg, Blackburn Mike, Everett H.R., "Using Video Sensor Networks to Command and Control Unmanned Ground Vehicles," Space and Naval Warfare Systems Center San Diego, 2000. www.spawar.navy.mil/robots/pubs/usis03diva.pdf

[04] Chang Edward Y. and Wang Yuan-Fang, "Toward Building a Robust and Intelligent Video Surveillance System: A Case Study," Dept. of Electrical and Computer Engineering Dept. of Computer Science, Univ. of California, 2003. www.cs.ucsb.edu/~yfwang/papers/icme04_invited.pdf

[05] Bodo Robert, Jackso Bennett, and Papanikolopoulos Nikolaos, "Vision-Based Human Tracking and Activity Recognition," 2002. AIRVL, Dept. of Computer Science and Engineering, University of Minnesota. http://mha.cs.umn.edu/Papers/Vision_Tracking_Recognition.pdf

[07] Ma Yunqian, Miller Ben, Buddharaju Pradeep, Bazakos Mike, "Activity Awareness: From Predefined Events to New Pattern Discovery," Proceedings of the Fourth IEEE International Conference on Computer Vision Systems (ICVS 2006). www.cs.colostate.edu/icvs06/Program.pdf

Aaron R. Rababaah

[08] Collins Robert T., Lipton Alan J., Fujiyoshi Hironobu and Fellow Takeo Nade, IEEE, "Algorithms for Cooperative Multi-sensor Surveillance," 2001. Processing of the IEEE, VOL. 89, NO. 10, October 2001.

[09] Goh King-Shy, Miyahara Koji, Radhakrishan Regunathan, Xiong Ziyou, and Divakaran Ajay, Audio-Visual Event Detection based on Mining of Semantic Audio-Visual Labels, MERL – A MITSUBISHI ELECTRIC RESEARCH LABORATORY, http://www.merl.comTR-2004-008 March 2004

[10] Varshney Pramod K., Mehrotra Kishan G., and Mohan Chilukuri K., Decision Making and Reasoning with Uncertain Image and Sensor Data, Syracuse University. http://www-video.eecs.berkeley.edu/vismuri/meeting2006/syracuse.ppt

[11] Klapuri, Audio signal classification, ISMIR Graduate School, October 4th-9th, 2004, http://mtg.upf.edu/ismir2004/graduateschool/people/Klapuri/classification.pdf

[12] Temko Andrey, Malkin Robert, Zieger Christian, Macho Dusan, Nadeu Climent, and Omologo Maurizio, Acoustic Event Detection And Classification In Smart-Room Environments: Evaluation Of Chil Project Systems, TALP Research Center, UPC, Barcelona, Spain, interACT, CMU, Pittsburgh, USA, ITC-irst, Povo (TN), Italy Zaragoza • Del 8 al 10 de Noviembre de 2006.

[13] Stauffer Chris, Automated Audio-visual Activity Analysis, Compute Science and Artificial Intelligence Laboratory, MIT-CAAIL-TR-2005-057, AIM-2005-026, 09, 20, 2005

[14] Chen Datong, Malkin Robert, Yang Jie, Multimodal Detection of Human Interaction Events in a Nursing Home Environment, School of Computer Science, Carnegie Mellon University, Sixth International Conference on Multimodal Interfaces (ICMI'04), Penn State University, State College, PA, October 14-15, 2004

[15] Gerosa L., Valenzise G., Tagliasacchi M., Antonacci F., and Sarti A.. Scream And Gunshot Detection In Noisy Environments, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy, VISNET II, a network of excellence 2006.

[16] Hall David L. and McMullen Sonya A. H. Mathematical Techniques in Multisensor Data Fusion, 2nd edition, 2004. Artech House, INC. Norwood, MA.

[17] Brooks Alex and Williams Stefan, "Tracking People with Networks of Heterogeneous Sensors," ARC Centre of Excellence in Autonomous Systems. School of Aerospace, Mechanical, and Mechatronic Engineering, University of Sydney, NSW Australia, 2003 www.acfr.usyd.edu.au/publications/downloads/2004/Brooks210/Brooks03Tracking.pdf

[18] Meyer Michael, Ohmacht Thorsten and Bosch Robert GmbH and Michael Hotter, "Video Surveillance Applications Using Multiple Views of a Scene," IEEE AES Systems Magazine, March 1999.

[19] Turollu A., Marchesotti L. and Regazzoni C.S., "Multicamera Object Tracking In Video Surveillance Applications," DIBE, University of Genoa, Italy, 2001. http://cmp.felk.cvut.cz/cvww2006/papers/39/39.pdf

[20] Niu Wei, Long Jiao, Han Dan, and Wang Yuan-Fang, "Human Activity Detection and Recognition for Video Surveillance," the 2004 IEEE International Conference on Multimedia and Expo. www.cs.ucsb.edu/~yfwang/papers/icme04_tracking.pdf

[21] Siebel Nils T. and Maybank Steve, "Fusion of Multiple Tracking Algorithms for Robust People Tracking, Annotated Digital Video for Intelligent Surveillance and Optimized Retrieval (ADVISOR),". Computational Vision Group, Department of Computer Science, The University of Reading, Reading RG6 6AY, England, 2001. www.cvg.reading.ac.uk/papers/advisor/ECCV2002.pdf

[22] Ellis Tim. Multi-camera Video Surveillance. 2002. Information Engineering Center, School of engineering, City university, London. 0-7803-7436-3/02 2002 IEEE.

[23] Snidaro L., Niu R., Varshney P.K., and Foresti G.L., "Automatic Camera Selection and Fusion for Outdoor Surveillance Under Changing Weather Conditions," Dept. of Mathematics and Computer Science, University of Udine, Ital, Department of EECS, Syracuse University, NY, 2003. www.ecs.syr.edu/research/SensorFusionLab/Downloads/Rrixin%20Niu/avss03.pdf

[24] Boult Terrance E., "Geo-spatial Active Visual Surveillance on Wireless Networks," Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop. http://vast.uccs.edu/.../PAPERS/IEEE-IAPR03-Geo-Spatial_active_Visual_Surveillance-On-Wireless-Networks-Boult.pdf

[25] Chang Edward Y. and Wang Yuan-Fang, "Toward Building a Robust and Intelligent Video Surveillance System: A Case Study," Dept. of Electrical and Computer Engineering Dept. of Computer Science, Univ. of California, 2003. www.cs.ucsb.edu/~yfwang/papers/icme04_invited.pdf

[26] A. Turolla, L. Marchesotti and C.S. Regazzoni, "Multicamera Object Tracking in Video Surveillance Applications", DIBE, University of Genoa, Italy

[27] Sensory Physiology Lettuces at University Minnesota Dulutth, 03.17.2012. http://www.d.umn.edu/~jfitzake/Lectures/DMED/SensoryPhysiology/GeneralPrinciples/CodingTheories.html

[28] MatLab Documentation on Fuzzy Logic. MatLab7 R14. The MathWorks Documentation. http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/fuzzy.html?BB=1

[29] MatLab Documentation on Signal Processing – Signal Windowing models. MatLab7 R14. The MathWorks Documentation. http://www.mathworks.com/documentation/signalprocessing/hamming

Kun Chen, Yan Ma & Jun Liu

Segmentation by Fusion of Self-Adaptive SFCM Cluster in Multi-Color Space Components

Kun Chen   *kun_1949301@126.com*
Student, Department of Information Mechanical
and Electrical Engineering
Shang Hai Normal University
Shang Hai, 200234, China

Yan Ma   *ma-yan@shnu.edu.cn*
Professor, Department of Information Mechanical
and Electrical Engineering
Shang Hai Normal University
Shang Hai, 200234, China

Jun Liu   *liujunout@126.com*
Student, Department of Information Mechanical
and Electrical Engineering
Shang Hai Normal University
Shang Hai, 200234, China

Abstract

This paper proposes a new, simple, and efficient segmentation approach that could find diverse applications in pattern recognition as well as in computer vision, particularly in color image segmentation. First, we choose the best segmentation components among six different color spaces. Then, Histogram and SFCM techniques are applied for initialization of segmentation. Finally, we fuse the segmentation results and merge similar regions. Extensive experiments have been taken on Berkeley image database by using the proposed algorithm. The results show that, compared with some classical segmentation algorithms, such as Mean-Shift, FCR and CTM, etc, our method could yield reasonably good or better image partitioning, which illustrates practical value of the method.

Keywords: Color image Segmentation, Histogram, SFCM, Fusion, Multi-color Space Components

1. INTRODUCTION

Image segmentation is a popular technique for image processing. The purpose of image segmentation is to divide an image into regions that can be considered homogeneous with respect to a given criterion such as gray level, color or texture, etc [1-2]. Image segmentation is one of the most widely studied problems in image analysis and computer vision, and it is a significant step towards image understanding. Since color images carry much more color information which is important to human perception, with the rapid growing of computer processing ability, recently color image segmentation has became a hot research topic. It is widely applied in many areas such as: image compression, internet video transmission, medical image diagnosis and target tracking, etc. We should solve two problems for image segmentation: (1) choose the right color space; (2) select the appropriate segmentation strategy. Since the selection of color space depends on specific image and segmentation strategy, nowadays there is no color space can be suited for all color images [3].

Many methods have been proposed and studied in the last decades to solve the color image segmentation problem. Some researchers prefer to use more complicated feature selection procedures or more elaborate clustering techniques and then improve the final segmentation

result by complex optimization method. Some segmentation techniques integrated with specific theory, method and means has emerged, such as segmentation based on lossy data compression [4-5], wavelet-domain hidden markov models [6], graph-based [7-8], Mean-Shift [9] and etc. Some researchers also use information fusion strategies to get better performance. They prefer to fuse the results associated with the simple method applied on different color spaces rather than to consider complex segmentation theory or model. Eg. Mignotte [10] proposed a method called FCR by fusion of multi-color spaces based on local histogram and K-means clusters. First, a simple clustering model based on local histogram has been proposed in [10], then, the model has been applied into RGB、HSV、YIQ、XYZ、LAB、LUV color spaces to achieve six segmentation results. Finally, six segmentation results have been fused to achieve the best segmentation results. It is a simple and effective method which makes use of the advantage of many different color spaces. But it also has some problems as follows: (1) The runtime of local histogram clustering modeling is too long. (2) The number of clusters is fixed, therefore, it can not meet the self-adaptive requirement for different images.

Learning from Mignotte's idea, we propose a novel, simple, efficient and self-adaptive method by fusion of multi-color space components. First, we choose six different color components elaborately through various experiments: Gray component, V(HSV) component, I(YIQ) component, Cr(YCbCr) component, B(LAB) component and U(LUV) component. Then we propose a peak-finding algorithm to determine cluster number of each component and initialize cluster centroid for SFCM clustering. Then, A clustering method is proposed to fuse six different segmentation results, where the cluster number is the mean of the above six cluster numbers. Finally, we propose region merging method to merge the previous segmentation results. The proposed method is tested on Berkeley natural image database. Extensive experiments show that, the method is simple, efficient, and robust to noise. Compared with FCR, our method can get better result and faster. Compared to the state-of-the-art segmentation methods recently proposed in the literature, our method performs competitively in terms of visual evaluations and quantitative performance measures.

2. INITIAL SEGMENTATION

FCM algorithm has been used as one of the most popular cluster techniques for image segmentation in computer vision and pattern recognition. It is developed by Dunn [11] in 1973 and improved by Bezdek[12] in 1981. Although FCM has been widely used in image segmentation domains, it still exists the following problems: (1) In terms of performance the algorithm depends on the initial cluster centroids; (2) The cluster number must be fixed before clustering; (3) High computational complexity; (4) No consideration of spatial information. Taking into account above problems, we use histogram technique to find initial cluster centroids and determine cluster number. We only cluster 1-D component of each color space in terms of computational complexity, therefore the method is simple and rapid. We use SFCM [13] to consider spatial information and achieve initial segmentation results. After initial segmentation, we achieve six different initial segmentation results from different color space components (Gray component, V component, I component, Cr component, B component, U component).

2.1 Peak Finding

How to determine initial cluster centroids has always been a problem of clustering. Good initial cluster centroids not only can yield better cluster results but also can make cluster faster. Selecting initial cluster centroids randomly is likely to lead the optimization of the algorithm's objective function to local extreme, therefore the accuracy of the cluster results will be affected. In this paper we utilize histogram technique to find cluster centroid. Here we take gray component as an example to propose the peak finding algorithm. In this way we also can obtain the peaks of other components. The procedure is as follows:

  1) Quantize gray component into 0-255 intensity levels, count the frequency, and create the histogram. Let $g(i)$ be the gray component histogram, x_i be the number of pixels associated with

ith intensity level in g(i).The histogram of gray component can be represented by the following equation:

$$g(i) = x_i, \quad 0 \leq i \leq 255 \tag{1}$$

2) Smooth histogram. Use 1D Gaussian filters with size of 1×5 for $g(i)$ to smooth twice, and the result of smoothing depends on Gaussian standard covariance σ_g. The histogram is more smoother with bigger σ_g. We have a new histogram $T_g(i)$ after smoothing.

3) Search for initial peaks. We search turning points on which gradient value varies from positive to negative. We take these turning points as initial peaks and get initial set of peaks P_1.

4) Remove small peaks. If the value of peak in set P_1 is less than threshold T_1, it is removed from P_1. So we have new set of peaks P_2.

5) Remove adjacent peaks and generate final peaks. If two peaks in P_2 are close enough, we think the gray values of the regions represented by the two peaks are similar. Therefore, we remove the smaller one while the distance between two peaks is less than threshold T_2. We get final set of peaks P_3.

2.2   Spatial FCM Clustering

The classical FCM algorithm is to assign pixels to each cluster by using fuzzy memberships. Let $X = (x_1, x_2, \cdots x_n)$ denotes an image with n pixels to be partitioned into c clusters, where x_i represents multispectral (features) data. The result of classification can be represented by a fuzzy membership degree matrix $U = \{\mu_{ik}\}$, where μ_{ik} represents the membership degree of kth pixel to ith cluster centroid. it is subject to the following constraints:

$$\mu_{ik} \in [0,1], \forall i,k \; ; \; 0 < \sum_k \mu_{ik} < n, \forall i \; ; \; \sum_i \mu_{ik} = 1, \forall k \tag{2}$$

FCM algorithm is an iterative optimization that minimizes the cost function defined as follows:

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ik}^{l} \left\| x_k - v_i \right\|^2, \tag{3}$$

Where $U = \{\mu_{ik}\}$ is the membership degree matrix according to Eq.(2), $V = \{v_1, v_2, \cdots v_c\}$ is the set of cluster centroids, $\left\| x_k - v_i \right\|$ represents the distance of pixel x_k to cluster centroid v_i, and we use Euclidean distance in initial segmentation. The parameter l controls the fuzziness of the resulting partition, and $l = 2$ is used in this study.

The membership functions μ_{ik} and the centroids V_i are updated iteratively as follows:

$$\mu_{ik} = \frac{\left\| x_k - v_i \right\|^{-2/(l-1)}}{\sum_{j=1}^{c} \left\| x_k - v_j \right\|^{-2/(l-1)}} \tag{4}$$

$$v_i = \frac{\sum_{k=1}^{n} \mu_{ik}^{l} x_k}{\sum_{k=1}^{n} \mu_{ik}^{l}} \tag{5}$$

The standard FCM algorithms is optimized when pixels close to their centroids are assigned high membership values, while those that are far away are assigned low values.

One of the problems of classical FCM algorithm in image segmentation is the lack of spatial information. Since image noise and artifacts often impair the performance of FCM segmentation, it would be attractive to incorporate spatial information into FCM. Chuang et al. [13] proposed a spatial FCM algorithm in which spatial information can be incorporated into fuzzy membership functions directly using

$$\mu_{ik}^{'} = \frac{\mu_{ik}^p h_{ik}^q}{\sum_{j=1}^{c} \mu_{jk}^p h_{jk}^q} \qquad (6)$$

Where p and q are two parameters controlling the respective contribution. The variable h_{ik} includes spatial information by

$$h_{ik} = \sum_{j \in N_k} \mu_{ij} \qquad (7)$$

Where N_k denotes a local window centered around the image pixel k. The weighted μ_{ik} and the centroid v_i are updated as usual according to Eq. (4) and (5).

3. FUSION OF INITIAL SEGMENTATION RESULTS

We get six different initial segmentation results from six different color space components by using the method proposed in section 2. The cluster number of them is different, we record them as $K_i, 1 \leq i \leq 6$ (for example K_1 represent the cluster number of gray component, K_2 represent V component, etc). We use SFCM algorithm again to fuse above six results which with different cluster number and get a new result I_{fusion} after fusion.

3.1 Extract Feature Vector

For each initial segmentation result with $K_i (1 \leq i \leq 6)$ cluster number, considering the squared fixed-size ($N_W \times N_W$) neighborhood centered around the pixel. Let W_x represent the neighborhood of pixel location x. We calculate the normalized local histogram of the class labels for each pixel within W_x:

$$h(W_x) = (\frac{n_1}{N_w^2}, \frac{n_2}{N_w^2} \cdots \frac{n_{K_i-1}}{N_w^2}, \frac{n_{K_i}}{N_w^2}) \qquad (8)$$

Where $h(W_x)$ represent the feature vector of pixel location x in one of the six segmentation results, n_j denotes the number of pixels whose class labels are j within W_x. We do the same process toward six different segmentation result described above. After that, we get six feature vector location in the same place for each pixel. Then combine them in series and normalized.

Finally, We get the fused local histogram of the class labels $h^*(W_x)$ with dimension $M = \sum_{i=1}^{6} K_i$, which is used as feature vector for input in the final clustering.

3.2 Fusion of Initial Segmentation by SFCM

We adopt SFCM algorithm (described in Section 2.2) again to partition $h^*(W_x)$ into N classes.

$$N = ceil(\sum_{i=1}^{6} K_i / 6) \qquad (9)$$

Where ceil(A) represents round the elements of A to the nearest integers. We get segmentation result I_{fusion} by fusion, in which the distance between two feature vectors from local histogram of the class labels is calculated by Bhattacharya distance:

$$D_B[h_1^*, h_2^*] = \left(1 - \sum_{i=1}^{M} \sqrt{h_1^* \cdot h_2^*}\right)^{1/2} \tag{10}$$

Where h_1^*, h_2^* denote two normalized feature vectors from local histogram of the class labels, M denotes the dimension of feature vector.

4. REGION MERGING

Segmentations with clustering are often featured with numerous discrete small regions. The spatial connectivity between pixels in the same cluster could hardly be guaranteed. These minor regions on one hand preserves the image detail but on the other hand largely affects the segmentation quality. To generate reasonable segmentations, a simple and effective region merging strategy is necessary for this issue. In this paper, the region merging method is presented in LUV color space. The steps are as follows:

1、 Relabel regions after segmentation by 8-neighbors, which yields that not adjacent and color-homogeneous regions are marked with different labels.

2、 Search adjacent regions after relabeling.

3、 Calculate the mean value of L, U, V components for relabeled regions.

4、 Merge small regions. If the size of region is smaller than threshold T_3, it will be merged into its bigger adjacent region with the smallest Euclidean distance in LUV color space, and whose size is greater than T_3.

5、 Merge big regions. If the size of region is smaller than threshold $T_4 (T_3 << T_4)$, we calculate Euclidean distance in LUV color space with its adjacent regions whose size is greater than T_4. Search the smallest distance dc. If $dc < T_5$, the region will be merged into its adjacent bigger region with the smallest distance, and vice versa.

5. EXPERIMENT RESULTS AND ANALYSIS

The proposed algorithm is demonstrated on the computer Inter Core2 Duo CPU T6570 2.10GHz. We use Matlab R2011a to test the segmentation results on natural images in the Berkeley segmentation database[14], which also contains benchmark segmentation results obtained from human subjects. We have done numerous experiments which show that the results are best when the involved parameters σ_g, T_1, T_2 chosen to 3, 0.001S (S denotes the size of image),15, the window size $N_w \times N_w$ chosen to 5×5 and T_3, T_4, T_5 chosen to 0.003S, 0.05S, 50. We will analyze our algorithm from the following aspects: the choice of different color components, whether the algorithm is robust to noise and compare the algorithm with some state-of-art methods qualitatively and quantitatively.

The quantitative comparison is based on the following performance measures, namely a probabilistic measure called PRI [15,16] (higher probability is better) and three metrics VoI [17], GCE [14], and BDE [18] (lower distance is better). The qualitative meaning of these performance measures are recalled as follows.

   1) PRI (Probabilistic Rand Index) counts the fraction of pairs of pixels whose labellings are consistent between the computed segmentation and the ground truth, averaging across multiple ground truth segmentations to account for scale variation in human perception.

   2) VoI (Variation of Information) defines the distance between two segmentations as the average conditional entropy of one segmentation given the other, and thus roughly measures the amount of randomness in one segmentation which cannot be explained by the other.

   3) GCE (Global Consistency Error) measures the extent to which one segmentation can be viewed as a refinement of the other. Segmentations which are related in this manner are

considered to be consistent, since they could represent the same natural image segmented at different scales.

    4) BDE (Boundary Displacement Error) measures the average displacement error of boundary pixels between two segmented images. Particularly, it defines the error of one boundary pixel as the distance between the pixel and the closest pixel in the other boundary image.

5.1 Choice of Different Color Components

Extensive experiments show that the selection of different color components has important influence on the segmentation result. In order to compare with FCR [10] algorithm, we choose six components to fuse. Which six different color components are the best? We use self-adaptive histogram and SFCM clustering techniques to quantitatively test the components of HSV, YIQ, YCbCr, LAB, LUV color spaces and gray component on randomly chosen images.

TABLE 1 shows the PRI, VoI, GCE and BDE performance of these 14 components on 100 randomly chosen images in the Berkeley segmentation database. Best performance of each measure is marked with bold. Second best is marked with underline. In PRI indice, V(HSV) component is best, Gray component is second best; In VoI indice, B component is best, I is second best; In BDE indice, Cr component is best, V component is second best. TABLE 1 also shows that some component is the best in one indice, but worse in other indices. Eg. A component has the best GCE indice, but PRI、BDE is worse. Therefore, we need to consider different performance measures of components together to select the best components. In the analysis, we choose Gray, V(HSV), I, Cr, B, U as six different components to fusion.

Component	PRI[15,16]	VoI[17]	GCE[14]	BDE[18]
Gray	<u>0.7045</u>	3.0394	0.3894	10.3919
H	0.6773	2.8031	0.3204	12.6409
S	0.6860	3.1189	0.3909	12.2657
V(HSV)	**0.7146**	3.0790	0.3953	<u>10.1278</u>
Y	0.6980	2.9322	0.3855	10.4577
I	0.6833	<u>2.6635</u>	0.3169	11.6272
Q	0.6511	2.9453	0.3485	11.9675
Cb	0.6782	2.9347	0.3612	10.2231
Cr	0.6776	2.9786	0.3565	**9.8885**
L	0.6958	2.8327	0.3282	10.7621
A	0.6190	2.6721	**0.2878**	14.6134
B	0.6767	**2.6045**	<u>0.2952</u>	11.5503
U	0.6568	2.7091	0.3013	10.7357
V(LUV)	0.6901	2.8074	0.3258	10.9033

TABLE 1: The Performance Measures of 14 Components.

5.2 Robust to Noise

According to [13],we know that SFCM algorithm is less sensitive to noise. Because the clustering of our algorithm is based on SFCM, we conclude that our algorithm may be robust to noise. In order to test it, we add the Gaussian noise (mean value is 0,variance is 0.03) to two randomly chosen images for segmentation. FIGURE 1 shows their original images, noise images and segmentation result by our proposed method. The result shows that even with Gaussian noise, we still can clearly get the correct part of the segmentation result, which proven the algorithm's robustness to noise. The reasons can be concluded as follows: First, our method use SFCM clustering which considering spatial information and can get better clustering results to noise image. Second, the proposed method adopts region merging technique after fusion of different segmentations, which can also effectively remove small noises.

| (a) Original image | (b) Noise image | (c) Segmentation |
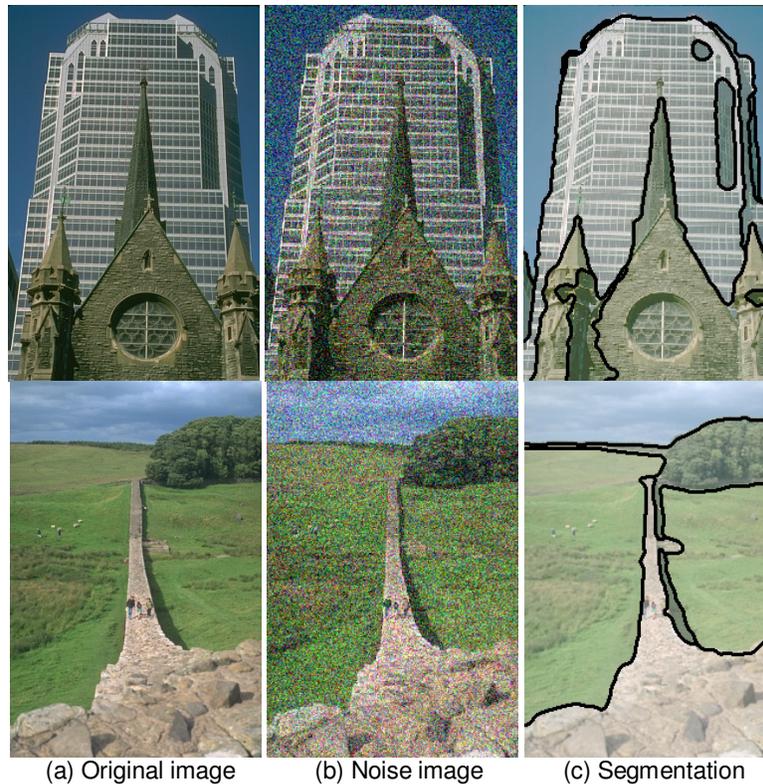
FIGURE 1: Noise Image Segmentation

5.3 Comparison with State-of-the-art Methods

We test 300 images on Berkeley image database and compare our method with state-of-the-art methods such as: Mean-shift [9], NCuts [7], FH [8], CTM [4,5] and FCR [10].

FIGURE 2 shows the segmentation results of FCR, Mean-shift, CTM and our proposed method with 5 randomly chosen images. FIGURE 2(a) is original images. FIGURE 2(b) shows FCR segmentation results. FIGURE 2(c) is Mean-shift results. FIGURE 2(d) shows CTM results. FIGURE 2(e) is our proposed method. It is obvious that FCR and Mean-shift methods have over-segmentation problem in FIGURE 2. For certain images, these two methods can only yield small piece regions, and can't generate the right object, especially Mean-shift method. Our method can get better results which is close to human perception and has less over-segmentation problem.

TABLE 2 shows the mean value of performance measures over the 300 images of the Berkeley image database in different methods. Best performance of each measure is marked with bold. Second best is marked with underline. From TABLE 2 we can see that our method outperforms other methods for several different internal parameters, all the well-known segmentation algorithms presented in TABLE 2 in terms of PRI and BDE indices, second best in VoI indice and is obviously better than FCR in PRI, VoI and BDE indices.

TABLE 3 shows the average runtime of 100 randomly chosen images in the same platform. It is obvious that our method faster than FCR algorithm.

(a) Original Image

(b) FCR

(c) Mean-shift

(d) CTM

(e) Proposed Method

FIGURE 2: Comparison of FCR, Mean-shift, CTM and Our Method

Algorithms	PRI[15,16]	Vol[17]	GCE[14]	BDE[18]
Humans	0.8754	1.1040	0.0797	4.9940
FCR(K_1 =6, K_2 =6, k =0.13)[10]	<u>0.7842</u>	2.3925	0.2169	<u>9.2463</u>
CTM(η =0.1)[4,5]	0.7561	2.4640	**0.1767**	9.4211
CTM(η =0.2)[4,5]	0.7617	**2.0236**	<u>0.1877</u>	9.8962
Mean-shift[9]	0.7550	2.4770	0.2594	9.7001
NCuts[7]	0.7229	2.9329	0.2182	9.6038
FH[8]	0.7841	2.6647	0.1895	9.9497
Our Method	**0.7906**	<u>2.1395</u>	0.2218	**9.0652**

TABLE 2: Performance Measures Comparison to State-of-the-art Methods

Algorithms	Our Method	FCR
Runtime/s	138.732	317.859

TABLE 3: Runtime of Our Method and FCR.

6. CONCLUSION

This paper proposes a novel, simple, efficient and self-adaptive method by fusion of multi-color space components. Results show that the method provides good segmentation on a variety of color images. Histogram and SFCM cluster techniques are used in initial segmentation. The strategy not only can locate initial cluster centroids quickly but also can solve the problem of that clustering number is fixed. Then an effective fusion and region merging strategy is used to make segmentation result more close to human perception. The proposed method has been successfully applied on the Berkeley image database, and performs competitively among the recently reported state-of-the-art segmentation methods in terms of visual evaluations and quantitative performance measures. In our experiments, several limitations are found for the algorithm. One case is when the color of an image is too close, the segmentation result is bad. Another case is the algorithm only consider color information, do not consider other information such as texture. Future research work is on how to solve these problems and improve the results.

7. REFERENCES

[1]   H. D. Cheng, X. H. Jiang, J. L. Wang. " Color Image Segmentation Based On Homogram Thresholding And Region Merging". Pattern Recognition, vol.36, pp. 373-393, 2002.

[2]   H. D. Cheng, J. Li." Fuzzy Homogeneity And Scale-Space Approach To Color Image Segmentation". Pattern Recognition, vol.36, pp. 1545-1562, 2003.

[3]   K. Y. Lin, J. H. Wu, L. H. Xu. " A Survey On Color Image Segmentation Techniques". Journal

of   Image And Graphics, vol. 10, pp. 1-10, 2005.

[4]   A. Y. Yang, J. Wright, S. Sastry, and Y. Ma. "Unsupervised Segmentation Of Natural Images Via Lossy Data Compression". Comput. Vis. Image Understand, vol. 110, pp. 212-225, 2008.

[5]   Y. Ma, H Derksen, W. Hong, and J. Wright. "Segmentation Of Multivariate Mixed Data Via Lossy Coding And Compression". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, pp. 1546-1562, 2007.

[6]   H. Choi, R. G. Baraniuk. "Multiscale Image Segmentation Using Wavelet-Domain Hidden Markov Models". IEEE Transactions on Image Processing, vol. 10, pp. 1309-1321, 2001.

[7]   J. Shi and J. Malik. "Normalized Cuts and Image Segmentation". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, pp. 888-905, 2000.

[8]   P. Felzenszwalb and D. Huttenlocher, "Efficient Graph-Based Image Segmentation". Int. J. Comput. Vis., vol. 59, pp. 167-181, 2004.

[9]   D. Comanicu, P. Meer. "Mean shift: A Robust Approach Toward Feature Space Analysis". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, pp. 603-619, 2002.

[10]   M. Mignotte. "Segmentation By Fusion Of Histogram-Based K-Means Clusters In Different Color Spaces". IEEE Transactions on image processing, vol. 17,pp. 780-787, 2008.

[11]   J.C. Dunn. "A Fuzzy Relative Of The ISODATA Process And Its Use In Detecting Compact, Well Separated Cluster". Cybernetics, vol. 3, pp. 32–57, 1973.

[12]   J. C. Bezdek, "Pattern Recognition With Fuzzy Objective Function Algorithms," in Plenum Press, New York, 1981.

[13]   K. S. Chuang, H. L. Tzeng, S. Chen, J. Wu, T. J. Chen. "Fuzzy C-Means Clustering With Spatial Information For Image Segmentation". Computerized Medical Imaging and Graphics, vol. 30, pp. 9-15, 2006.

[14]   D. Martin, C. Fowlkes, D. Tal, et al. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in Proc. of the 8th IEEE International Conference on Computer Vision, 2001, pp. 416-423.

[15]   R. Unnikrishnan, C. Pantofaru, M. Hebert. "A measure for objective evaluation of image segmentation algorithms," in Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, pp. 34-41.

[16]   R. Unnikrishnan, C. Pantofaru, M. Hebert. "Toward Objective Evaluation Of Image Segmentation Algorithms". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, pp. 929-944, 2007.

[17]   M. Meila. "Comparing clusterings - An axiomatic view," in Proceedings of the 22nd International Conference on Machine Learning, 2005, pp. 577-584.

[18]   J. Freixenet, X. Munoz, D. Raba, et al. "Yet another survey on image segmentation: region and boundary information integration," in Proceedings of European Conference on Computer Vision, 2002, pp. 408-422.

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 6, 2012, IJIP appears in 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
- Photographic emulsions
- Prepress and printing technologies
- Remote image sensing

- 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
- Photopolymers
- Protocols for packet video
- Retrieval and multimedia

- Storage and transmission
- Video coding algorithms and technologies for ATM/p

CALL FOR PAPERS

Volume: 6 - **Issue: 4** - August 2012

i. Paper Submission: May 31, 2012   **ii. Author Notification:** July 15, 2012

iii. Issue Publication: August 2012

CONTACT INFORMATION