Volume 6 • Issue 4 • August 2012

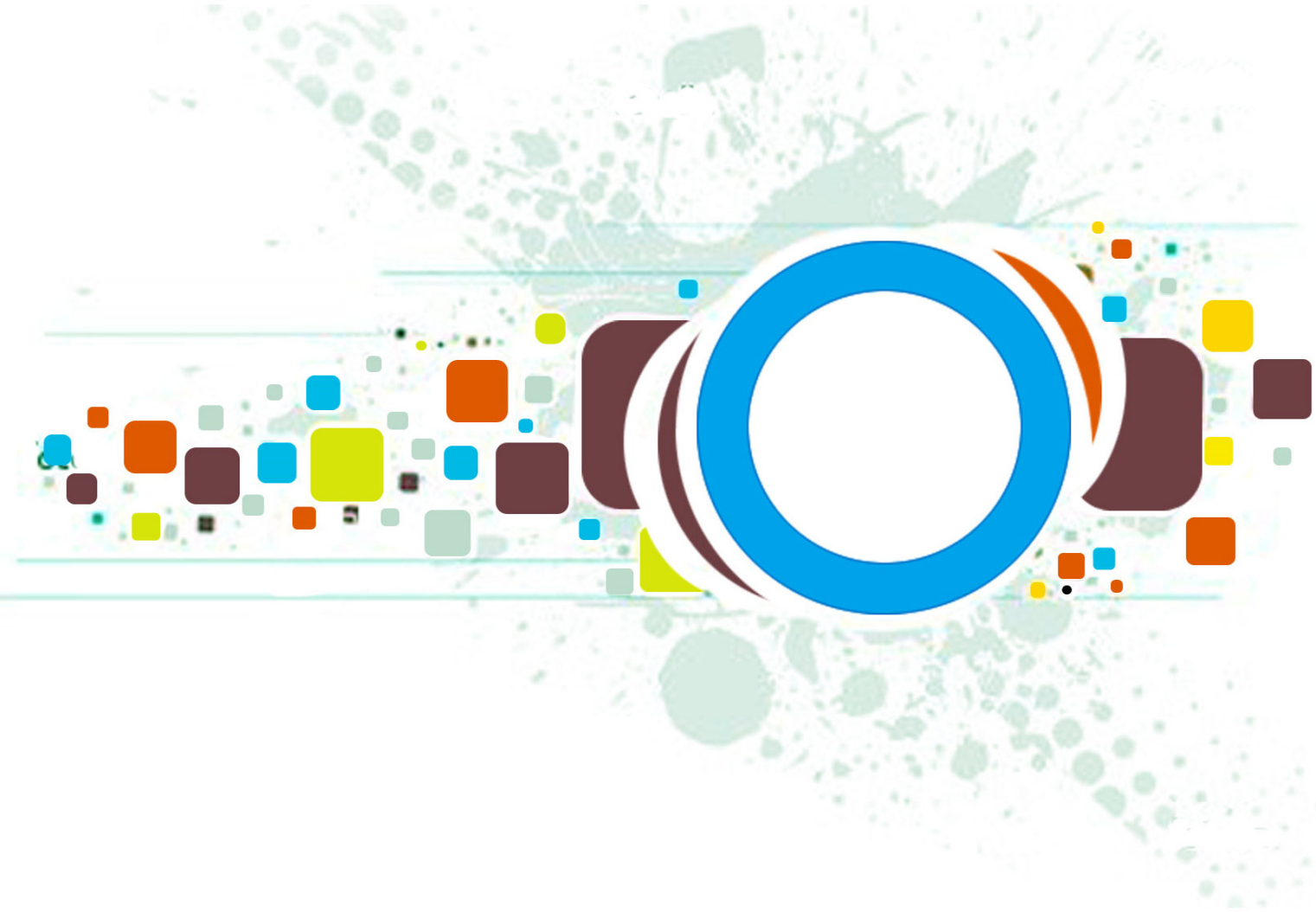
Editor-in-Chief
Professor Hu, Yu-Chen

INTERNATIONAL JOURNAL OF

IMAGE PROCESSING (IJIP)

ISSN : 1985-2304

Publication Frequency: 6 Issues Per Year



CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

VOLUME 6, ISSUE 4, 2012

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-2304

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

IJIP Journal is a part of CSC Publishers

Computer Science Journals

http://www.cscjournals.org

INTERNATIONAL JOURNAL OF IMAGE PROCESSING (IJIP)

Book: Volume 6, Issue 4, August 2012
Publishing Date: 10-08-2012
ISSN (Online): 1985-2304

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

Published in Malaysia

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

CSC Publishers, 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 *Fourth* 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. Started with Volume 6, 2012, IJIP appears with more focused issues. Besides normal publications, IJIP intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

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

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

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

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

Editorial Board Members
International Journal of Image Processing (IJIP)

EDITORIAL BOARD

TABLE OF CONTENTS

Volume 6, Issue 4, August 2012

Pages

A novel Super Resolution Algorithm Using Interpolation and LWT Based Denoising Method

Sapan Naik, Asst. Professor *sapan_say@yahoo.co.in*
Department of Computer Science and Technology,
Uka Tarsadiya University, Maliba Campus,
Tarsadi - 394 350, India

Viral Borisagar, Asst. Professor *viralborisagar@yahoo.com*
Department of Computer Engineering,
Government Engineering College, Sector 28,
Gandhinagar, India

Abstract

Image capturing technique has some limitations and due to that we often get low resolution(LR) images. Super Resolution(SR) is a process by which we can generate High Resolution(HR) image from one or more LR images. Here we have proposed one SR algorithm which takes three shifted and noisy LR images and generate HR image using Lifting Wavelet Transform(LWT) based denoising method and Directional Filtering and Data Fusion based Edge-Guided Interpolation Algorithm.

Keywords: Super Resolution, Interpolation, Wavelet Lifting Scheme

1. INTRODUCTION

Processing power limitations and channel capabilities are some of the factors because of which images are transmitted at low bit rates and down sampled and due to this reason we get LR compressed images. Computational process known as super resolution(SR) image reconstruction is used to reconstruct HR image from one or more noisy, blurred and down sampled LR images.

For the same scene we can have different images with different looks. Due to capturing technique images are aliased as well as shifted with sub pixel precision[14]. The basic idea for increasing the spatial resolution in SR image reconstruction technique is the availability of several LR images captured from the same scene. Many different methods like nonuniform interpolation, frequency domain approach, regularization and projection onto convex set [13,16] are available for doing Super Resolution.

Related problem to SR called Image restoration is one of the famous area in image processing applications in which we recover degraded images(remove noise, blur etc. from source image)[2,3,15]. The main difference between SR technique and restoration is that restoration does not change size of the image while in SR we do restoration as well as increase the size of image. Super Resolution is very useful in many practical cases like video applications, medical imaging and satellite imaging. SR is also useful in Synthetic zooming of region of interest (ROI) which is used in surveillance, forensic, scientific, medical, and satellite imaging.

In this paper we have presented one SR algorithm in which we have taken three LR images which are shifted and noisy. First we estimate shift between images using shift estimation algorithm, using these information next we align all three images and fuse them so we get single image. On this image we have applied LWT based denoising method for removing noise. Finally using Directional Filtering and Data Fusion based Edge-Guided Interpolation Algorithm we can get SR image. We have used wavelet lifting based denoising algorithm because implementation of lifting scheme is easy. It is fast as it can be computed in linear time and requires less memory. We can improve image quality by adding detail information and achieve high compression ratio too.

Paper is organized as follows. In section II we have briefly described shift estimation and fusion method, Section III provides wavelet lifting scheme for HAAR wavelet and denoising algorithm based on HAAR LWT, in section IV we have presented interpolation algorithm. We have presented our proposed algorithm and experiment results in section V. Finally section VI contain conclusion.

2. SHIFT ESTIMATION AND FUSION METHOD

Here we have considered RGB images. It is stored as [m,n,1:3] array of class double. Each color component contains floating-point values in the range of 0 to 1. Each row contains the red, green, and blue components of a single color. Array of RGB image can be of class uint16, uint8 or double. Pixel with color component (1,1,1) is displayed as white while (0,0,0) is displayed as black. The third dimension of the data array contains three color components for each pixel. As shown in figure, the red, green, and blue color components of the pixel (9,7) are stored in RGB(9,7,1),RGB(9,7,2), and RGB(9,7,3) respectively[5].

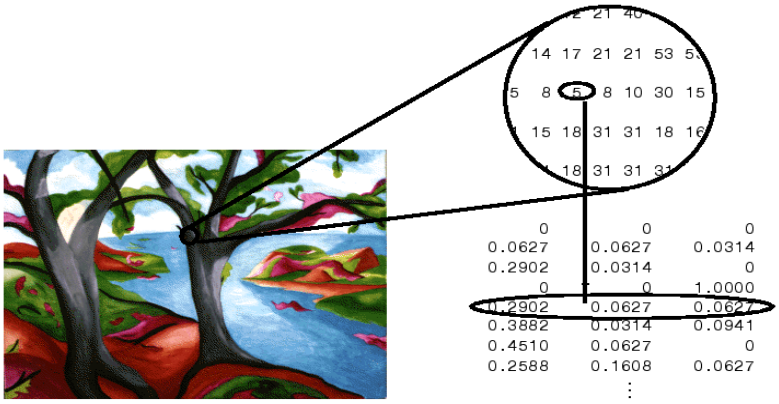


FIGURE 1 : RGB image[5]

2.1 Shift Estimation Method

For doing super resolution, first we need to align LR images and for that we have to do image registration. We can align two or more images of the same scene using registration (Images are taken from different viewpoints or taken at different times)[4,8]. From available LR images, we consider one image as reference image (base image) to which we can compare other input images. Main goal of registration is to do alignment of input images with the reference image by applying a spatial transformation.

We have considered three shifted images so our first task is to estimate shift between these images in both x and y coordinates. For shift estimation (here we have not consider rotation in images), we have used A Frequency Domain Approach to Registration presented in [9] by Patrick Vandewalle. In Fourier domain, shift between images can be expressed as

$$F_2(x) = e^{j2\Pi xT\Delta s} F_1(x)$$

So we need to find out Δs only. Here we have briefly presented shift estimation algorithm.

Step 1 : Make input images circularly symmetric.
Step 2 : Compute Fourier transform of all LR images.
Step 3 : Consider one image as reference image.
Step 4 : Compute phase difference between reference image and other images.
Step 5 : For all frequencies, write linear equation with unknown slope Δs.
Step 6 : Find shift parameter Δs.

2.2 Fusion Method

For doing fusion of three images, we have consider each pixel of the images separately. We fetch R, G and B component of each pixel from all three images and fuse all R,G and B component separately. For example three images are i1, i2 and i3, then consider pixel (3,5) of all images. We find mean value of R component of i1(3,5),i2(3,5) and i3(3,5), Same way for G and B component. From all these three value we get single pixel (3,5) in fused image. Below we have shown the code for doing fusion of three images.

```
        for i=1:m
        for j=1:n

        R = [i1(i,j,1) i2(i,j,1) i3(i,j,1)];
        G = [i1(i,j,2) i2(i,j,2) i3(i,j,2)];
        B = [i1(i,j,3) i2(i,j,3) i3(i,j,3)];

        fused(i,j,1)= mean(R);
        fused(i,j,2)= mean(G);
        fused(i,j,3)= mean(B);
        end
end
```

we have used mean value for fusion as shown above(we can take median value also but it takes lot of time for computation) and we have taken care of shift between the images that we have found initially.

3. WAVELET LIFTING SCHEME

Next we have to apply denoising method on fused image. For that we have used HAAR lifting wavelet Transform. For signal decomposition, we can use analysis filter bank which consist of low pass and high pass filters at each decomposition stage and split signal into two bands. The low pass filter fetch the coarse information(corresponds to an averaging operation) while high pass filter fetch detail information(corresponds to a differencing operation) of the signal. Finally the filtering operation's output is decimated by two[6,11].

For two-dimensional transform, the image is filtered along the x-dimension using low pass and high pass analysis filters and decimated by two. Then it is followed by filtering the sub-image along the y-dimension and decimated by two. Finally, the image has been split into four bands denoted by LL, HL, LH, and HH, after one level of decomposition[12,17]. The LL band is again subject to the same procedure. This process of filtering the image is called pyramidal decomposition of image. This is depicted in Fig. 2. The reconstruction of the image can be carried out by reversing the above procedure and it is repeated until the image is fully reconstructed.

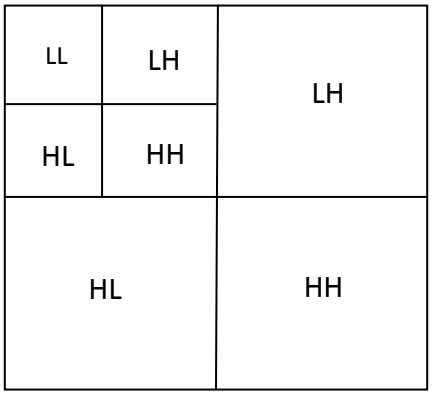


FIGURE 2 : One level Decomposition of image

3.1 Lifting Scheme Haar Transform[1]

The wavelet lifting scheme is a method for decomposing wavelet transforms into a set of stages. The forward lifting wavelet transforms has three steps, split step, predicate step and update step. Split step divides the data set into odd and even elements. The predict step uses a function that approximates the data set. The difference between the approximation and the actual data replaces the odd elements of the data set. The even elements are left unchanged and become the input for the next step in the transform. The update step replaces the even elements with an average. The simplest version of a forward wavelet transform expressed in the lifting scheme is shown in figure 3.

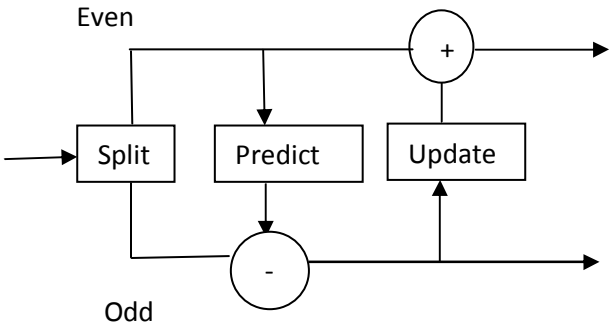


FIGURE 3 : Forward wavelet lifting Scheme[18]

In HAAR forward wavelet lifting Scheme, odd elements are equal to even elements in prediction step. The odd element are replaced by the difference of predicted value and actual value of odd element. For iteration y and element x, the new odd element, y+1,x would be

$$\text{odd } y+1, x = \text{odd } y, x - \text{even } y, x$$

Even elements are replaced by average of even/odd pair in update step.

$$\text{even } y+1,x = (\text{even } y,x + \text{odd } x,y) / 2$$

Value of oddy,x element has been changed by the difference between this element and its even predecessor.

$$\text{odd } y,x = \text{even } y,x + \text{odd } y+1,x$$

We get average by substituting above,

$$\text{even } y+1,x = (\text{even } y,x + \text{even } y,x + \text{odd } y+1,x) / 2$$
$$\text{even } y+1,x = \text{even } y,x + (\text{odd } y+1,x / 2)$$

As in shown in below figure, for the next recursive step averages become the input.

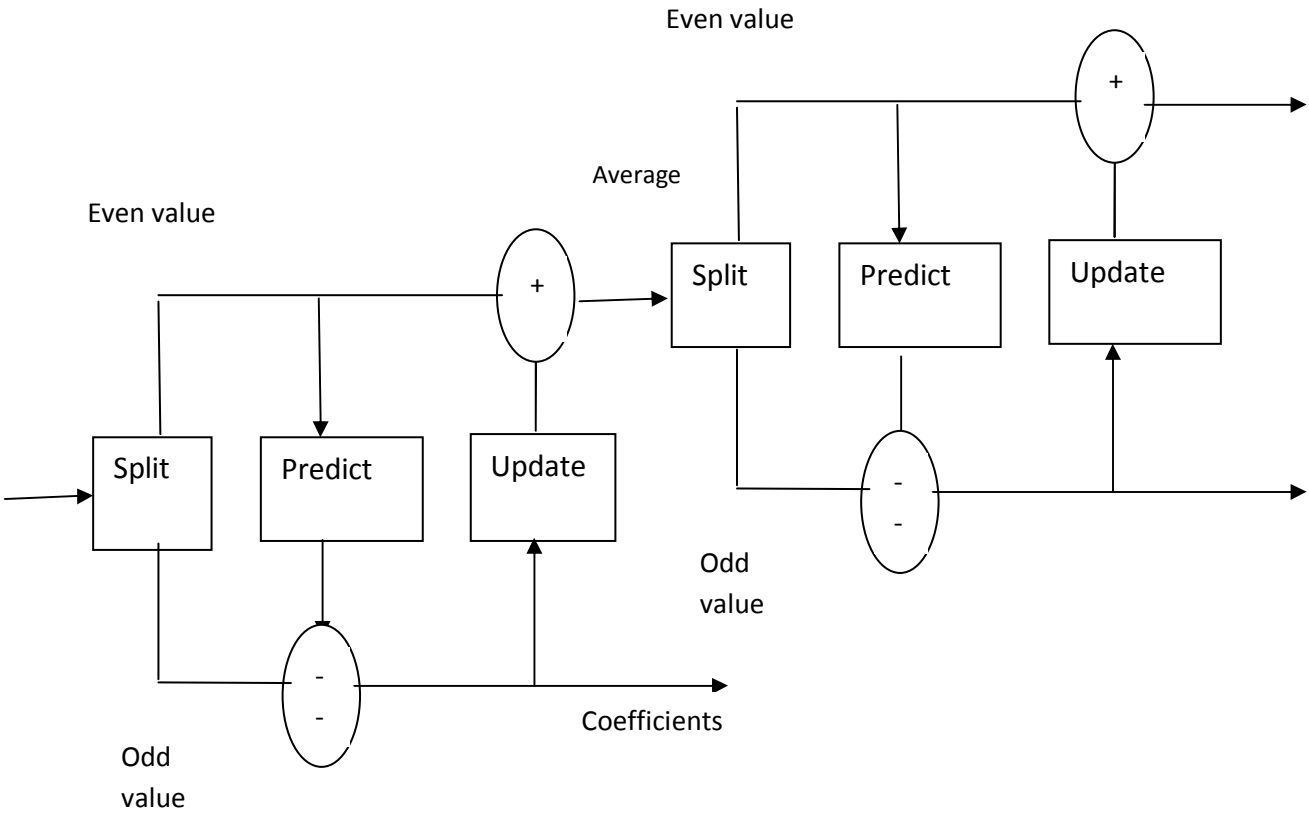


FIGURE 4 : Two steps in Forward wavelet lifting scheme[18]

Good thing in lifting scheme is that, inverse lifting is same as forward lifting. In HAAR wavelet lifting schemes what we have to do is, replace addition with substitution and visa versa and replace split step with merge step.

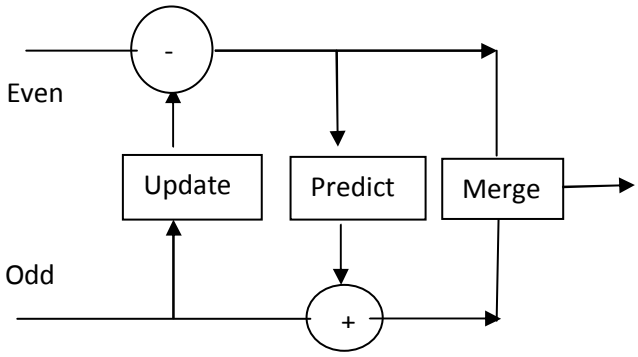


FIGURE 5 : Inverse lifting scheme [18]

3.2 Denoising using Lifting Wavelet Transform (LWT)

Denoising techniques are necessary to remove additive noise at the same time maintain the important details of the image. LWT based denoising method gives good result as wavelet transform contain large coefficients of image which represent detail of image at different resolution. Two methods are available for denoising. i)Hard Thresolding ii) Soft Thresolding[1].

HARD THRESHOLDING : I(P,T) = P if |P| > T and I(P,T) = 0 if |P| < T
SOFT THRESHOLDING : I(P,T) = sign(P) * max(0, |P| - T)

Where T is the threshold level, P is the input subband and D is the denoised band.

In LWT image is decomposed into four bands(LL,LH,HL,HH). For our denoising algorithm we only consider LH,HL and HH subbands because LL subband contains main information about the image while main noise is present in other three subbands. we have used the Median Absolute Deviation (MAD) to calculate noise level.

σ = median |Si,j| /0.6745 where Si,j = LH,HL,HH

Threshold value is calculated by

T = σ – (|Harmonic Mean – Geometric Mean|)

Where Harmonic Mean = M2/ $\sum_{i=1}^{M}\sum_{j=1}^{M} 1/g(i,j)$

And Geometric Mean = [$\prod_{i=1}^{M}\prod_{j=1}^{M} g(i,j)$] (1/M2)

So the procedure is, from LH subband calculate noise level(σ) than find threshold value(T) for that subband and finally apply soft thresolding method to get denoise LH subband. Apply same procedure for HL and HH subband.

4. INTERPOLATION ALGORITHM

In interpolation method main issue is to find out information of missing pixels from neighboring pixels as shown in figure.

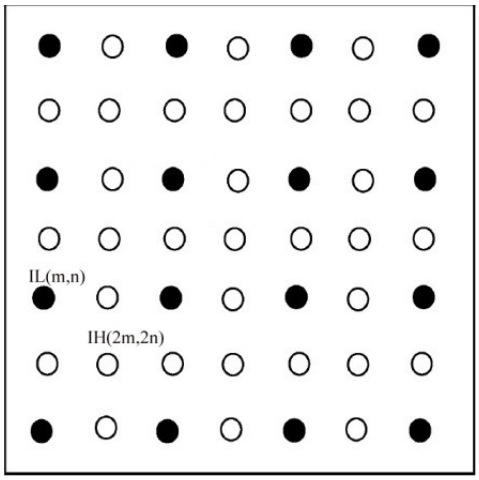


FIGURE 6 : Interpolation Image Structure

Interpolation method for HR image reconstruction suffers from aliasing problem if signal of LR image is down sampled and exceeds Nyquist sampling limit. In spatial locations our human visual system is very sensitive to the edges in image so it is important to suppress interpolation artifacts at the same time maintaining the sharpness and geometry of edges.

For interpolation process edge direction is very important and that's why we have use An Edge-Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion presented in [7]. We can use wavelet based Interpolation method presented in [10] also to do super resolution. For edge information they have partition pixels into two directional and orthogonal subsets. Directional interpolation is made for each Subset and two interpolated values are fused. Algorithm presented in [7] work for gray scale images only so we have done some modification so that it will work for RGB images. As shown below, We have stored each R,G and B components of one image into three

different images of two dimension (same as gray scale image) and give that as a input to original algorithm. Finally we have merged all three output arrays into single RGB image.

```
for i=1:m
for j=1:n
R(i,j)=Input(i,j,1);
G(i,j)=Input(i,j,2);
B(i,j)=Input(i,j,3);
end
end


for i=1: (2*m)
for j=1: (2*n)
RGB(i,j,1)=Output(i,j);
end
end      % same way RGB(i,j,2)and RGB(i,j,3)is achieved
```

5. PROPOSED SUPER RESOLUTION ALGORITHM

Here we have presented our proposed algorithm.

1. Take three shifted, low resolution and noisy images.
2. Apply shift estimation algorithm as explained above and find shift in x and y coordinates and align all three images.
3. Apply Gaussian filter to remove some noise.
4. Fused all three images using the fusion method explained and get a single image.
5. Apply HAAR Wavelet Lifting Scheme on this single image(we will get four subbands LL,LH,HL and HH).
6. Apply LWT based denoising method on LH,HL and HH subbands.
7. Apply inverse lifting scheme to recover image with less noise.
8. Get super resolution image by applying interpolation method explained in above section.

For experiment, we have generated three shifted, low resolution and noisy images from single HR image. We have performed experiment on the computer with configuration of Intel i3 processor, 4GB RAM and 512MB NVidia graphics card. For performance evaluation of algorithm, we have considered PNSR ratio and visual quality as parameter. The PSNR is defined as: $PSNR = 20 \cdot \log 10(MAX\ i / \sqrt{MSE})$. We have shown single input image instead of three as other input images are same except little shift in x and y coordinates.

6. CONCLUSION

SR image reconstruction is very important in many practical applications. We have taken LWT base denoising method as lifting schemes are easy to implement, fast ,require less memory and inverse lifting has same complexity as forward lifting. Interpolation method that we have taken is also gives good result as it works on edges as well. In our future studies, we would like to make this algorithm faster for practical use and want to use different lifting scheme and interpolation methods.

7. REFERENCES

[1] A.Jensen, A.la Courharbo, " Ripples in Mathematics : The Discrete Wavelet Transform", springer,2001.

[2] Ashwini M. Deshpande, Suprava Patnaik, Comparative Study and Qualitative-Quantitative Investigations of Several Motion Deblurring Algorithms,2nd International Conference and workshop on Emerging Trends in Technology (ICWET) 2011,(IJCA),pg 27-34.

[3] Ayers G. R., and Dainty J. C. Iterative blind deconvolution methods and its applications. Optics Letter, vol. 13, no. 7, July 1988.

[4] E.D. Castro, C. Morandi, "Registration of translated and rotated images using finite Fourier transform" , IEEE Transactions on Pattern Analysis and Machine Intelligence 700–703,1987.

[5] Image Types :: Displaying Bit-Mapped Images : (MATLAB®),http://www.mathworks.in/help/techdoc/creating_plots/f2-10709.html

[6] James S. Walker, "Wavelet-based Image Compression" , Sub-chapter of CRC Press book: Transforms and Data Compression.

[7] Lei Zhang, Xiaolin Wu, "An Edge-Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 8, AUGUST 2006.

[8] Patrick Vandewalle, Luciano Sbaiz, Sabine SÄusstrunk, Martin Vetterli," Registration of aliased images for super-resolution imaging",

[9] Patrick Vandewalle, Sabine Susstrunk,Martin Vetterli," A Frequency Domain Approach to Registration of Aliased Images with Application to Super-resolution", Hindawi Publishing Corporation,EURASIP Journal on Applied Signal Processing,Volume 2006, Article ID 71459, Pages 1–14.

[10] Ping-Sing Tsai, Tinku Acharya," Image Up-Sampling Using Discrete Wavelet Transform", 9th JCIS 2006, 8 to 11 October 2006.

[11] R.Sudhakar, Ms R Karthiga, S.Jayaraman, "Image Compression using Coding of Wavelet Coefficients – A Survey", ICGST-GVIP Journal, Volume (5), Issue (6), June 2005, pp.25-38.

[12] Rohit Arora, Madan lal Sharma, Nidhika Birla, "An Algorithm for Image Compression Using 2D Wavelet Transform" , International Journal of Engineering Science and Technology (IJEST) ISSN : 0975-5462 Vol. 3 No. 4 Apr 2011 pp. 2758 – 2764.

[13] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical review," IEEE Signal Processing Mag., vol. 20, pp. 21–36, May 2003.

[14] S. Chaudhuri, Ed., *Super-Resolution Imaging*. Norwell, MA: Kluwer, 2001.

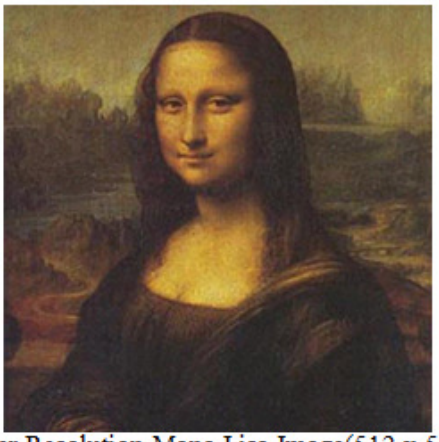
[15] S.P. Kim, N.K. Bose, and H.M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Trans.Acoust., Speech, Signal Processing*, vol. 38, pp. 1013-1027, June 1990.

[16] Sina Jahanbin, Richard Naething, "Super-resolution Image Reconstuction Performance", May 16, 2005

[17] Vinay U. Kale, Nikkoo N. Khalsa, "Performance Evaluation of Various Wavelets for Image Compression of Natural and Artificial Images" ,International Journal of Computer Science & Communication, Vol. 1, No. 1, January-June 2010, pp. 179-184.

[18] www.bearcave.com/misl/misl_tech/wavelets/lifting/basiclift.html

Mona Lisa(256 x 256)



Super Resolution Mona Lisa Image(512 x 512)

Lena(256 x 256)

Super Resolution Lena Image(512 x 512)

Butterfly(120 x 120)

Super Resolution Butterfly Image(240 x240)

African face(120 x 120)

Super Resolution African face Image(240 x240)

PSNR(dB)			
Mona Lisa	Lena	Butterfly	African Face
28.7684	33.6667	31.2543	29.8879

Figure 7 : Experiment Results

Blind Source Separation Using Hessian Evaluation

Jyothirmayi M *mjyothiprasanna@gmail.com*
Department of Instrumentation
M.S.Ramaiah Institute of Technology,
Karnataka, Bangalore-560054, India

Elavaar Kuzhali S *kuzhalisubbiah @yahoo.com*
Department of Instrumentation
M.S.Ramaiah Institute of Technology,
Karnataka, Bangalore-560054, India

Sethu Selvi S *selvi@msrit.edu*
HOD, Department of Electronics and Communication
M.S.Ramaiah Institute of Technology,
Karnataka, Bangalore-560054, India

Abstract

This paper focuses on the blind image separation using sparse representation for natural images. The statistics of the natural image is based on one particular statistical property called sparseness, which is closely related to the super-gaussian distribution. Since natural images can have both gaussian and non gaussian distribution, the original infomax algorithm cannot be directly used for source separation as it is better suited to estimate the super-gaussian sources. Hence, we explore the property of sparseness for image representation and show that it can be effectively used for blind source separation. The efficiency of the proposed method is compared with other sparse representation methods through Hessian evaluation.

Keywords: Blind Source Separation, Infomax Algorithm, Sparseness, Hessian Evaluation.

1. INTRODUCTION

Blind source separation (BSS) is the process of extracting the underlying sources from the mixed images or observed signals, and since no *a priori* knowledge of the mixed sources is known or very little information is available, it is called *blind*. Independent component analysis (ICA) is most widely used technique to solve the blind source separation [1-7, 20] problem. BSS is based on the assumptions that source signals are independent and non gaussian. The problem of BSS can be stated as follows.

Given M linear mixtures of N original images mixed via a $M \times N$ mixing matrix, these mixtures can be represented as a linear equation of the form

$$X = AS \quad (1)$$

S is the original sources to be extracted, X is the observation random vector, and A is a full rank $M \times N$ mixing matrix. The task is to estimate the mixing matrix A and then recover the source images S. The complexity of source separation is based on the dimensions M and N. If $M=N$, the mixing matrix A is a square matrix and the original sources can be estimated by a linear transformation. If $M>N$, the mixing matrix is a full rank over complete matrix and the original sources can be estimated using linear transformation involving pseudoinverse matrix.

After estimating the matrix A using BSS algorithm, the inverse of A is computed, $W=A^{-1}$ called separator and the independent sources are obtained simply by

$$U = WX \quad (2)$$

There are several methods [5], [6] to separate the independent components from the original data. Bell and Sejnowski [7] developed a neural learning algorithm for separating the statistically independent components of a dataset through unsupervised learning. The algorithm is based on the principle of maximum information transfer between sigmoid neurons. The features obtained were not very interesting from a neural modeling viewpoint, and the mixing matrix reduces the sparsity of the original images which motivates us to find better models like exploiting the property of sparseness, Zibulevsky et al.(2002).The advantage of this property is that, two or more sources being active at the same time is low. Thus, sparse representations provide good separability as most of the energy is confined to a single source, at a given time instant.

Sparse coding is a method for finding suitable representation of data in which the components are rarely active. It has been shown [9, 10-17] that this sparse representation can be used to solve the BSS problem. ICA algorithms i.e., FASTICA uses kurtosis as a sparseness measure and since kurtosis is sensitive to the outliers as it applies more weight on heavy tails rather than on zero, the sparseness measure is mostly unreliable. When the sources are locally very sparse the matrix identification algorithm is much simpler. Sparse representation of image matrix can be performed using clustering algorithms [12, 13, 15], Gradient Ascent Learning [14], Laplacian Prior [16, 7], wavelet [17] , Finite difference method (FDM)[29] etc.

In this paper, the sparsity representation of the image mixture is exploited using Hessian transformation and the sparseness is measured using l_0 norm Donoho [26]. The Hessian transform method provides a powerful approach to solve differential equations, non-linear problems and is widely used in the field of applied sciences. The proposed BSS algorithm is more efficient and leads to improved separation quality which is measured in terms of Signal to Noise ratio (SNR), Mean squared error (MSE), Structural Similarity Index Measure (SSIM) and estimating the Mutual information (MI) of the separated images with original source images.

The rest of the paper is organized as follows: Section 2 deals with the method used for sparse representation of data, sparse measure using l_0 norm and the algorithm used for separation. Section 3 illustrates the results where we compare the separated images with original images and Section 4 gives the conclusion.

2. SPARSE REPRESENTATION

In order to represent the sparsity of the original images that solves the BSS problem, several methods are proposed [5, 6, 12, 13, 14, 15, and 21]. A simpler and efficient method to make the image sparse is by calculating the Hessian matrix of the image. The problem of estimating the Hessian matrix is to derive the second order derivative of a given image. The equation for the second order can be derived using Taylor series expansion [23].

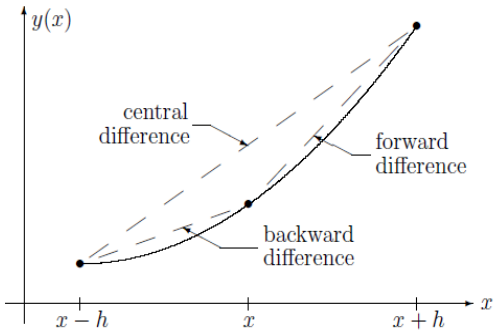


FIGURE 1: Finite difference approximations to derivatives

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(x) + O(h^4) \quad (3)$$

Finding the first derivative and second derivative

$$y'(x) = \frac{y(x+h) - y(x)}{h}$$

$$y''(x) = \frac{y'(x+h) - y'(x)}{h}$$

$$y''(x) = \frac{y(x+2h) - 2y(x+h) - y(x)}{h^2} \quad (4)$$

Similarly, by replacing h by –h in the Taylor series of Equation (3) and deriving the second derivative, the backward difference equation is obtained as ,

$$y''(x) = \frac{y(x) - 2y(x-h) + y(x-2h)}{h^2} \quad (5)$$

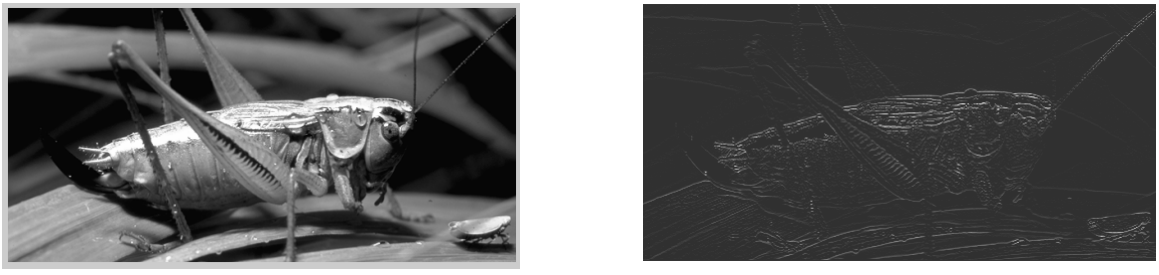
The average of the two Equations (4) and (5) results in a central difference approximation (Hessian), of the form

$$y''(x) = \frac{y(x+h) - 2y(x) + y(x-h)}{h^2} \quad (6)$$

as represented in Figure 1.Hence for the given image matrix X, the Hessian matrix obtained will be of the form

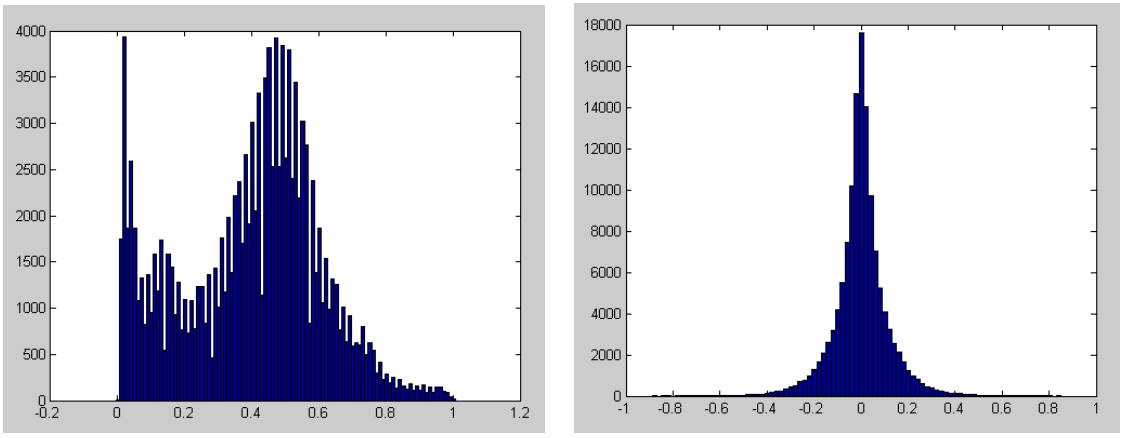
$$H(x) = \begin{bmatrix} \frac{\partial^2 x}{\partial x_1^2} & \frac{\partial^2 x}{\partial x_1 \partial x_2} & .... & \frac{\partial^2 x}{\partial x_1 \partial x_n} \\ \frac{\partial^2 x}{\partial x_2 \partial x_1} & \frac{\partial^2 x}{\partial x_2^2} & .... & \frac{\partial^2 x}{\partial x_2 \partial x_n} \\ \vdots & \vdots & .... & \vdots \\ \frac{\partial^2 x}{\partial x_m \partial x_1} & \frac{\partial^2 x}{\partial x_m \partial x_2} & .... & \frac{\partial^2 x}{\partial x_n^2} \end{bmatrix} \quad (7)$$

The Hessian evaluation method can be used for sparse representation of the image since it acts as an edge detector which provides a two-level image, the edges and the homogeneous background. By using this method, the separation matrix estimated to separate the image mixture is similar to that of the method used (FASTICA) for image separation. In Figure 2, shows the natural image as well as the image obtained from the above method. Figure 3, represents the histograms of the original and the Hessian transformed image from which the sparsity of the latter can be inferred.



(a) (b)

FIGURE 2: a) Original Image and **b)** Hessian Transformed Image



(a) (b)

FIGURE 3: (a) Original Image Histogram and **(b)** Hessian Transformed Image Histogram

2.1 Sparsity Measure

Sparsity measures are used to calculate a number that describes the sparsity of an image vector $C = [c_1 c_2 .... c_N]$. The most common sparsity measure is the ℓ_p norm defined by

$$\left\| \vec{C} \right\|_p = \left(\sum_j C_j^p \right)^{1/p} \quad \text{for } 0 \le p \le 1 \tag{8}$$

The simplest is the l_0 norm that calculates the number of non-zero coefficients in the image vector

$$\left\| \vec{C} \right\|_0 = \#\{c_j \neq 0,\ j = 1....N\}$$

Since this traditional method is unsuited for many practical scenarios [15,24,25],a modified approximation method suggested by Donoho [26] is being used where a threshold is used .

$$\left\| \vec{C} \right\|_0 = \#\{c_j \neq \varepsilon,\ j = 1....N\}, \tag{9}$$

where ε is some threshold value.

Under this measure, the sparse solution is obtained by finding the number of non-zero elements in a block. Bronstein et al. [5], discusses that the use of block partitioning, which is the natural way of handling mixing coefficients in an image that vary spatially and also gives a better refinement to the sparseness. Hence the block having the maximum sparse is selected to estimate the separation matrix *W* Equation (2). Figure 4 shows sparseness measure values using l_0 norm for different sparse functions.

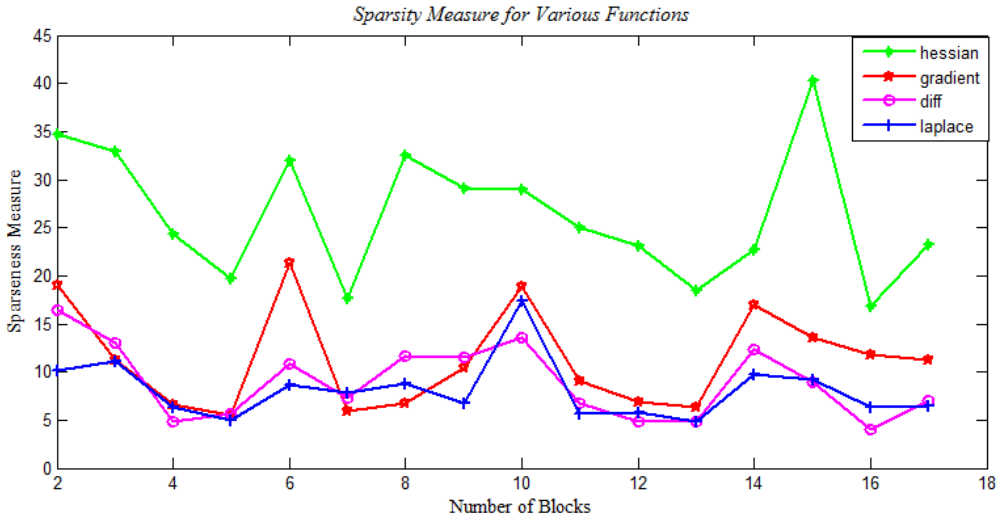


FIGURE 4: Sparsity measure plot obtained for different functions

2.2 Algorithm

Figure 5 shows the flow diagram for the representation of sparse matrix using second order differentiation (Hessian) and the selection of the best block by evaluating the quality factor for the generation of the separation matrix. Infomax ICA algorithm [10] is used for solving BSS problem.

Steps for Hessian Transformation and selection of best sparse block for BSS

1. Normalize and mean removed data is formed from N images.
2. Mixed images are obtained by linearly mixing with a random matrix.
3. The second order derivative (Equation 6) is applied for each of the mixed images to get sparse images. The component in the X direction and Y direction is considered.(Figure 5)
4. The sparse images along X and Y directions are divided into blocks of equal size.
5. The blocks having same spatial location are considered for evaluation of the sparseness (l_0 norm).
6. The blocks having maximum sparseness is considered for estimating the separation matrix using Infomax algorithm.

2.3 ICA-BSS Algorithm

The block having maximum sparseness can be represented as a linear equation of the form

$$X = \sum_{i=1}^{n} a_i s_i = AS \qquad (10)$$

where *A* is the unknown fixed matrix *M*×*N* of full rank, called *mixing matrix* and its columns are known as *ICA basis vectors*. The fundamental problem of ICA is then to recover the underlying causes *S* from the mixture *X*. Since only *X* is observed the problem changes to estimating *W*.

$$S = (A^T A)^{-1} A^T X = WX \qquad (11)$$

In order to determine *W*, we use the Sejnowski infomax algorithm [10] that maximizes the information for ICA. A feedforward neural network Figure 6 is used to train the weights of training images. As shown in the Figure 6, *X* is the input to the neural network, the output U is the summed output of the weights connected to neuron and the input *X*.

$$u_i = \sum_{j=1}^{m} w_{ij} x_j \qquad \text{i.e} \qquad U = WX \tag{12}$$

ICA works by adjusting the unmixing coefficients of W in order to maximize the uniformity (entropy) of the distribution $Y = f(u)$ where f is cumulative density function (cdf). Therefore, if W is optimal, then Y will have maximum entropy and are therefore independent which ensures that the extracted sources are also independent. Hence the gradient update rule for the weight matrix, W used is as follows:

$$\Delta W \ \alpha \ \nabla_W H(Y) W^T W \ = (I + Y' U^T) W \tag{13}$$

where $Y' = (1 - 2Y_1)$

$$Y_i' = f_i''(U_i) / f_i'(U_i),$$

$$f_i(u) = 1 / (1 + e^{-u})$$

ICA finds a matrix *W*, such that the rows of $X = AS$ are statistically independent as possible. Hence we find the sources S=*WX*, where $W = A^{-1}$.

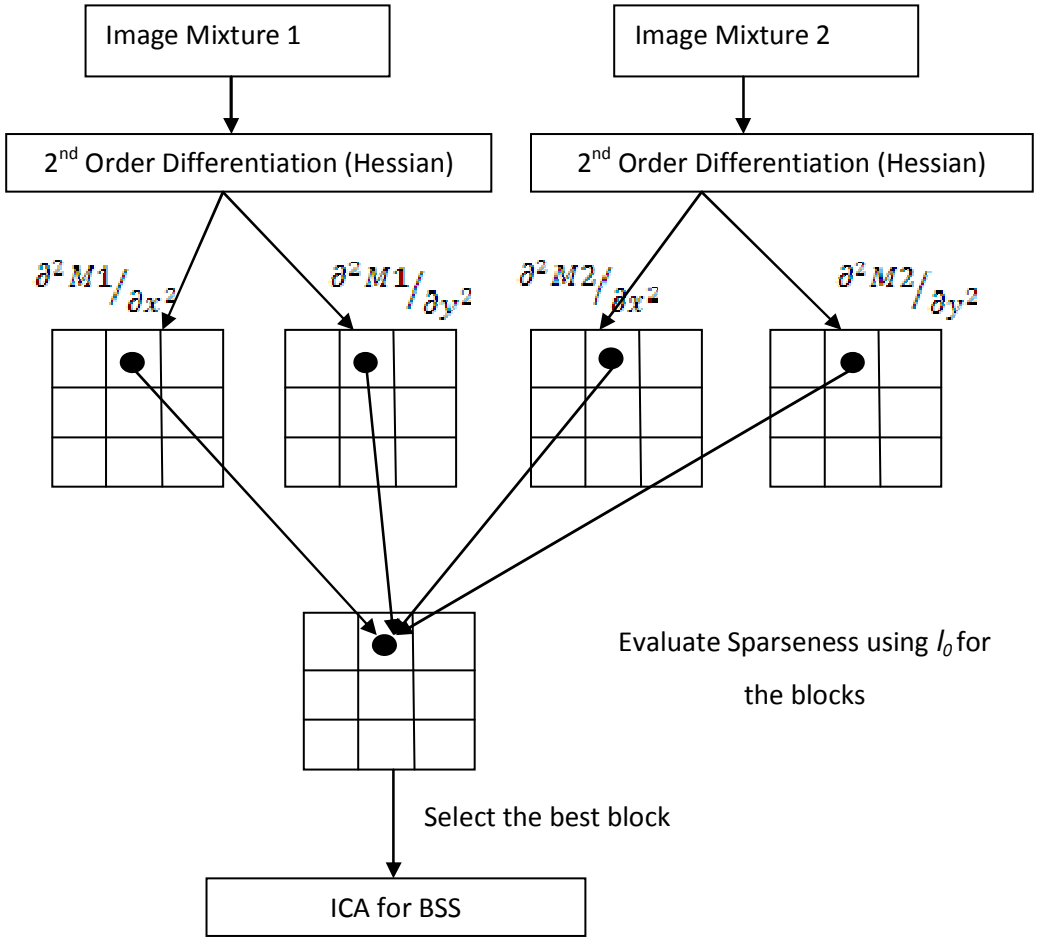


FIGURE 5: Block schematic describing the algorithm for BS

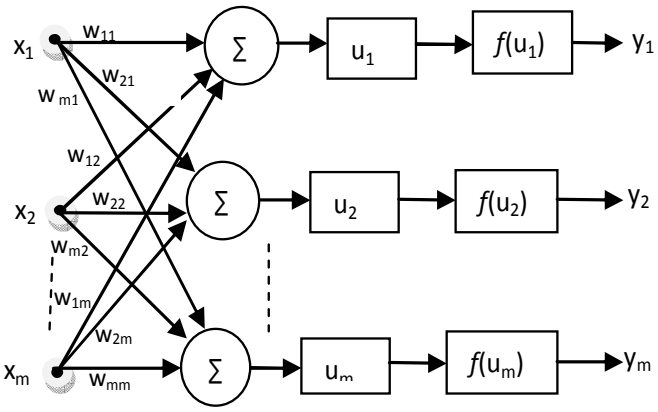


FIGURE 6:. Neural Network to train the weights

3. SIMULATION RESULTS

Simulation experiments are conducted to demonstrate the feasibility of the proposed BSS method. All simulations are carried on natural images[1] of dimension 256 x 512. The algorithms are developed on MATLAB environment. The images are mixed with a random matrix 4 x 4 (example Equation (14)) and the Hessian transform as in Equation (6) is applied on it to get sparse images. The resultant of Hessian derivative along the X direction and Y direction are considered.

$$M = \begin{bmatrix} -0.0716 & 0.3296 & -2.6350 & -0.3276 \\ -2.4146 & 0.5985 & 0.0281 & -1.1582 \\ -0.6943 & 0.1472 & -0.8763 & 0.5801 \\ -1.3914 & -0.1014 & -0.2655 & 0.2398 \end{bmatrix} \quad (14)$$

The sparse images are then divided into blocks of equal size 64 x 128 for which the l_0 norm, Equation (9), is applied to evaluate the sparseness measure. Figure 4 shows the sparseness measure values for different blocks and for different sparse representation methods. The blocks which has maximum value of the sparseness (e.g for hessian, block 15 from figure 4) is considered for evaluating the separation matrix. The separation matrix is obtained by using Infomax algorithm indicated in Equations (10-13). The results shown in Figure 7 and Figure 8 can be used as a subjective quality assessment. The histogram plot shows that the separated images are similar to original. Mean Squared Error (MSE), Signal to Noise Ratio (SNR) are used as objective image quality measures. The Structural Similarity Index Measure (SSIM) a well-known quality metric is used to measure the similarity between two images. It was developed by Wang et al. [16], and is considered to be correlated with the quality perception of the Human visual system (HVS). The Mutual Information between the extracted source and the corresponding original source is also considered as a quality metric. Here, 5000 samples (pixels) are randomly selected from the separated image and the original source to form pixel pairs. The Mutual Information between these 5000 pixel pairs was estimated using $I^{(1)}$ estimator described by Kraskov et.al.(2004) with k=2, (k recommended can be between 2 and 4 which is the nearest neighbor order)[27,28].

A comparison of proposed method with other existing sparse representation methods such as Gradient, Laplace [5] and Finite Difference Method [29] for assessing the amount of separation achieved is tabulated in Tables 1 to 4. The lower value for MSE as in Table 1 indicates, better

quality separation. The higher values of SNR, SSIM, MI in Table 2, 3 and 4a implies that the separated images are closer to original source images, whereas Table 4b, the low values corresponds to the comparison of extracted image with the wrong source image. The values in the last column of the Tables 1-4 show clearly that the hessian representation method has an advantage over other sparse representation methods.

[1] http://www.cis.hut.fi/projects/ica/data/images/

FIGURES	FDM	LAPLACE
Value 1	Value 2	Value 3
Value 4	Value 5	Value 6
Value 7	Value 8	Value 9

Figure	FDM	Laplace	Gradient	Hessian
S1	1.82	7.96	2.65	1.49
S2	0.83	0.009	2.32	0.065
S3	4.58	65.18	6.76	3.53
S4	0.38	18.87	1.25	1.31

TABLE 1: MSE values for the four separated images

Figure	FDM	Laplace	Gradient	Hessian
S1	38.54	23.10	36.89	39.37
S2	41.23	60.79	36.77	52.32
S3	34.82	23.29	33.12	33.28
S4	44.45	27.54	39.30	39.76

TABLE 2: SNR values (in dB) for the four separated images

Figure	FDM	Laplace	Gradient	Hessian
S1	0.8590	0.8570	0.9998	0.9979
S2	0.9883	0.9884	0.9994	0.9999
S3	0.9986	0.9986	0.9999	0.9981
S4	0.8739	0.8744	0.9931	0.9993

TABLE 3: SSIM values for the four separated images

Figure	FDM	Laplace	Gradient	Hessian
S1	5.33	2.9	5.15	5.4
S2	5.46	4.39	4.60	5.73
S3	4.63	2.85	4.13	4.97
S4	5.4	2.07	5.03	5.59

TABLE 4a: MI values for the four separated images with corresponding source image.

Figure	FDM	Laplace	Gradient	Hessian
S1	0.31	0.18	0.28	0.18
S2	0.13	0.09	0.11	0.10

TABLE 4b: MI values for the four separated images with other source image

4. CONCLUSIONS

For the solution to source separation, natural images are considered, as the natural scenario provides various mixtures of images. The proposed sparse representation method using block partitioning approach is more suitable for both spatially variant and invariant images. Experiments conducted on the natural image data set shows the efficiency of this approach and its advantages over previously-proposed sparse representation methods. We observe that there is improvement in MSE, SNR, MI and SSIM values. These values can be considered as objective evaluation for the image quality. Figure 8 shows the histogram for the original and separated images which clearly show that separated images are equal to the original images.

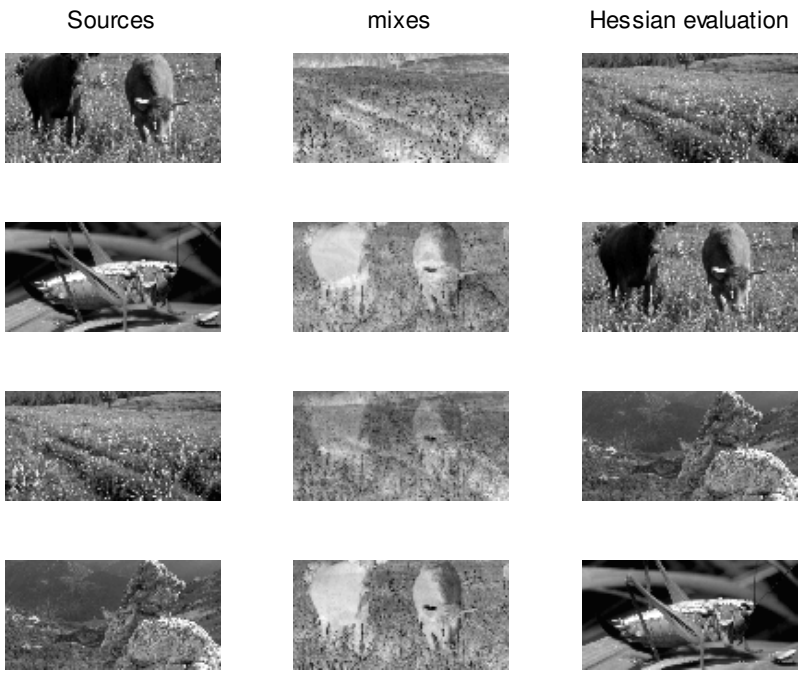


FIGURE.7: Original images, Mixed and Separated images for the proposed Method

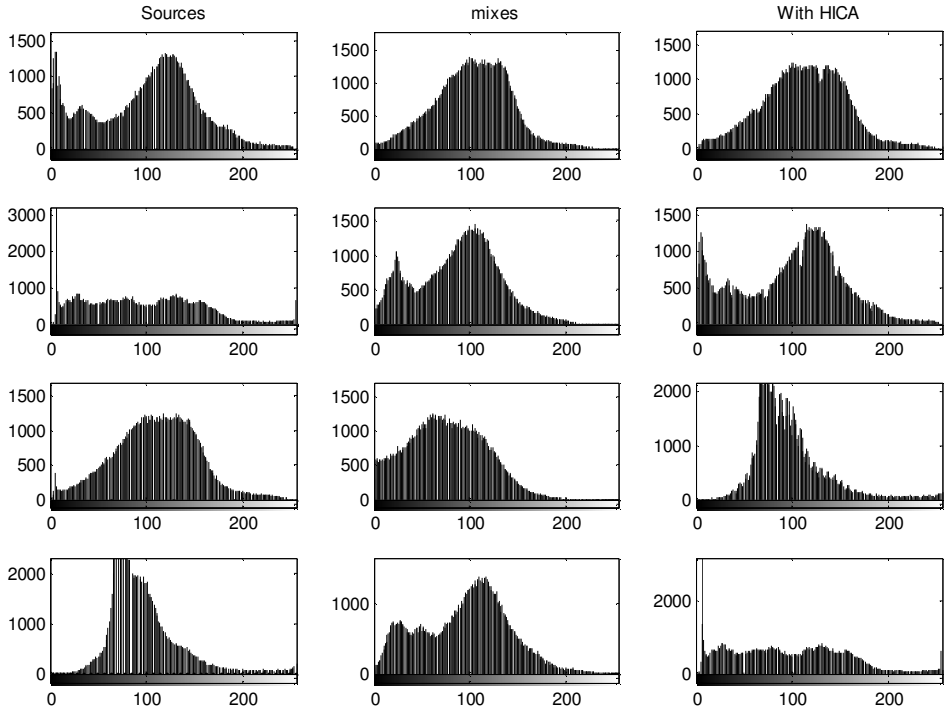


FIGURE 8: Histogram plot for original, mixed and separated images

5. REFERENCES

[1] Jutten.C, Herault. J.Blind separation of sources part I: "'A adaptive algorithm based on neuromimetic architecture", Signal Process, ,24:1-10,1991

[2] Girolami. M. "An alternative perspective on adaptive independent component analysis algorithms". Neural Computation, 10 (8) :2103-2114, 1998

[3] A. Hyvärinen and E. Oja., A Fast Fixed-Point Algorithm for Independent Component Analysis, Neural Computation, 9(7):1483-1492,1997

[4] Aapo Hyvärinen, Jarmo Hurri, Patrik O. Hoyer, Natural Image Statistics-A Probabilistics approach to early computational vision, springer 2009.

[5] M. M. Bronstein, A. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi, "Separation of reflections via sparse ICA," in *Proc. International Conference on Image Processing ICIP*, September 2003, vol. 1, pp. 313–316.

[6] M. Zibulevsky and B. A. Pearlmutter, "Blind source separation by sparse decomposition in signal dictionary," *Neural Computation*,vol. 13, no. 4, pp. 863–882, 2001.

[7] Pando Georgiev, Fabian Theis, and Andrzej Cichocki, "Sparse Component Analysis and Blind Source Separation of Underdetermined Mixtures" IEEE Transactions on Neural Networks, VOL. 16, NO. 4, July 2005

[8] P Comon. "Independent component analysis –a new concept".Signal Processing, 36:287-314, 1994.

[9] Hyvarinen A. "Independent Component Analysis:Algorithms and Applications". Neural Networks, 2000, 13:411-430.

[10] Bell, A., & Sejnowski, T. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6), 1129-1159(1995)

[11] Zibulevsky M., Pearlmutter B. A., Boll P., & Kisilev P. (2000) Blind source separation by sparse decomposition in a signal dictionary. In Roberts, S. J. and Everson, R. M. (Eds.), *Independent Components Analysis: Principles and Practice*, Cambridge University Press

[12] Yuanqing Li, Andrzej Cichocki, Shun-ichi Amari, Sergei Shishkin , *Sparse Representation and Its Applications in Blind Source Separation* NIPS-2003

[13] W. Souidene, A. A¨ıssa-El-Bey, K. Abed-Meriam, and A. Beghdadi, "Blind image separation using sparse presentation," in Proc. IEEE International Conference on Image Processing ICIP, September 2007, vol. 3, pp. 125–128.

[14] Hyvarinen A. "Survey on Independent Component Analysis". Neural Computing Surveys, 1999, 2:94-128.

[15] Yuanqing Li, Andrzej Cichocki, Shun-ichi Amari, Sergei Shishkin, Jianting Cao, Fanji Gu, *Sparse Representation and Its Applications in Blind Source Separation* Seventeenth Annual Conference. on. Neural Information Processing Systems 2003.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error measurement to structural similarity" IEEE Transactios on Image Processing, vol. 13, no. 1, Jan. 2004

[17] V.K.Ananthashayana and Jyothirmayi M, " Blind Source Separation Using Modified Gaussian Fastica", International Conference on Signal Processing, Communications and Networking, Sept 2009.

[18] J. K. Lin, D. G. Grier, and J. D. Cowan. "Feature extraction approach to blind source separation".In *IEEEWorkshop on Neural Networks for Signal Processing (NNSP)*, pages 398.405, 1997.

[19] T.-W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski. "Blind source separation of more sources than mixtures using overcomplete representations". *IEEE Signal Processing Letters*, 4(5):87.90, 1999.

[20] A. Jourjine, S. Rickard, and O. Yilmaz."Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures". In *IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP2000)*, volume 5, pages 2985.2988, June 2000.

[21] P. Bo_ll and M. Zibulevsky."Blind separation of more sources than mixtures using the sparsity of the short-time fourier transform". In *2nd International Workshop on Independent Component Analysis and Blind Signal Separation*, pages 87.92, Helsinki, Finland, June 19.20, 2000.

[22] M. Zibulevsky and B. A. Pearlmutter."Blind source separation by sparse decomposition in a signal dictionary",*Neural Computation*, 13(4):863.882, Apr. 2001

[23] Gordon D. Smith, 'Numerical solution to partial differential equation: finite difference methods', Oxford University Press, third edition, 2004.

[24] J. Karvanen and A. Cichocki, "Measuring sparseness of noisy signals,"in *Proc. ICA03*, 2003.

[25] Niall Hurley and Scott Rickard, "Comparing Measures of sparcity",in IEEE Transactions on Information Theory , VOL. 55, NO. 10, OCTOBER 2009

[26] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans.Inf. Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.

[27] A. Kraskov, H. Stögbauer and P. Grassberger, "Estimating Mutual Information", Physical Review E, vol. 69, pp. 066138, 2004.

[28] L. B. Almeida, "Separating a Real-Life Nonlinear Image Mixture", *Journal of Machine Learning Research*, vol. 6, pp. 1199–1232, 2005.

[29] Jyothirmayi M, Elavaar Kuzhali S and Sethu Sevi S, "Blind Image Separation using Forward Difference Method",Signal and Image Processing :An International Journal, vol2,No 4, pp121-127, Dec2011.

Unsupervised Building Extraction from High Resolution Satellite Images Irrespective of Rooftop Structures

Lizy Abraham
Dept. of Electronics & Communication. Engg. *lizytvm@yahoo.com*
LBS Institute of Technology for Women
Trivandrum, Kerala, India

Dr.M.Sasikumar
Head of the Department *drmsasikumar@ yahoo.com*
Marian Engineering College
Trivandrum, Kerala, India

Abstract

Extraction of geospatial data from the photogrammetric sensing images becomes more and more important with the advances in the technology. Today Geographic Information Systems are used in a large variety of applications in engineering, city planning and social sciences. Geospatial data like roads, buildings and rivers are the most critical feeds of a GIS database. However, extracting buildings is one of the most complex and challenging tasks as there exist a lot of inhomogeneity due to varying hierarchy. The variety of the type of buildings and also the shapes of rooftops are very inconstant. Also in some areas, the buildings are placed irregularly or too close to each other. For these reasons, even by using high resolution IKONOS and QuickBird satellite imagery the quality percentage of building extraction is very less. This paper proposes a solution to the problem of automatic and unsupervised extraction of building features irrespective of rooftop structures in multispectral satellite images. The algorithm instead of detecting the region of interest, eliminates areas other than the region of interest which extract the rooftops completely irrespective of their shapes. Extensive tests indicate that the methodology performs well to extract buildings in complex environments.

Keywords: PSMS Image, Rooftop Detection, Otsu's Thresholding, Area Analysis.

1. INTRODUCTION

Cartographic feature extraction systems can be categorized by the type of sensor data used. Some of the researches concentrate on the fusion of more than one data sources. These methods usually use the advantages of height information in 3-d data set. One common approach is to use more than one aerial or satellite images and getting the height information using photogrammetric calculations [1, 2, 3, 4]. Also usage of new technologies such as LIDAR (Light Detection And Ranging), which provides high vertical accuracy and high point density, becomes popular. Some of these make use of the fusion of LIDAR and satellite image [5, 6]. Another important category of the Cartographic feature extraction systems extracts objects by using monocular aerial or satellite imagery. Wei and Zhao [7] introduce an approach, where they first cluster of the satellite image using an unsupervised learning method and used the shadow information to verify the existence of building. Then, for each building boundary, Canny operator is used for extracting edges and finally the system detects lines using Hough transform. Mayungaa [8] works on an active contour model which is commonly known as a snake algorithm for a semiautomatic building extraction method. In this method the user has to click to the approximate center of each building; then the algorithm generates the border of this building. Jin and Davis [9] proposed an automated building-extraction strategy for high-resolution satellite imagery that utilizes structural, contextual, and spectral information. The system runs automatically without pre-classification or any training sets, although some initial algorithm parameters must be set by the user.

Recent researches in this area focus on automatic and unsupervised extraction of buildings. Akçay and Aksoy [10] proposed a method for unsupervised segmentation and object detection in high-resolution satellite images but the system performance varies depending on different rooftop structures. Aytekin and Erener [11] proposed an algorithm for automatic and unsupervised building extraction from urban environments. Better performance is ensured by the method but the major drawback is over detection. Use of local feature vectors and a probabilistic framework for the extraction of buildings having diverse characteristics and appearance is also discussed [12, 13]. Though the method is efficient the algorithm is not strictly unsupervised. Recently, a novel approach for automatic detection of buildings with a gable roof from very-high-resolution aerial images, covering particularly rural areas is proposed [14]. The method can be modified for other rooftop structures but a single algorithm is not sufficient for detection of buildings irrespective of rooftop structures.

Most of the works in the literature are either designed for specific applications or need some prior knowledge, such as human's interaction for the extraction of buildings. Recent works which are focussed on unsupervised and automatic detection techniques are mostly restricted to specific types of shapes or surface features. A complex urban environment includes various shapes and surface materials which make the detection process complicated and in many cases pixels belonging to roof tops of buildings may wrongly identify as road pixels because both have linear features. Depending on roof top structure the system performance varies drastically and a single algorithm which is fully automatic and unsupervised which can be applied for any type of roof top structures with any complexity levels is difficult. A solution for this problem is proposed in this paper by eliminating inhomogeneities due to varying hierarchy. The algorithm, instead of focussing the region of interest, considers regions other than the areas having building features. The method is evaluated with various qualitative and quantitative measures which validates the superior performance of the proposed method.

The paper is organized as follows: The methodology and the algorithm are discussed in Section 2. Result analysis and performance evaluation are described in Section 3. Finally the paper is concluded in Section 4.

2. THE METHODOLOGY

In this paper, Quickbird satellite imagery is used to test the building extraction strategy. The input Pan sharpened multi-spectral satellite (PSMS) imagery is derived by pan sharpening from a low resolution(2.4m) multi-spectral (R,G,B,NIR) bands and high resolution (0.6m) panchromatic band Quick bird image. Pan sharpening is a pixel level fusion technique used to increase the spatial resolution of the multi spectral image using spatial information from the high resolution panchromatic image while preserving the spectral information in the multispectral image. Thus the input PSMS image obtained will have a resolution of 0.6m with multispectral (R, G, B, NIR) bands. This is done using Multispec32 which is a freeware multispectral image data analysis system. Multispec32 generates the .lan file of the PSMS image using high resolution panchromatic and low resolution multispectral satellite imagery. The resultant PSMS image has the same resolution as that of PAN image (Figure 1).

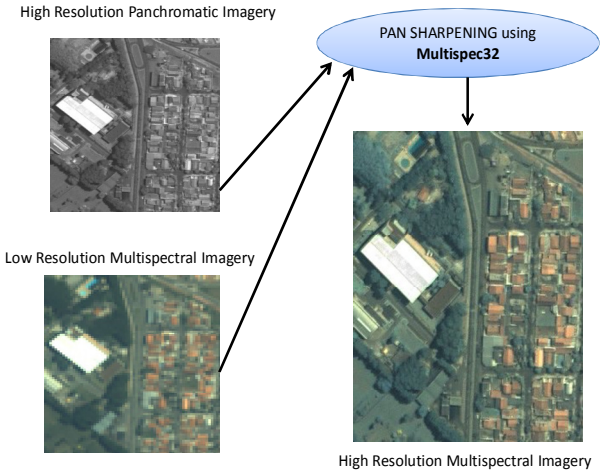


FIGURE 1: Pan Sharpening of Satellite Image

The method first calculates NDVI (Normalized Difference Vegetation Index) and chromaticity to intensity ratio for the initial level of segmentation. Next, rooftops and roads are detected and eliminated. Then principal component analysis and area analysis is done to get accurate results. The block diagram of the proposed system is shown in figure 2.

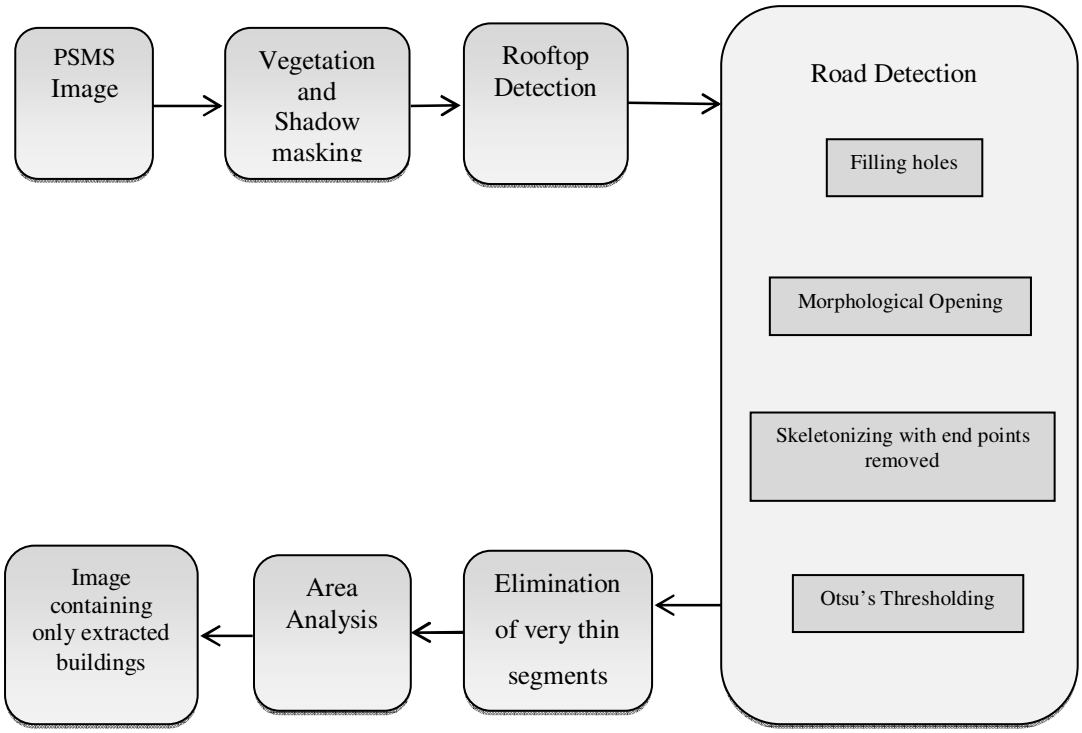


FIGURE 2: Overall Flow of the System

2.1. Calculation of Normalized Difference Vegetation Index

The NDVI is a simple graphical indicator that can be used to analyze remote sensing measurements, typically but not necessarily from a space platform, and assess whether the target being observed contains live green vegetation or not. Healthy vegetation absorbs most of the visible light that hits it, and reflects a large portion of the near-infrared light. Unhealthy or

sparse vegetation reflects more visible light and less near-infrared light. It is possible to measure the intensity of light coming off the Earth in visible and near-infrared wavelengths and quantify the photosynthetic capacity of the vegetation in a given pixel (one pixel is 0.6 square km in our concern) of land surface. Nearly all satellite Vegetation Indices employ this difference formula to quantify the density of plant growth on the Earth — near-infrared radiation minus visible radiation divided by near-infrared radiation plus visible radiation. The result of this formula is called the Normalized Difference Vegetation Index (NDVI). Written mathematically, the formula is:

$$NDVI = \frac{NIR - R}{NIR + R} \tag{1}$$

where NIR is the reflectance value in the near-infrared channel, and R is the reflectance value in the red channel. Calculations of NDVI for a given pixel always result in a number that ranges from minus one (-1) to plus one (+1); however, no green leaves gives a value close to zero. A zero means no vegetation and close to +1 (0.8 to 0.9) indicates the highest possible density of green leaves.

IF (NDVI≥0.06) THEN ASSIGN Vegetation region;
ELSE ASSIGN Building region; (2)

The NDVI image can be obtained in MATLAB by reading the NIR data of the image from its .lan file generated using the software Multispec32. Thus vegetation can be eliminated by the comparison of obtained NDVI image and the original image. Regions are considered to be vegetation and the building hypotheses are rejected when the NDVI value is higher than 0.06. Figure 3 shows PSMS Quickbird image and corresponding vegetation masked image.

FIGURE 3 (a): Quickbird PSMS Image **(b)** Vegetation masked Image

2.1. Calculation of Chromaticity to Intensity Ratio

To detect shadow in the PAN sharpened, vegetation masked image, the ratio chromaticity to luminescence has to be found [15]. For this RGB space is converted to YIQ space. As the shadow regions are comparatively darker compared to other regions they will have higher ratio of Q to I. A suitable threshold is determined by Otsu's method [16] as the amount of shadows present in the high-resolution imagery will vary depending on the sun azimuth and elevation angles and the sensor azimuth angle.

Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold, i.e. the pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of

foreground and background spreads is at its minimum. This can be achieved by finding a threshold with the maximum *between class* variance and minimum *within class* variance. This can be calculated by the following equations (3 and 4).

$$\textit{Within}\ \text{Class Variance}\ \sigma_w^2 = \mathrm{W}_b\sigma_b^2 + \mathrm{W}_f\sigma_f^2 \tag{3}$$

$$\begin{aligned}\textit{Between}\ \text{Class Variance}\ \sigma_B^2 &= \sigma^2 - \sigma_w^2 \\ &= \mathrm{W}_b(\mu_b\text{-}\mu)^2 + \mathrm{W}_f(\mu_f\text{-}\mu)^2 \ (\text{where}\ \mu = \mathrm{W}_b\mu_b + \mathrm{W}_f\mu_f) \\ &= \mathrm{W}_b\mathrm{W}_f(\mu_b\text{-}\mu_f)^2 \end{aligned} \tag{4}$$

where weight, mean and variance are represented by *w,* μ and σ. The foreground and background regions are distinguished as *f* and *b*. Figure below shows the result of shadow masking.

FIGURE 4: Shadow masked Image

2.2. Rooftop Detection

For the detection of rooftops a segmentation algorithm is required. Most of the segmentation algorithms work only on gray scale images but our image is a 4-band multispectral image. In order to retain the spectral features even after segmentation a mean shift algorithm is used [17]. As for image segmentation, the aim is to cluster pixels sharing a similarity in pixel values. For this purpose, the filtering procedure is run and all convergence points are stored. The mean shift vector always points toward the direction of the maximum increase in the density. The mean shift procedure, obtained by successive computation of the mean shift vector $m_h(x_t)$ and translation of the window $x_{t+1} = x_t + m_h(x_t)$ is guaranteed to converge to a point where the gradient of density function is zero. Coming to our concern, iterative calculation of mean shift vectors converges to a stationary point of the density, which corresponds to the modes of the image, i.e. homogenous structures in general. The pixel points converging to the same mode, are closer to each other in terms of spatial extend and color bandwidth. These pixels are segmented as the same cluster. In fact, mean shift vectors are aligned towards the similarity of colors incorporating spatial information. The algorithm for mean shift segmentation is as follows (Table.1):

1. Convert the image from RGB space to LUV space.
2. Select a data point (center pixel) from the image randomly (not labeled).
3. Set the value of that point as the mean.
4. Find all the data points having the same value (a value within a specified range).
5. Cluster those points.
6. Select a point from the cluster.
7. If it is within a window or specified bandwidth(radius) from the center pixel,
   Yes-label it
   No-no change
8. Go to step 5 until all the points in the cluster are evaluated.
9. Calculate the peaks of points having the same label.
10. If distance between the peaks of two labeled segments is less than or equal to bandwidth/2,
    Yes-merge them
    No-no change
11. Compute the mean shift vector of the clustered points $m_h(x)$.

$$\mathbf{m}_h(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)} - \mathbf{x}$$

where h is the bandwidth parameter and g is the gradient density function.

12. Translate the window to the next point $\mathbf{x}^{t+1}$.

$$\mathbf{x}^{t+1} = \mathbf{x}^t + \mathbf{m}_h(\mathbf{x}^t)$$

13. If $\mathbf{x}^{t+1} = \mathbf{x}^t$
    Yes- go to step 14
    No- go to step 2
14. If all the points are labeled
    Yes-end
    No-go to step 1.

TABLE 1: Algorithm for mean shift segmentation

FIGURE 5: Mean shift segmented Image

2.3. Road Detection

After segmentation, some noises seen with the buildings in segmented regions. The edges of these regions are irregular. Hole filling followed by morphological opening operation is used to remove pseudo pixels and smooth building's edge. A hole may be defined as a background

region surrounded by a connected border of foreground pixels. Let *A* denote a set of points whose elements are 8-connected boundaries, each boundary enclosing a background region (a hole). Given a point in each X_k defined by (5) as a hole, the objective is to fill all the holes with ones.

$$X_k = (X_{k-1} \oplus B) \cap A^{C} \qquad k=1,2,3,..... \tag{5}$$

Opening operation is of the form (6):

$$X \circ B = (X \Theta B) \oplus B \tag{6}$$

That is, image *X* is eroded by structure element *B* (equation 7), then it is dilated by *B*. We define,

$$B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{7}$$

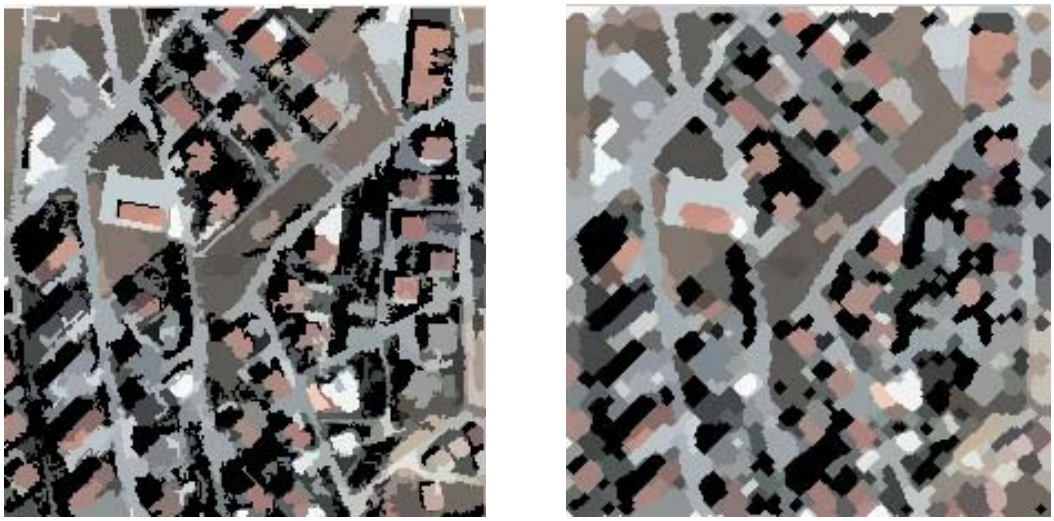


FIGURE 6 (a): Image after hole filling **(b)** Image after morphological opening

Figure 6(a) and 6(b) show the results of hole filling and morphological opening. After morphological treatment, edges have become smooth, but there are still some discrete noise. Pixels belonging to rooftop of buildings may wrongly identify as road pixels because both have linear features. To eliminate this, skeletonise the image (figure 7(a)), as now our region of interest is main roads alone.

For skeletonization, label pixel p if and only if the rules 1, 2, 3, 4 are all satisfied.

Rule 1: The pixel under consideration must presently be black. If the pixel is already white, no action needs to be taken.
Rule 2: At least one of the pixel's close neighbours must be white.
Rule 3: The pixel must have more than one black neighbour.
Rule 4: A pixel cannot be removed if it results in its neighbours being disconnected.

After each iteration the labeled pixels are deleted. The algorithm is performed until one pixel wide skeleton is obtained.

For identifying whether a segment is road or not, find out the length of the segments which is equal to the number of pixels in skeleton. From the distribution of length of segments threshold is automatically estimated using Otsu's method. If length is greater than this threshold then classify it as road. In the skeletonised image, there may be unwanted branches which are obviously non-road segments. So end points of the skeletons, which have only one neighbor, are removed (figure 7(b)) for better result. Figure 8 is the road masked image after these operations.

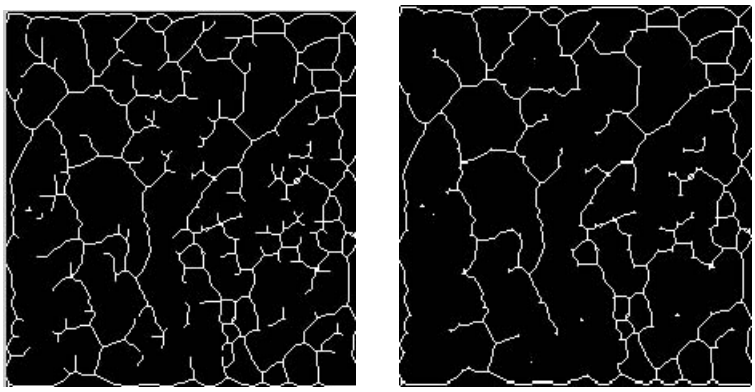


FIGURE 7 (a): Skeletonized Image **(b)** Skeletonized Image with end points removed

FIGURE 8: Road masked image

2.3. Elimination of Very Thin Segments

Some small objects such as cars, trees on the side of roads can also be viewed as thin stripes after skeletonization. This can be eliminated by applying, principal component analysis (PCA) to each segment. Very thin segments show large variances along the first principal component whereas small variance along the second principle component. Therefore, the ratio of the corresponding eigenvalues, provides the variances along the corresponding eigenvectors, gives an idea of how thin the segment is. Higher ratios represent unreasonably thin segments. Then, this ratio is thresholded in order to detect whether the segment is road or not. The threshold is automatically determined by Otsu's method.

Principal component analysis (PCA) is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values

of uncorrelated variables, called principal components. The number of principal components is less than or equal to the number of original variables. This transformation is defined in such a way that the first principal component has as high a variance as possible, and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to (uncorrelated with) the preceding components. PCA is the simplest of the true eigenvector-based multivariate analysis.

PCA is done by calculating the mean of each co-ordinate X and Y of the segment and then finding the covariance matrix. Covariance matrix for a set of data with n dimensions is given by (8) and (9).

$$C^{m\times n} = \operatorname{cov}(Dim_i, Dim_j) \tag{8}$$

$$\operatorname{cov}(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})\,(Y_i - \overline{Y})}{(n-1)} \tag{9}$$

Then find the eigen values λ_1,λ_2 of C_x using (10).

$$|C_x - \lambda I| = 0 \tag{10}$$

The eigen vector with the highest eigen value is the principal component. The eigen vectors are ordered by eigen value from highest to lowest. This gives the components in the order of significance. Form the feature vector as (11).

$$\textit{Feature vector} = (\textit{eig1 eig2 eig3 } \ldots. \textit{ eign }) \tag{11}$$

The final data is derived by the following equation (12).

$$\textit{Final data} = \textit{Row feature vector x Row Data Adjust} \tag{12}$$

Row Feature Vector is the matrix with the eigenvectors in the columns transposed so that eigen vectors are now in the rows, with the most significant eigenvector at the top. Row Adjust Data is the mean adjusted data transposed. The data items are in each column, with each row holding a separate dimension.

2.4. Area Analysis

After these operations, there are still a lot of regions which are similar to building areas. The size of these areas is very small compared to buildings and can be eliminated by connected component labelling. To verify the hypothesized connected components as building regions, area analysis is performed. For that, the minimum enclosing rectangle (MER) [18] of each and every connected component is found. Then rectangular fit is calculated as the area of the component divided by the area of its MER. If the rectangular fit is lower than a threshold, the connected component is rejected which reduces the percentage of over detection. The results of PCA and area analysis for the PSMS image are shown in figure 9.

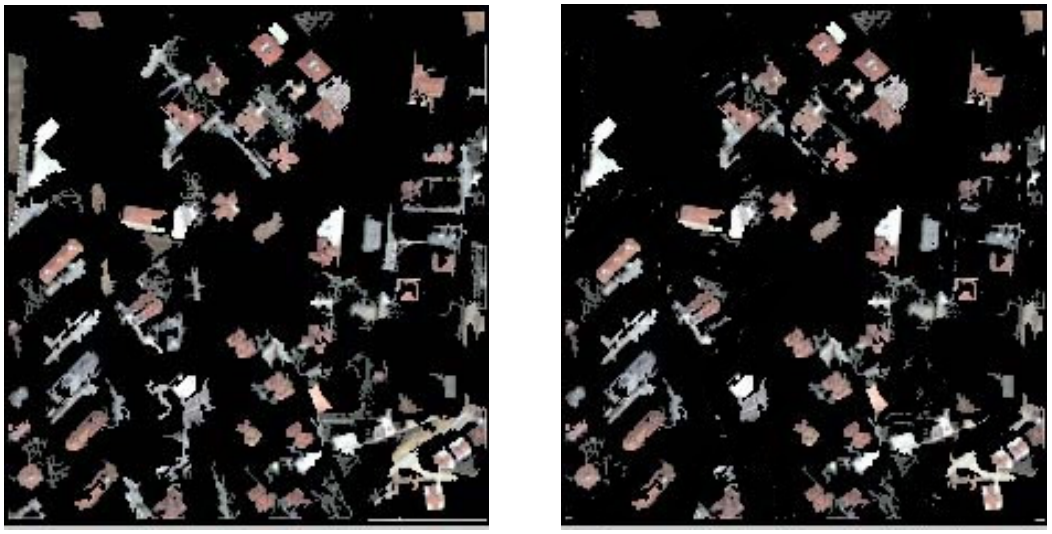


FIGURE 9 (a) : Image after PCA **(b)** Image after Area Analysis

3. RESULTS & DISCUSSION

An algorithm for fully automatic and unsupervised building extraction is presented. Figure 10 shows the extracted results compared with the manually labelled buildings used as reference. It can be seen that irrespective of shape, most of the buildings are detected without fail.

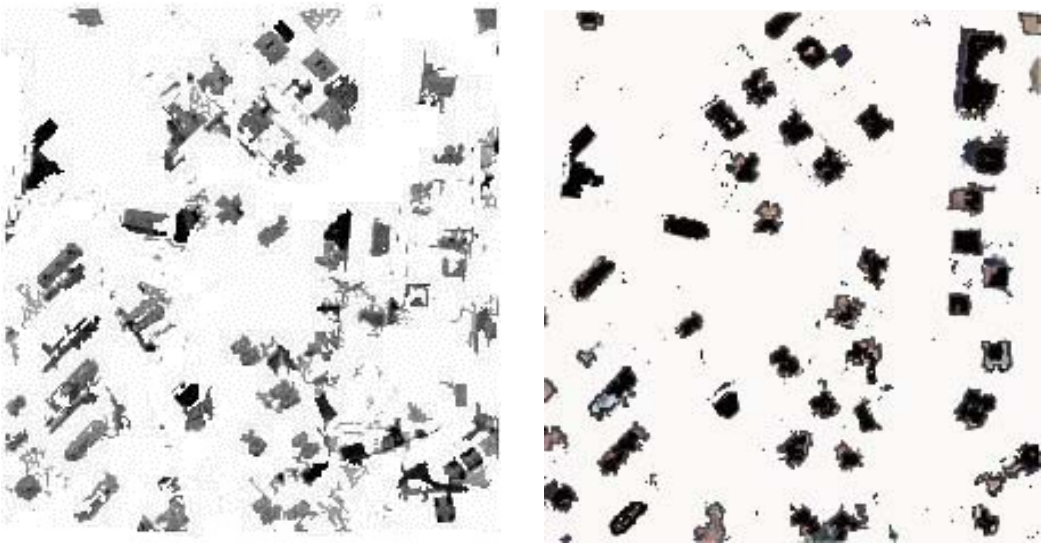


FIGURE 10 (a): Extracted buildings **(b)** Manually labelled buildings

Within the test area, 49 buildings were manually delineated. As seen from the output image the same number of buildings is detected but many of the scattered points can also be considered as buildings. The manually delineated buildings were used as a reference building set to assess the unsupervised building extraction accuracy. For performance evaluation, we use the evaluation measures widely accepted for building extraction [19, 20]. The extracted buildings and the manually detected buildings are compared pixel-by-pixel. The pixels in the image are categorized into four types:

(1) True positive (TP): Both manual and unsupervised methods label the pixel belonging to the buildings.

(2) True negative (TN): Both manual and unsupervised methods label the pixel belonging to the background.
(3) False positive (FP): The unsupervised method incorrectly labels the pixel as belonging to a building.
(4) False negative (FN): The unsupervised method does not correctly label a pixel belonging to a building.
Based on these categories the system performance is evaluated using the following measures:

$$\text{Branching Factor} = \frac{FP}{TP} \tag{13}$$

$$\text{Miss Factor} = \frac{FN}{TP} \tag{14}$$

$$\text{Detection Percentage} = 100 \times \frac{TP}{TP + FN} \tag{15}$$

$$\text{Quality Percentage} = 100 \times \frac{TP}{TP + FP + FN} \tag{16}$$

The *detection percentage* is the percentage of building pixels correctly labelled by the proposed method. The *quality percentage* measures the quality of the extraction process. Performance evaluations for three different images of various complexity levels are shown in Table 2. fig. 12 and fig. 13 show the building extraction results compared with the manually detected reference for image-2 and image-3 which are shown in fig. 11.

Reference Images	Branching factor	Miss factor	Detection percentage	Quality percentage
Image 1	0.12	0.09	95.3	81.5
Image 2	0.24	0.31	83.7	70.6
Image 3	0.38	0.47	74.6	60.3

TABLE 2: Performance evaluation of extracted buildings

FIGURE 11 (a) : Image-2 **(b)** Image-3

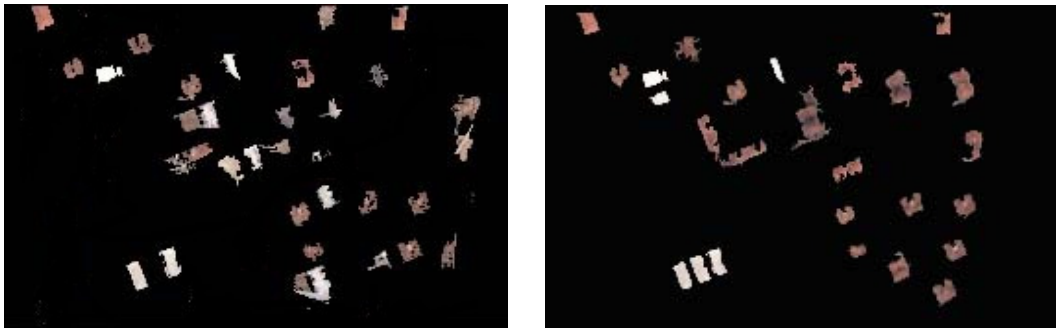


FIGURE 12: Extracted results compared with manually labelled reference for Image-2

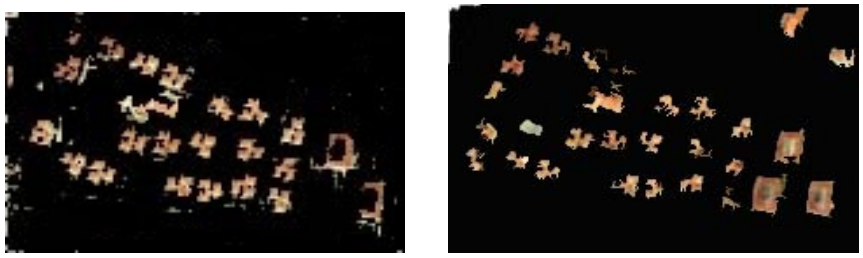


FIGURE 13: Extracted results compared with manually labelled reference for Image-3

The results of the performance evaluation show that the algorithm provides 81.5 % quality percentage for image 1 which is considerably good. The detection percentage for the same case is 95.3% which is very high compared to previous methods.

In our work, masking is performed on areas other than the region of interest which helped to achieve improved performance of the algorithm. It became possible to incorporate both spatial and spectral properties of the image through the use of mean shift segmentation. Principal component analysis using statistical properties helped to achieve better adaptability for different complexity levels of urban areas. Finally extensive analysis with respect to area is done, which removes most of the unwanted pixels classified as buildings. In this approach, building extraction is completely automatic and unsupervised. Here we only provide an input which is the pan-sharpened multispectral image and we get an output which is the candidate buildings.

5. CONCLUSION & FUTURE WORK

In this paper, a fully automatic and unsupervised system is designed for extracting high density urban area buildings irrespective of rooftop structures from satellite images. Though the variety of the type of buildings and shapes of rooftops makes the building extraction complicated, acceptable accuracy for the extracted buildings is obtained. Quantitative analysis of the work indicates that our method is better compared to existing methods. The output image includes scattered black points which may be detected as buildings. Also some roads and pavements are also detected as roads. Future work will include post processing algorithms to reduce these errors.

6. REFERENCES

[1] Henricsson O., “The Role of Color Attributes and Similarity Grouping in 3-D Building Reconstruction”, *Computer Vision and Image Understanding, 1998*.

[2] A. Fischer, T.H.Kolbe, F. Lang, A. B. Cremers, W. F¨orstner, L. Pl¨umer, and V. Steinhage, “Extracting buildings from aerial images using hierarchical aggregation in 2D and 3D”, *Computer Vision and Image Understanding*, pages 185–203, 1998.

[3] T. Moons, D. Fr`ere, J. Vandekerckhove, and L. Van Gool, “Automatic modeling and 3D reconstruction of urban house roofs from high resolution aerial imagery”, In Proceedings, *Fifth European Conference on Computer Vision*, *Vol. I*, pages. 410–425, 1998.

[4] D. Koc San and M. Turker, “Automatic Building Extraction From High Resolution Stereo Satellite Images”, *ISPRS commission VII,* 2007

[5] G. Sohn and J. Dowman, "Building Extraction Using LIDAR DEMs and IKONOS Images", *ISPRS, Volume XXXIV, PART 3/W13*, 2003.

[6] F. Rottensteiner and C. Briese., “A New Method for Building Extraction in Urban Areas from High-Resolution LIDAR Data”, *Symposium der ISPRS-Comm. III. International Archives of Photogrammetry and Remote Sensing, Volume XXXIV / 3A*, pp. 295 – 301, 2002.

[7] Yanfeng Wei, Zhongming Zhao, and Jianghong Song. “Urban building extraction from high-resolution satellite panchromatic image using clustering and edge detection”, *Geoscience and Remote Sensing Symposium*, 2005.

[8] S.D Mayungaa , Dr. Y. Zhanga, and Dr. D.J. Colemana, “Semi-Automatic Building Extraction Utilizing Quickbird Imagery”, *IAPRS Vol. XXXVI, Part 3/W24*, 2005.

[9] Jin, X. and Davis, C. H., “Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information,” EURASIP Journal on Applied Signal Processing, 2196–2206 (2005).

[10] H. Akçay and S. Aksoy, “Automatic detection of geospatial objects using multiple hierarchical segmentations,” *IEEE Trans. Geosci. Remote Sens.*,vol. 46, no. 7, pp. 2097–2111, Jul. 2008.

[11] Ö. Aytekin, A. Erener, I. Ulusoy, H. S. Düzgün, “Automatic and Unsupervised Building Extraction in Complex Urban Environments from Multi-Spectral Satellite Imagery”, 4th International Conference on Recent Advances in Space Technologies, Space for the Developing World, RAST 2009, Istanbul, Turkey, June 2009.

[12] Katartzis, A. and Sahli, H., “A stochastic framework for the identification of building rooftops using a single remote sensing image,” IEEE Transactions on Geoscience and Remote Sensing 46, 259 – 271 (2007).

[13] B. Sirmacek and C. Unsalan, "A probabilistic framework to detect buildings in aerial and satellite images", IEEE Transactions on Geoscience and Remote Sensing, Vol. 49 (1), pp. 211-221, January 2011.

[14] Hazelhoff, L., De With, P.: “Localizations of buildings with a gable roof in very-highresolution aerial images” In: Proceedings of IS&T SIE Electronic Imaging, Visual Information Processing and Communication II (2011).

[15] Tsai, 2006, “A comparative study on shadow compensation of color aerial images in invariant color models”, IEEE Transactions On Geoscience And Remote Sensing, vol. 44, no. 6, June 2006.

[16] N. Otsu, "A threshold selection method from graylevel histograms," *IEEE Trans. Sys. Man Cyber.*, vol. 9, no. 1, pp. 62-66, 1979.

[17] D. Comaniciu and P. Meer. Mean shift: "A robust approach toward feature space analysis”, IEEE Trans. Pattern Anal. Machine Intell., 24:603–619, 2002.

[18] L. da F. Costa and R. M. Cesar Jr., "*Shape Analysis and Classification: Theory and Practice"*, CRC Press, Boca Raton, Fla, USA, 2001.

[19] J. A. Shufelt, "Performance evaluation and analysis of monocular building extraction from aerial imagery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 4, pp. 311–326, 1999.

[20] D. S. Lee, J. Shan, and J. S. Bethel, "Class-guided building extraction from Ikonos imagery," *Photogrammetric Engineering and Remote Sensing*, vol. 69, no. 2, pp. 143–150, 2003.

Change Detection of Water-Body in Synthetic Aperture Radar Images

Sree Sharmila T *sreesharmilat@ssn.edu.in*
Assistant Professor
SSN College of Engineering
Chennai, India

Ramar K *kramar_nec@rediffmail.com*
Einstein College of Engineering
Tirunelveli, India

Vidhusha S *vidhushas@ssn.edu.in*
Assistant Professor
SSN College of Engineering
Chennai, India

Abstract

Change detection is the art of quantifying the changes in Synthetic Aperture Radar (SAR) images occurring over a period of time. Remote sensing has been instrumental in performing change detection analysis. The impact of applying the combination of texture features for classification techniques to separate water bodies from land masses is empirically investigated in this paper. First, the images are classified using unsupervised Principle Component Analysis (PCA) based K-means clustering for dimension reduction. Then, texture features like Energy, Entropy, Contrast, Inverse Difference Moment, Directional Moment and Median are extracted using Gray Level Co-occurrence Matrix (GLCM) and these features are utilized in Learning Vector Quantization (LVQ) and Support Vector Machine (SVM) classifiers. This paper aims to apply a combination of the texture features in order to significantly improve the accuracy of detection. The utility of detection analysis influences management and policy decision making for long-term construction projects by predicting preventable losses.

Keywords: Change Detection, Classification, Learning Vector Quantization, Support Vector Machine, Texture Features

1. INTRODUCTION

Remote sensing and related techniques such as geographic information systems have a persistent impact on the conduct of realistic work. With satellite instruments, it is possible to observe a target repeatedly, thereby contributing effectively to perform change detection in areas of interest.

This work mainly focuses on change detection, which happens because of many possible environmental and human actions. SAR images enable direct observation of the land surface at repeated intervals, allowing mapping, monitoring and assessment. Change detection analysis is effective in long-term planning. PCA based K-means clustering is a statistical technique widely used for dimensional reduction. Usually the clustering methods are developed for different purposes. Clustering algorithms used for unsupervised classification of remote sensing data vary according to the efficiency with which clustering takes place. The unsupervised clustering provides the cluster information about the water body in a relatively quick manner [5]. This lacks complete information about the region of interest and particularly subtle variations therein. To avoid unexpected groupings, supervised classification is recommended.

The mapping of classes is much more accurate in supervised classification but is heavily dependent on the input given. The researchers compared and analyzed SVD (Singular Value Decomposition) and Gray Level Co-occurrence Matrix (GLCM) as two methods for feature extraction [4, 2]. The comparison result shows that GLCM gives a better result than SVD. Hence, the texture features from the training data sets were extracted by using GLCM and then subjected to the LVQ.

The classes of interest that is the coastal and non-coastal areas and the learning rate are determined to optimize the classification accuracy of the SAR images [1]. This paper aims to further improve the accuracy of classification by combining the six texture features instead of using a single or different combination of texture features. Like PCA, LVQ is also used in feature vector dimension reduction. Here is the chance to optimize training data through reducing the number of samples for this analysis. After achieving the desired accuracy by LVQ, the features were subjected onto SVM. In SVM, the two classes were identified and differentiated which helped in finding out the classes in target image. Once the images were classified using supervised classification the change map is constructed to depict the changes [8,13,14].

2. METHODOLOGIES USED FOR CHANGE DETECTION

Figure 1 and Figure 2 depict the major steps in the nomenclature of water bodies and land masses region using unsupervised and supervised classification.

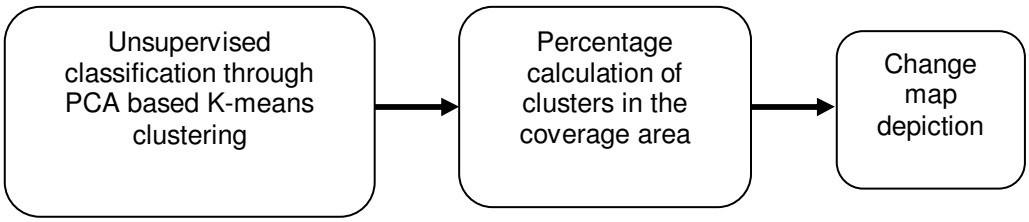


FIGURE 1: Modules of Clustering

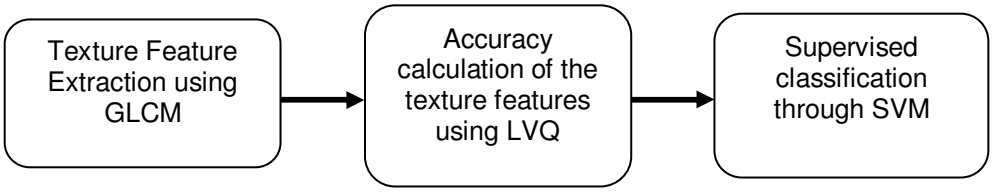


FIGURE 2: Feature Extraction and Supervised Classification

2.1 PCA based K-Means Clustering

The most important aspect of image classification is finding groups in data [4, 5]. Pixels with similar intensities forms clusters in images. Gray values are specifically used for this purpose. The K-means algorithm has some further refinements for its applications on change detection, by splitting and merging of clusters. This is illustrated in Figure 1. First, the given image values are preprocessed by using PCA. Using the most important components of PCA, the image information is mapped into the new feature space. Then, the K-means algorithm is applied to the data in the feature space. The final objective is to distinguish the different clusters using eigen values. Clusters are grouped if the cluster standard deviation exceeds a threshold and the number of pixels is twice for the minimum number of pixels. The main intention of K-means algorithm is to reduce variability within clusters [5]. The objective function is the sum of square distances between cluster centre and its assigned pixel value.

$$F = \sum_{i=0}^{n} [x_i - C(x_i)]^2 \quad (1)$$

where, x_i is the pixel value assigned to mean value of the cluster $C(x_i)$. Next the Mean Squared Error (MSE) is determined; minimizing the error is equivalent to minimizing the sum of squared distances.

$$\text{Error} = \frac{\sum_{i=0}^{n} [x_i - C(x_i)]^2}{(N - C)} \quad (2)$$

where, N indicates how many number of pixels, C specifies the expected number of clusters. K-means is very responsive to initial values. It is often not obvious that the clustering with the lesser MSE is truthfully the better classification. Thus, the results obtained have aided in attaining this objective and thereby PCA based K-means classification has been efficiently applied for the change detection analysis.

The unsupervised classification provides the cluster information about the water body in a relatively quicker manner [12]. This clustering lacks complete information about the region of interest and particularly subtle variations therein. Supervised classification gives the better results compared to the unsupervised classification.

2.2 Texture Feature Extraction

Texture portrays a rich source of data about the natural landscape. The ways of extracting the texture features have been performed through GLCM. It is a perfect and an efficient tool to perform the extraction of texture features [4, 6]. The basis of GLCM is assigning the relationship between two neighbouring pixels in one offset as the second order texture. The gray value relations in a target image are transformed into the co-occurrence matrix by a given kernel mask 3×3. In the transformation from the image onto the co-occurrence matrix, the neighbouring pixels in 0° direction can be used. It contains information of the position of pixels having similar gray level values.

The texture features extracted for classification are Energy (E), Entropy (Ent), Contrast (Con), Inverse Difference Moment (IDM), Directional Moment (DM) and the Median (M). Energy can be defined as the measure of the extent of pixel pair repetitions. Entropy is the measure of randomness that is used to characterize the texture of the input image. Its value will be maximum when all the elements of the co-occurrence matrix are the same. The contrast is a measure of intensity of a pixel and its neighbour over the image. Contrast is 0 for a constant image. Inverse Difference Moment is a measure of image texture. IDM has a range of values so as to determine whether the image is textured or non-textured. Directional moment is a textural property of the image computed by considering the alignment of the image as a measure in terms of the angle [15-17]. There are four orientations namely 0°, 45°, 90° and 180°. The following defines the texture features subjected for classification.

$$E = \sqrt{\sum_{i=0}^{M-1} \sum_{j=o}^{M-1} M^2(i, j)} \quad (3)$$

$$Ent = \sum_{i=0}^{M-1} \sum_{j=o}^{M-1} M(i,j)(-\ln(M(i,j))) \quad (4)$$

$$Con = \sum_{i=0}^{M-1} \sum_{j=o}^{M-1} (i-j)^2 M(i,j) \quad (5)$$

$$IDM = \sum_{i=0}^{M-1} \sum_{j=o}^{M-1} \frac{1}{1+(i-j)^2} M(i,j) \quad (6)$$

$$DM = \sum_{i=0}^{M-1} \sum_{j=o}^{M-1} M(i,j) \left| i-j \right| \quad (7)$$

$$M = \sum_{i=0}^{M-1} \sum_{j=o}^{M-1} \frac{M(i,j)+1}{2} \quad (8)$$

where i and j are the coefficients of co-occurrence matrix, $M(i,j)$ is the element in the co-occurrence matrix at the coordinates i and j, M is dimension of the co-occurrence matrix.

2.3 Learning Vector Quantization

LVQ is a variation of the well-known Self Organizing Map (SOM) architecture. This has a feed-forward network structure with a single layer of neurons arranged in rows and columns. Each neuron is fully connected to input layer source units. The different stages of LVQ are:

Step 1: Initialization – Assign the initial weight vector w_j by selecting the random values.
Step 2: Sampling – Describe a sample input vector x from the input space. Here the network chooses five texture features as input to classify the two classes.
Step 3: Matching – Find weight vector closest to the input vector named as winning neuron I(x), which is defined as,

$$\sum_{i=1}^{n} (x_i - w_{ji})^2 \quad (9)$$

Step 4: Updating – Apply the following weight update function to update the weights

$$\Delta w_{ji} = \eta\, T_{j,I(x)}\left(x_i - w_{ji}\right) \tag{10}$$

where $T_j, I_{(x)}$ is a Gaussian neighbourhood and η is the learning rate.

Step 5: Continuation – Repeat the steps 2 to 5 to reach the best accuracy of input space.

2.4 Support Vector Machine

A support vector machine is a set of related supervised learning algorithms that analyze the inputs and recognize the classes, used for classification and regression. SVM takes a set of input data; for each given input, it determines whether the input is a member of which of two classes. This makes SVM as a linear classifier [7]. The training algorithm of SVM constructs a model that assigns new data into one category or the other of two classes. Figure 3 depicts basic building blocks of SVM as follows,

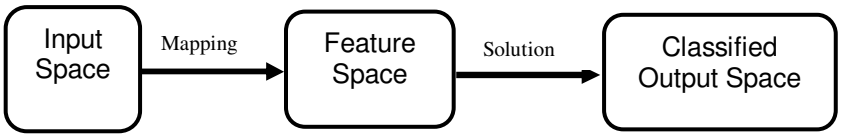


FIGURE 3: The working of SVM algorithm

The goal is to separate the two classes without loss of generality by a function which is induced from available training data sets. The task is to produce a classifier that will work in a generalized manner [3, 9]. The experiment for handling the data set is tested on a change detection problem with a large set of data points [11]. The application of SVM for the desired problem is minimizing the error through maximizing the margin which means that it maximizes the distance between it and the nearest data point of each class [10]. Since SVM are known to generalize well even in high dimensional spaces under small training sample. This linear classifier is termed as the optimal separating hyper plane. Cover's theorem states that if the transformation is non-linear and the dimensionality of the feature space may be transformed into a new feature space is high enough, then the input space may be transformed into a new feature space where the patterns are linearly separable with high probability. This non-linear transformation is performed in an implicit way is called kernel function [9,10].

SVM is known to generalize well even in high dimensional spaces under small training sample conditions and have shown to be superior to traditional neural networks. The experiment for handling the data set is tested on a change detection problem with a large set of data points. The support vector machine classifier uses large set of training and testing data for classification of coastal areas.

3. RESULTS AND DISCUSSIONS

3.1 Data Sets

This paper deals with the Land Remote-Sensing Satellite (LANDSAT) images taken from different time frames of the coastal regions of the world. Some of the sample input study area imageries are shown in Figure 4 and 5.

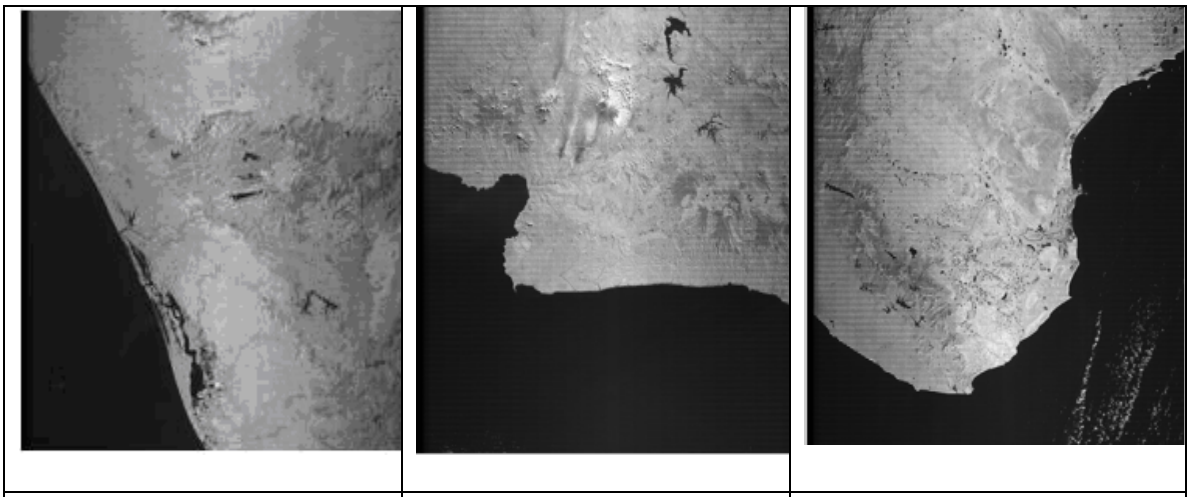


FIGURE 4: Input Landsat images of coastal landscape taken during 2005

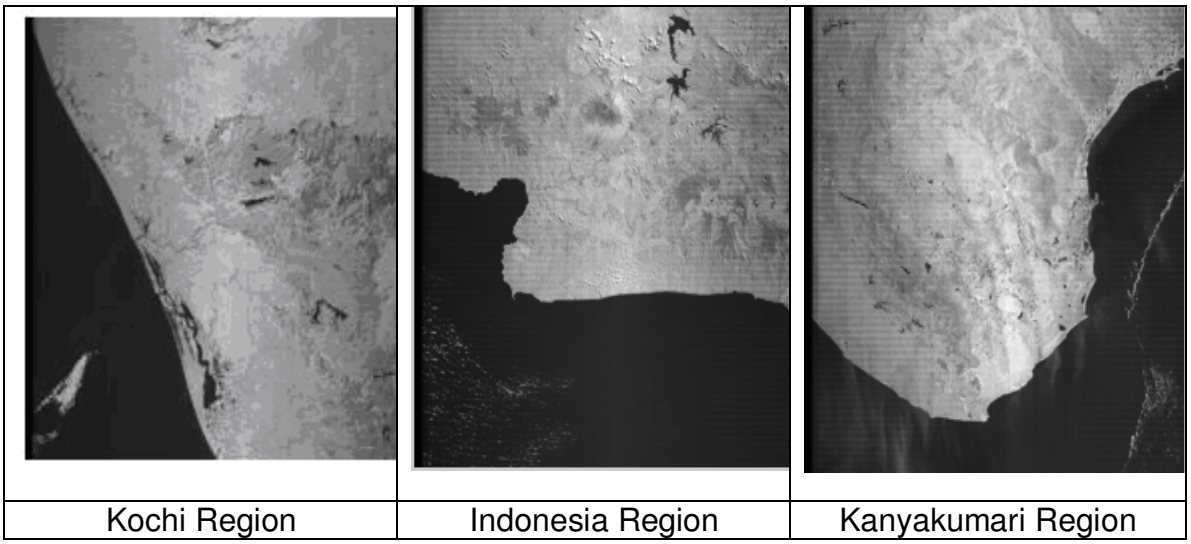


FIGURE 5: Input Landsat images of coastal landscape taken during 2010

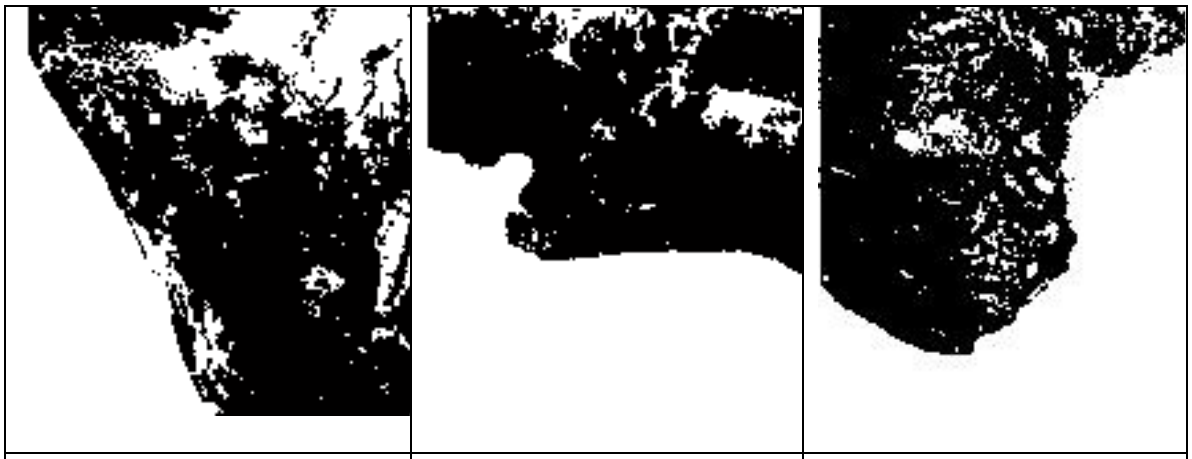
3.2 Results of Classification Through PCA Based K-means Clustering

The input SAR images have been subjected to unsupervised classification performed using a PCA based K-means classifier. Figure 6 and Figure 7 depict the result of classification which facilitates differentiation of water bodies from land masses. These results have been further utilized in calculating the percentage of coverage area to detect changes over a period of time.

3.3 Results of Supervised Classification

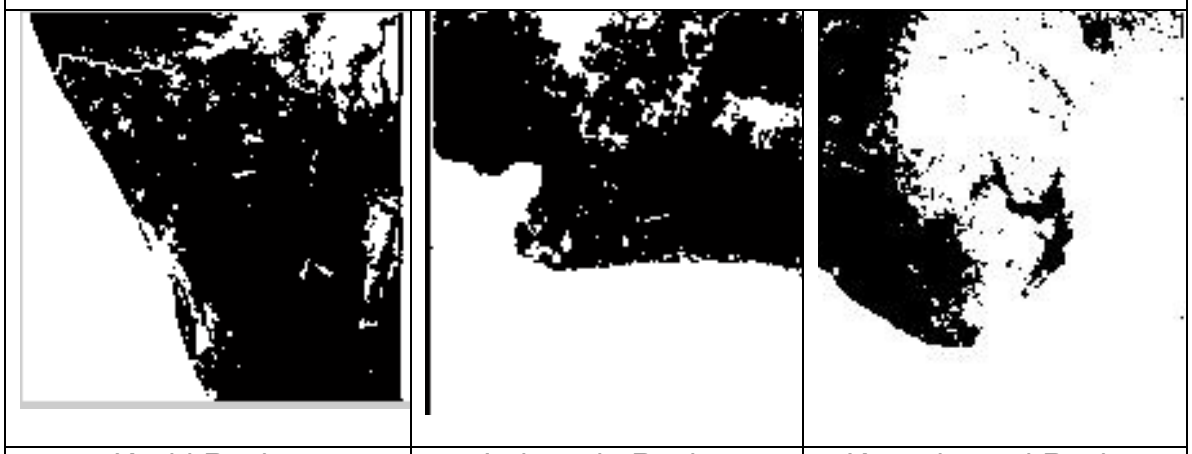
The six texture features are extracted using GLCM and subjected to LVQ to satisfy the necessary and sufficient conditions to achieve maximum accuracy. The training data has led to results with 96% accuracy.

Subjecting the combination of extracted texture features onto the SVM classifier resulted in 98% accuracy for classifying input data. The classified results are depicted in Figure 8.



Kochi Region	Indonesia Region	Kanyakumari Region

FIGURE 6: PCA based K-means clustering performed on 2005 images



Kochi Region	Indonesia Region	Kanyakumari Region

FIGURE 7: PCA based K-means clustering performed on 2010 images

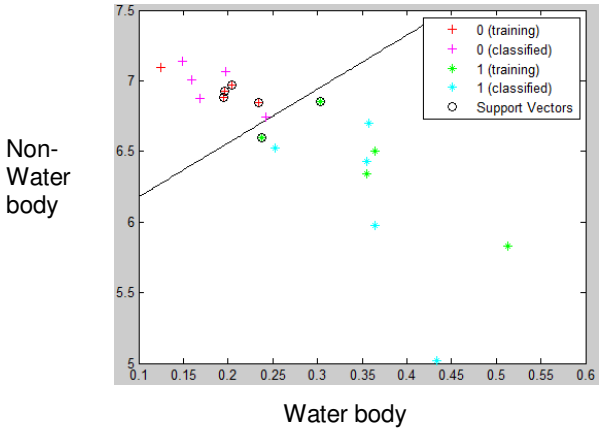


FIGURE 8: The Classification Using SVM

So it is possible to quantifiably depict the amount of increase of water body in 2010 when compared to 2005 as shown in Figure 9.

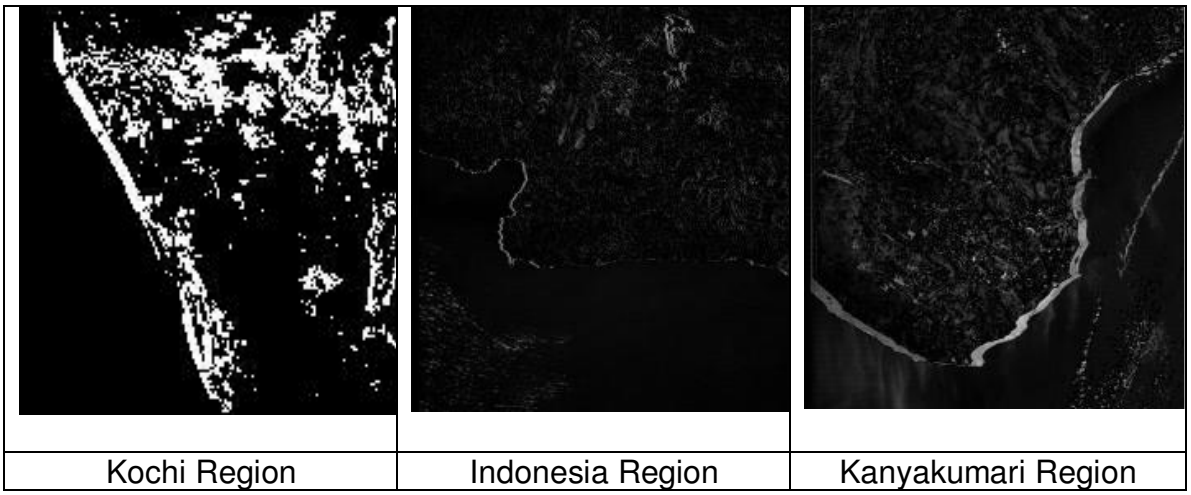


FIGURE 9: Change depicted over the Five Years

Area	2005	2005	2010	2010	Changes detected in the period of 5 years
	Water	Non-water	water	Non-water	
Kochi	26.1200	73.88	29.2	70.8	3.08%
Kanyakumari	46.4966	53.5034	44.2856	55.7144	0.16%
Indonesia	46.6476	53.3524	46.6599	53.3401	0.01%
Kolkata	27.4445	72.555	31.7780	68.2220	4.333%
Vishakhapatnam	62.8494	37.1506	63.5956	36.4044	0.74%
Sydney	56.3477	43.6523	60.0342	39.9658	3.68%
Bhubaneswar	62.8494	37.1506	63.5956	36.4044	0.74%

TABLE 1: Coverage areas of water body and non-water body of different landscapes in percentage

The percentage of changes in water bodies and land masses can be interpreted from the results shown in Figure 9. The count has been materialised by taking pixel values of the classified image into account as the input, and then pixels of the water region are counted first which leads to the count of the land mass. The change in coverage area of water body has recorded a significant increase over five years in the study region, as shown in Table 1. The increase in the coverage areas of water bodies due to global warming has been substantially proven through these results.

4. CONCLUSION

The images are classified using the unsupervised algorithms which do not need pre determined cluster definitions, namely the PCA based K-Means Clustering. In order to support these findings the supervised mechanisms, LVQ and SVM are employed. Instead of applying the texture features separately, the combination of textures from the training data sets are subjected to the LVQ to yield better results. After achieving the required accuracy, the features are subjected to SVM. In SVM, the two classes are identified and differentiated, which helped in finding out the classes in the target image accurately. This work will intend to define a mechanism to aid long-term planning by predicting preventable losses and to generalise this scheme for supporting crisis management due to global warming.

5. REFERENCES

[1] Kaichang Di F., Ruijin Ma, A., Jue Wang, W., Ron Li, U., "Coastal mapping and change detection using high-resolution IKONOS satellite imagery", Digital Government Society of North America, pp.1-4, 2003.

[2] Krishnan, V., Swoboda, J., Yarman, C. E., and Yazici, B., "Multistatic Synthetic Aperture Radar Image Formation", IEEE transactions on image processing, volume: 19, no: 5, pp.1290-1306, 2010.

[3] Kwang In Kim, Keechul Jung Se, Hyun Park and Hang Joon Kim., "Support Vector Machines for Texture Classification", IEEE Transaction on Pattern Analysis and Machine Intelligence, volume: 24, no: 11, pp. 1542-1550, 2002.

[4] Nath, K., and Deb S. K., "Water-Body Area Extraction from High Resolution Satellite Images - An Introduction, Review, and Comparison", International Journal of Image Processing (IJIP), volume: 3, Issue: 6, pp. 353-372, 2010.

[5] Ravichandran, K. S., and Ananthi, B., "Color Skin Segmentation Using K-Means Cluster", International Journal of Computational and Applied Mathematics, volume: 4, no: 2, pp.153-157, 2009.

[6] Rowman, J. F., and George, P. T., "Geological Units Classification of Multispectral Images by Using Support Vector Machines", International Conference on Intelligent Networking and Collaborative Systems, pp.267-272, 2009.

[7] Sun, B. X., and Huang, D.S., "Texture Classification Based on Support Vector Machine and Wavelet Transform", Proceeding of the Fifth World Congress on Intelligent Control and Automation, pp. 15-19, 2004.

[8] Peng Xu, Min Dai, Andrew K. Chan., "Texture Classification Using Optimized Support Vector Machines", IEEE (C), pp.544-547, 2004.

[9] Zhang Zhaohui, T., Veronique Prinet R. and Songde, M.A., "Water body extraction from multisource satellite images", Proc. IEEE Geoscience and remote Sensing Symposium, volume: 6, pp.3970-3972, 2003.

[10] Hyeran Byun and Seong-Whan Lee., "Application of Support Vector machines for Pattern Recognition: A Survey", SVM 2002, LNCS 2388, pp.213-236, 2002.

[11] Guosheng Wang., "A Survey on Training Algorithms for Support Vector Machine Classifiers", Fourth International Conference on Networked Computing and Advanced Information Management, IEEE, pp. 3-128, 2008. DOI 10.1109/NCM.2008.103.

[12] Gonzalez, C. Woods, R.E., "Digital Image Processing" (Publishing House of Electronics industry, 2002, 2nd edn.).

[13] Bagli, S and Soille, P., "Morphological automatic extraction of coastline from landsat images", In Proceedings of the Fifth International Symposium on GIS and Computer Cartography for Coastal Zone Management, volume 3, pp. 58–59, Genova, 2009.

[14] Clausi, D.A, and Yue, B., In Proceedings of the Fifth International Symposium on "GIS and Computer Cartography for Coastal Zone Management", volume 3, pp. 58–59, Genova, 2009.

[15] Christina Corbane, Nicolas Baghdadi, Xavier Descombes, Geraldo Wilson Junior, Nicolas Villeneuve and Michel petit., "Comparative Study on the Performance of Multiparameter SAR Data for Operational Urban Areas Extraction Using Textural Features", IEEE Geoscience and Remote sensing Letters, vol.6, no.4, pp.728-732, 2009.

[16] Chethan, Raghavendra and Hemantha Kumar., "Texture based Approach for Cloud Classification using SVM", International conference on advances in technologies in Communication and Computing, pp.688-690, 2009. DOI: 10.1109/ARTCom.2009.43.

[17] Zhi-Zhong Wang and Jun-Hai Yong., "Texture Analysis and Classification with Linear Regression Model Based on Wavelet Transform", IEEE Transactions on Image Processing, vol.17, no.8, pp. 1421-1430, 2008.

Two-Dimensional Block of Spatial Convolution Algorithm and Simulation

Mussa Mohamed Ahmed *mussa_m7@yahoo.com*
Department of Electronics and Communication
Faculty of Engineering, University of Aden-Yemen.

Abstract

This paper proposes an algorithm based on sub image-partitioning strategy. The proposed scheme divides a grayscale (or color) image into overlapped 6×6 blocks each of which is partitioned into four small 3x3 non-overlapped sub-images. A new spatial approach for efficiently computing 2-dimensional linear convolution or cross-correlation between suitable flipped and fixed filter coefficients (sub image for cross-correlation) and corresponding input sub image is presented. Computation of convolution is iterated vertically and horizontally for each of the four input sub-images. The convolution outputs of these four sub-images are processed to be converted from 6×6 arrays to 4×4 arrays so that the core of the original image is reproduced. The present algorithm proposes a simplified processing technique based on a particular arrangement of the input samples, spatial filtering and small sub-images. This results in reducing the computational complexity as compared with other well-known FFT-based techniques. This algorithm lends itself for partitioned small sub-images, local image spatial filtering and noise reduction. The effectiveness of the algorithm is demonstrated through some simulation examples.

Keywords: Spatial Convolution, Algorithm, Partitioning, Flipping, Simulation.

1. INTRODUCTION

Convolution is a very useful tool to determine the response of a given system. For example, in a wavelet bank filter, bands are computed using convolution [1]. In image processing, convolution operation is useful for detecting the edges [2]. So far, several methods have been developed to compute 2-D system responses such as the circulant or Toeplitz matrices [3], sliding window method as a spatial filtering [4], which requires moving the center of a mask through an image and does not require folding a mask as in case of tabular method. However, the computation of responses using the above methods is a redundant one, so it has a global response computation, which causes beside noise amplification, spatially undesirable noise distribution on output image also. Thus, they are only suitable for computation of global signals and not applied for specific area of interest (inside the image). In other hand, convolution algorithms in principle are complex due to the number of operations. Block spatial convolutions help in implementing local convolution or short convolutions without using FFT, which only estimate the direct convolution.

In some literature [3]-[5] it is noticed that the authors consider the filter size as a measure of the number of operations per output pixel (sample), for example an impulse response of M×N coefficients requires an M×N multiplications per sample. In contrary to this the authors in [6] conclude that the number of operations involved in a direct convolution is less for small arrays size and may take the form of a trapezoidal function. We have adopted this conclusion as a basis in our present work and have tested another function, namely a falling ramp. The advantage of the proposed algorithm in this paper resides in that it can be used also for measuring the degree of similarity between sub-images, which is not found in the previously mentioned literature. The algorithm integrates the convolution and correlation in one processing system. Partitioning(segmentation) of image to sub-images are applicable in techniques for hiding information and water marking where the spatial domain technique is the most common one [7], [8]. Available literature reveals a lack of details of partitioning and sequence of blocks. For this reason, the present work gains an importance. Exploiting image processing locality in cache pre-fetching [9] is one of the benefits of partitioning.

The authors in [10] propose a scheme of partitioning based on an anticlockwise division of a rectangular block image of size (64x512) for the iris ring into eight sub-images of size (64x64). A vector consisting of an ordered sequence of sub-image features are then extracted from the local information contained in the eight sub-images.

In [11] a partition fusion technique for multi-focus images is developed for improving the image quality. It compares conventional partition fusion technique for image sub- blocks taking even sizes only, which might result in asymmetry and a modified partition fusion method where sub-blocks are selected of the fused image based on their clarity measures. The clarity measure of an image sub block was determined by second order derivative of the sub image.

Algorithms [10],[11] are based on gray image. A gray image can provide enough information to identify different individuals, but the common in nature that colors are different.
Moreover, block sizes used are large and need large processing filters, which result in increased distortion at borders of images in addition to processing complexity [6].

It is not necessary to use length of filters radix-2 as in [5],[11], some time we need symmetric filter with odd length. Length of the filters directly affects computation time of analysis and re synthesis; shorter filters are favored in more cases. Linear phase response can be achieved by using symmetric filters. In algorithm [5] step number seven computes the inverse 2-D FFT, which needs to keep quarter the output array only, this means loss of time used to compute other three quarters.

Our scheme is expected to be clear in comparing it with a conventional convolution such as the sliding-window or double-circulant matrix [3],[12],[13]. It suggests a new block convolution where blocks are convolved independently with local filters for more reliable and fast processing. In our algorithm we, in advance compute desired parts (small sub-images), what we need only and without redundant operations. The rest of the paper is organized as follows: In Section 2, we provide the problem statement of this work, section 3 proposed design our algorithm, which consist of two steps partitioning and short convolution. The partitioning approach is described in subsection 3.1, short convolution is presented in sub section 3.2, while mathematical processing section is performed in section 3.3, experiment and performance evaluation of our approach are presented also. Section 4 deals with the evaluation results of simulation while section 5 represents discussion and conclusion. Appendix includes two-Matlab functions.

2. PROBLEM STATEMENTS

To realize short convolutions we need overlapping between sub-images to prevent discontinuity and to make real time processing. Extracting features such as point and edge detection and exploiting corner point in image alignment (registration) etc, needs local partitioning the entire image to sub-images. In other hand, short convolution helps to implement spatial and direct convolution with less operations, this gives exact solution rather than the FFT estimated solutions.

Now the problem is how to design the three distinct steps of the algorithm: step 1 is the partitioning of the input image to overlapped sub-images followed by a second partitioning of the previous one into smaller sub-images. Step 2 is the block convolution after a suitable flipping. The double partitioning is also useful for the extraction of local features or any other desired operations. Finally, the increase of pixels resulting after the convolution of the overlapped sub-images is compensated by converting the 6x6 sub-images at the output of the four processors into 4x4 sub-images. The algorithm is tested experimentally using the Matlab software package to evaluate its performance.

3. PROPOSED DESIGN ALGORITHM

This paper explores a new method for computing the 2-D convolution that is based on the algorithm represented by Figs. 1 and 2 using two main steps: partitioning (partitioning) and filtering as well as similarity measuring. Discrete input image is applied to the splitter input. Splitter shows itself as primary filter or discretizer. It determines the sub image size of a

specified periodic deterministic function. The four sampled output periodic blocks with specific size are applied to four convolution processors.

3.1 Partitioning

Assume that an X×Y grayscale (or color image layers) image (indices x and y as I(x, y)) (Fig. 1a) has to be partitioned into overlapped sub-images. Each has the dimension 6x6 as shown in Fig. 1b, X1×Y1= 6×6 and is a function of the indices x1 and y1; S1(x1, y1). As mentioned before, overlapping here reduces latency and prevents discontinuity resulting from partitioning, thus giving high efficiency. Each sub-image has to be partitioned to four small non-overlapped sub-images with indices x2 and y2; S2(x2, y2) the size of each of these sub-images must be equal to that of the spatial filter sizes; M×M=X2×Y2=3×3.

The first partitioning with overlap and non-overlap band (delta= Δ) is implemented using relations below:

$$\Delta = X1 - L = Y1 - L \tag{1}$$

$$x = x1 + (i - 1)\Delta, \qquad i = 1,2,\ldots(X - L)/\Delta) \tag{2}$$

$$y = y1 + (j - 1)\Delta, \qquad j = 1,2,\ldots(Y - L)/\Delta) \tag{3}$$

Where the indices i and j stand for the horizontal and vertical overlapped blocks, respectively. For example, Fig. 1a shows a partitioning with L = 2, Δ = 4.

The second partitioning is repeated in the same manner as before in order to give non-overlapped small sub-images (indices x2 and y2) with parameters:

$$x1 = x2 + (i1 - 1)\,M, \qquad i1 = 1,2,\ldots..X1/M \tag{4}$$

$$y1 = x2 + (j1 - 1)\,M, \qquad j1 = 1,2,\ldots..Y1/M \tag{5}$$

Where the indices i1 and j1 stand for the horizontal and vertical non-overlapped blocks, respectively, and (X-L)/(X1-L) = m and (Y-L) / (Y1-L) = n; m×n are the maximum number of overlapped sub-images, but X1/M=m1, Y1/M = n1; m1×n1 are the maximum number of non-overlapped sub-images. It should be to note that in Eqs. 4 and 5 the non-overlapping enlarges the band Δ to number M.

The partitioning and convolution processes can be performed in the one of the following manners:

1. Each time, a 6×6 overlapping block is subdivided into 4 small non-overlapping blocks each of 3×3 elements and the resulting four sub-blocks are then subjected to the convolution with the corresponding processors as shown in Figs. 1b, c and 2. This method is presented in this paper. The Matlab code for this method is given in Fig. 3.

2. The entire image of size X×Y is first divided into overlapped 6x6 blocks. Then the resulting image is subdivided into non-overlapped blocks of the size 3×3. Then each 4 blocks of the size 3×3 are subjected to convolution. The advantage of this method resides in its simplicity however it requires more memory space.

In many cases, partitioning helps to make processing implemented independently "in parallel "[5], [11]. In this paper, as we will see later on, the two points that need to be highlighted are: first, the four-processor banks in Fig. 2 are operating in parallel, and second, number of

operations is minimized by the factor of two. Taking into consideration these points, and in case of still image (single still picture), a fast image processing is expected. The issue of real time processing and video signal processing ' movie' requires further measures to ensure acceleration.

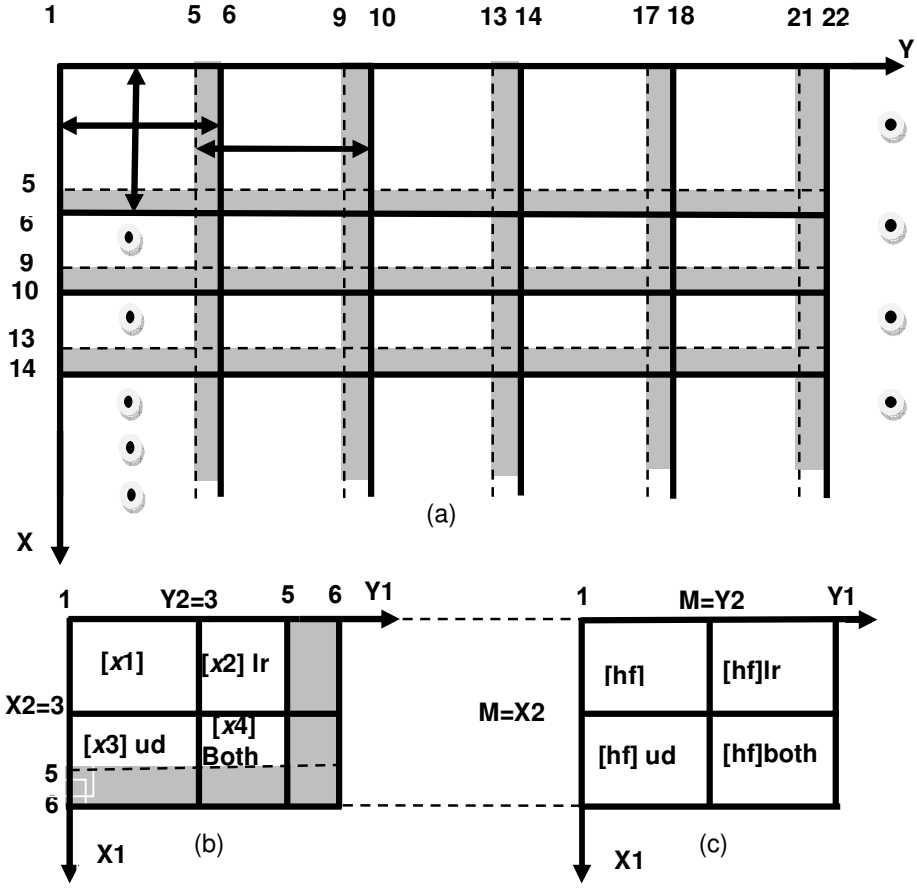


FIGURE 1: Layout of the Partitioning grid for the input Image Filtering pass.

(a) Formed input Image I(x, y) (b) First sub-Image with size X1×Y1 (Eq. 8)

(c) Impulse Responses of the System [hf] Partitioned to four M×M=3×3 Coefficients

3.2 Short Convolution Method

3.2.1 Description of Algorithm

The short convolution algorithm (length 3) as in Fig. 2 is used in this paper to develop an efficient implementation method for 2-D block processing. In our case each block filter inside the processor takes a set of input samples, (e.g. [x1]) and produces one pixel after convolution, (e.g. P_{11}). The size of the input set should coincide with the filter coefficients. It should be noted that an input samples set must decrease by one column from its preceding one and the same is applied to the filter coefficients. Referring to Fig. 2, the set [x1] is a 3×3-array, the set [x1c] is a 3×2-array (one column less than [x1]), and the set [x2c] is a 3×1-array (one column less than [x1c]. Similarly, the filter coefficients are denoted by [h3c], [h2c], and [h1c]. The convolution results in an array that is a function of the size; e.g., P11 [3×3], P12 [3×2], and P13 [3×1] (rows remain constant while columns change from 3 to1). These samples are returned as the first output row. The second output row is then obtained after omitting the first row from the original set [x1], and the third row from the original filter coefficient set [hf]. The new sets [x1r] and [h3r] are subjected to the convolution by repeating the previous processing method. The second row will be P21 [2×3], P22 [2×2], and P23 [2×1]. Finally, the third row is obtained after omitting the second row from the preceding input

samples set and coefficients set. Detailed description of the signal designation is illustrated in Table1 and Matlab function [y1] = newconv2DO(x1, hf). From the analysis of the processing, we observe that a change between output rows is obtained by decreasing the applied signal indices downwards and the system coefficients upwards. However, to get samples of the output row, the input signal decreases by a column from left to right and the system coefficients from right to left. In addition, it is seen that the number of operations per output pixel is varying with the output row. At the end of the process, the overall processor outputs are four sets each of 3x3 samples. Intermediate level samples, are concatenated to form convolved intermediate block of 6x6 samples. Finally the result is processed using Eq. 10 to obtain a 4x4 matrix sub-images (see Appendix for the Matlab function [yo] =calloutO (y1, y2, y3, y4)).

Signal	Description
[I]	Input Image (size(X×Y), overlapped gray level sub-images (S_{x1y1} size X1×Y1=6×6 Figs. 1 and 2)
[x4], [x3], [x2], [x1] = S_{x2y2}	Four 3×3 arrays resulted from X1xY1 sub image (see Eq. 6).
[hf]	Flipped filter impulse response
[x2c], [x1c], …	Matrices after omitting the second column, first column, …, respectively, from the preceding matrix, e.g. [x1].
[x2r], [x1r], …	Matrices after omitting the second row, first row, …, respectively, from the preceding matrix, e.g. [x1]
[h3c], [h2c], [h1c]	Matrices after omitting the third column, second column, or first column, respectively, from hf.

TABLE 1: Signals and Description

3.2.2 Mathematical Processing

The processor is ideally suited for real-time image processing applications, such as edge enhancement and edge detection.

For the convolution-transform for each pixel shown in Fig. 2, the coefficients $\boldsymbol{P_{r,c}}$ is obtained by computing the two-dimensional dot product of [hf] and each 3×3 sub image.
Convolution results as obtained from the first processor (y1) are as follows:

$$\boldsymbol{P_{r,c}} = \sum_{x2}^{M} \sum_{y2}^{M} \boldsymbol{hf_{x2,y2} S_{x2,y2}} \tag{6}$$

$$\boldsymbol{y1} = \begin{bmatrix} \boldsymbol{P_{11}} & \boldsymbol{P_{12}} & \boldsymbol{P_{13}} \\ \boldsymbol{P_{21}} & \boldsymbol{P_{22}} & \boldsymbol{P_{23}} \\ \boldsymbol{P_{31}} & \boldsymbol{P_{32}} & \boldsymbol{P_{33}} \end{bmatrix} \tag{7}$$

In Eq. 6 the size of the filter $(\mathrm{M}\times\mathrm{M})$ and the input signal must change from a maximum M (maximum degree of overlapping) to a minimum one (minimum degree of overlapping). In Fig. 2 the convolution blocks indicate that the size of arrays used for convolution decreases gradually from 3×3 to 1×1. This reflects that the number of operations is variable and follows a falling ramp function with the lag index, instead of a trapezoidal form as in [11]; i.e. processors work in one mode (decreasing) instead of three modes (increasing-constant-decreasing). Eq. 7 shows that, in our case the output convolution samples are equal to the input samples (Fig. 2); this is not the case in [5] where only a quarter of the output samples, estimated by inverse FFT, are used.
The remaining input sub-images are applied to the corresponding processors after appropriate flipping as given by Eq. 8 and require exactly the same operation steps as the first one. Flipping is applied differently in order to keep the algorithm applicable equally for all processors. Therefore, all processors perform same operation.

$$\boldsymbol{S_1(x1, y1)} = \begin{bmatrix} \boldsymbol{[x1]} & \boldsymbol{[x2 \times J]} \\ \boldsymbol{[J \times x3]} & \boldsymbol{[J \times (x4 \times J]} \end{bmatrix} \tag{8}$$

Here matrix J is the exchange matrix.

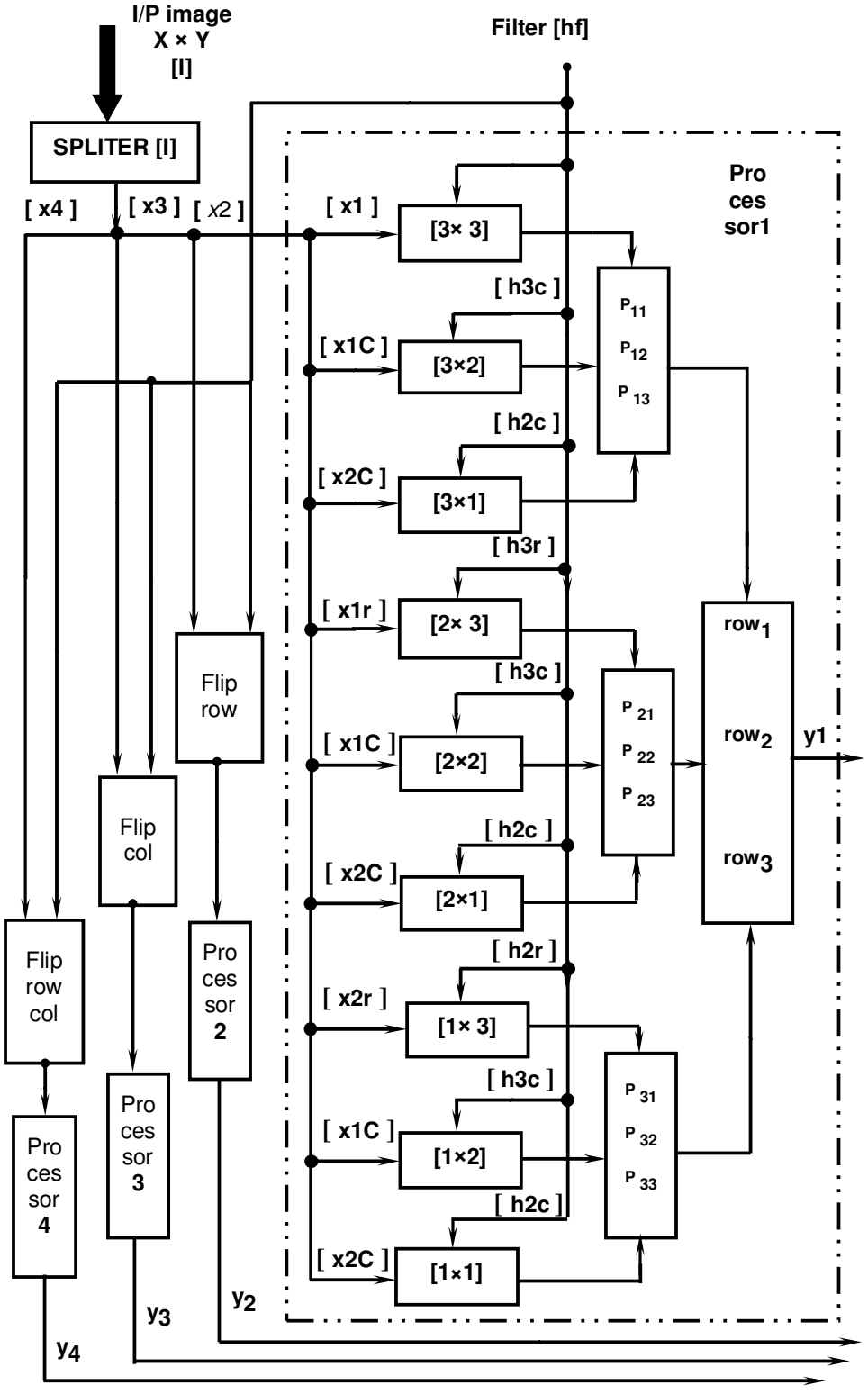


FIGURE2: Algorithm for small Convolution
Matlab function [y1] newconv2DO(x1, hf)

Similar operations as dictated by Eq. 8 are also performed with filter coefficients (Fig 1b). An intermediate stage is introduced according to Eq. 9. The external function y_{IT} successively converts the results of the four processors (6x6 pixels) to the final processing level, which is saved as output of 4x4 blocks (Eq. 10).

$$y_{IT} = \begin{bmatrix} y_1 & y_2 \\ y_3 & y_4 \end{bmatrix} \tag{9}$$

From y_{IT} we get first final sub image results as y_{1f} size 4×4.

$$y_{1f} = \begin{bmatrix} C_1 & P_{1,2}+P_{1,4} & P_{1,3}+P_{1,5} & C_2 \\ P_{2,1}+P_{4,1} & P_{2,2}+P_{2,4}+P_{4,2}+P_{4,4} & P_{2,3}+P_{2,5}+P_{4,3}+P_{4,5} & P_{2,6}+P_{3,6} \\ P_{31}+P_{5,1} & P_{3,2}+P_{3,4}+P_{5,2}+P_{5,4} & P_{3,3}+P_{3,5}+P_{5,3}+P_{5,5} & P_{3,6}+P_{5,6} \\ C_3 & P_{6,2}+P_{6,4} & P_{6,3}+P_{6,5} & C_4 \end{bmatrix} \tag{10}$$

Where C1, C2, C3 and C4 are the local corner convolved points of the four sub-images. Eq. 10 points out that the increase in the number of pixels due to overlapping (Fig. 1) is compensated here horizontally then vertically or parallel after convolution takes place using function[yo]=calloutO(y1,y2,y3,y4) (see Appendix). This is similar to the methods, which apply operations after convolution methods such as overlap-add, or overlap-save methods [12]. It should be noted that corner points have an important particularity that they do not add to neighboring pixels of sub-images and maintain properties of their own convolved sub-images. For sub image of size X2×Y2, to be convolved with a filter whose impulse response has a support M×M=X2×Y2=3×3 (Figs. 1b and c), the number of multiplications as indicated in processor1 of Fig. 2 is 36. If we apply the doubly circulant matrix 2-D filtering technique given by Eq. 11 [7] in Fig. 1, the number of multiplications would be constant and equal to nine per output sample. For an output matrix of 5×5 it reach 225 operations and for our case the output matrix 3x3 requires 81 operations, so the gain in simplicity more than 2.

$$\begin{bmatrix} y0 \\ y1 \\ y2 \\ y3 \\ y4 \end{bmatrix} = \begin{bmatrix} H0 & H4 & H3 \\ H1 & H0 & H4 \\ H2 & H1 & H0 \\ H3 & H2 & H1 \\ H4 & H3 & H2 \end{bmatrix} \begin{bmatrix} x0 \\ x1 \\ x2 \end{bmatrix} \tag{11}$$

In Eq. 11 H0 to H4 are basic matrices of dimension 3×3, and.x0, x1, x2 are input column vectors of length 3 stacked to give column–ordered vector length 9. Output vectors y0 to y4 are columns of length 5 and represent 2-D array size 5×5.

Noise labeled with convolution can be reduced by well-known linear or nonlinear techniques. Nonlinear methods often provide good noise cleaning technique in which each pixel is compared to the average of its eight neighbors. According to Eq. 12.

$$[C - 1/8\sum_{i=1}^{8} S_i] > T, \quad then \quad C = 1/8\sum_{i=1}^{8} S_i \tag{12}$$

Each core pixel C is compared to the average of its eight neighbors S_i. If the level of the difference is greater than some threshold level **T**, the pixel is judged to be noisy, and it is replaced by its neighborhood average.

Pseudo code below (Fig. 3) shows how we exploit previous partitioning for implementation block -convolution.

```
Clc; clear;
[I]=imread ('original image');
Ip=I(control size of input image);
A=double(Ip);
[X,Y]=size(A);
H=6;V=6;%size of sub-block1
p=2;%overlap elements
m=floor((X-p)/(H-p));%number of sub-blocks in horizontal
n=floor((Y-p)/(V-p));%number of sub-blocks in vertical
yot = cell(m, n);
for(i=1:m) % Global search(for all)
  for(j=1:n)
     for(h=1:H) % making sub-image 6x6 (Local sub-block1-overlapped)
        for(v=1:V)
          x=h+(i-1)*H-p*(i-1);
          y=v+(j-1)*V-p*(j-1);
          z{i,j}(h,v)=A(x,y);
          ss=z{i,j};%cell to array
        end
     end
    hf=[1 1 1;1 1 1;1 1 1];%any system
    x1=ss(1:3,1:3);A1 = newconv2DO(x1,hf);dad{1,1}=A1;
    x2= ss(1:3,4:6);A2=newconv2DO(fliplr(x2),fliplr(hf));dad{1,2}=A2;
    x3= ss(4:6,1:3);A3=newconv2DO(flipud(x3),flipud(hf));dad{2,1}=A3;
    x4=ss(4:6,4:6);A4=newconv2DO(flipud(fliplr(x4)),flipud(fliplr(hf)));dad{2,2}=A4;
    yo = calloutO(dad{1,1},dad{1,2},dad{2,1},dad{2,2});yot{i,j}=yo;
   end
end
y=conv2(Ip,h);%Matlab conv
L=cell2mat(yot);F=max(min(L));N=L/F;F2=max(min(y));
subplot(1,2,1), imshow(N);subplot(1,2,2), imshow(y/F2);
```

FIGURE3: Partitioning and Convolution Matlab code

4. EXPEREMENTAL RESULTS and PERFORMANCE EVALUATION

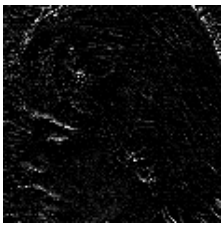
Grayscale image (possibly color image layers) of size 1206×1206 is used in the experiments of this paper (see Fig. 3). An original image as shown in Fig. 4a is subjected after partitioning to filtering with folded impulse response hf=[1, 2, 1; 0, 0, 0;-1, -2, -1] and without partitioning (Matlab convolution using either double circulant matrix or sliding window) with impulse response hm=[-1, -2, -1; 0, 0, 0; 1, 2, 1]. For enhancing output image point, gray level modification is undertaken i.e. dividing output image by F=max (min (output image matrix)).
The results of extracting horizontal Sobel high pass edge components of original image are shown in Figs. 4b and 4c.

Fig.4c (right) results due to the conventional convolution, whereas Fig.4b results due to the proposed block convolution. Comparing the two results, it is observed, that the image in Fig. 4b shows better local features than that in Fig4.c where local edge detail (inner borders) has been lost, and a distortion as global features (outer borders) of $\Delta x \times \Delta y$ = hf -1×hf -1 has been introduced.

Our aim here is to evaluate the results of both two-convolution methods shown before by using two objective evaluation criteria: spatial quality correlation coefficient [4,14]; the peak signal-to-noise ratio (PSNR) and the corresponding Mean Square Error (MSE), computed using Matlab functions below:

a - Orignal input image

b



c

FIGURE4: Image with a Sobel Horizontal edge detector
b-Output image due to proposed method c-Output image due to classical methods

$$c = corr2(y1\_n, y) \text{ and } [psnr, mse] = psnr\_mse\_maxerr(y1\_n, y) \quad (13)$$

Here ***y*** and ***y1_n*** are variables that represent images before addition of Gaussian noise (Fig4-c) and after it, respectively. Same process is repeated for the results of the proposed algorithm (Fig4-b). For illustration, the results of evaluation are shown together in Table-2 and Fig. 5 .

Higher PSNR of image means better quality of the measured image and vice versa to MSE. The PSNR results by the proposed method are higher than that by classical methods. The difference is around 7dB, this is why images resulted from proposed method are visually distinguishable better. Regarding this evaluation, we can say that there is a matching between objective and subjective evaluations.

Spatial quality of the proposed method is higher than that of the conventional convolution techniques mentioned above this page. Under the effect of white Gaussian, SNR from 0 db to 30 db, the performance of the proposed algorithm does not change and it preserves the same properties attained from noise free source. Moreover, the proposed technique could also generate high quality images when the noisy images de-noised with thresholding (see Eq, 12).The proposed algorithm guarantees that the quality of the resulting convolved image is always satisfactory for noise free source, noisy and de-noised convolved images.
The presented partitioning algorithm can be useful in many applications such as image classification, fusion [14] and identifications [9] and others.

N	0	30
psnr- conventional method	19.61	19.63
psnr-proposed method [db]	26.16	26.3
mse- conventional method	712.1	708.5
mse-proposed method	157.2	152.4

TABLE2: Evaluation results (min and max) from PSNR and MSE

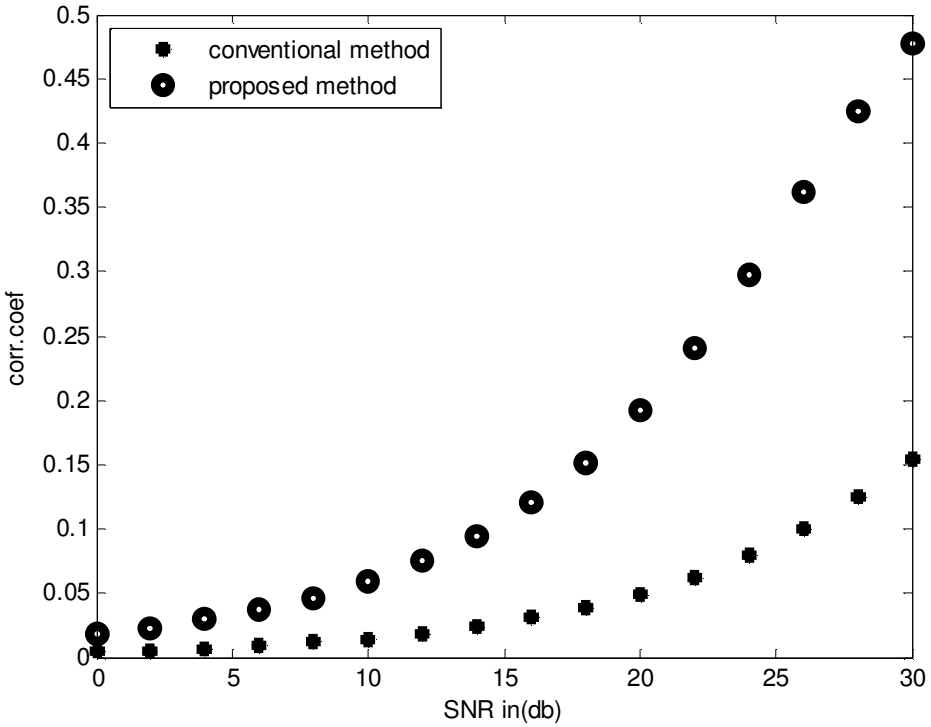


FIGURE 5 : Correlation coefficient for conventional and proposed methods

5. DISCUSSION and CONCLUSION

In this paper, an algorithm has been designed for decreasing complexity and minimizing number of operations enough to reconstruct the input image. We have presented and implemented a 2-D algorithm for the sub-image (block) digital filtering or cross correlation measuring based on double partitioning of large 2-D image into smaller sub-images. Moreover, the decomposition of computation into parallel processing as depicted in processor1 has reduced the complexity. The spatial filtering has been used to implement filtering stage. Since the convolution and correlation are two related operations, so this algorithm is common for them except that in correlation a mask needs not to be rotated by 180 degree prior correlations with the input image.

Comparison of this algorithm with Matlab convolution function or cross-correlations functions gives zero error in gray-level values with small different in sizes**.** This algorithm is designed so that the side effects of convolution at the boundaries of image are reduced. This is accomplished by neglecting a surrounding frame of size 2x2, which is equal to the filter size minus one (M-1). A positive effect of this is translated by a reduction in complexity, which raises the advantages of this algorithm and gives an example of zooming-in of an entire image as shown in Fig. 4b of Appendix . This can be interpreted by the fact that a local scale reduction of each small sub-images results in a change of distance from the sensor.
The proposed algorithm can be used in various applications as follows:

- Hiding information and specially with hiding technology based on block-partitioning strategy.
  We can embed hiding process before or after convolution stage. Possibly, with some arrangement both convolution, hiding processes and geometric transformation may be implemented in parallel. It is possible to approximate complex geometric transformation by partitioning an image.
- Image registration: corner points are specific points determined by all pixels of input sub image and have maximum number of operations, so they can be used as control points.

These above mentioned applications constitute the objects for future research work.

6. APPENDIX

Matlab-Functions: new convolution (newconv2DO) and call out results of convolution (callout O).

```
Function [y1] =newconv2DO(x1, hf)
p11=sum (sum (x1.*hf)); %1st o/p convolved sample of 1st row
xr=x1(1:3, 2:3); %omit 1st column [x1c] from [x1] (see Fig.2)
hr=hf (1:3, 1:2); %omit 3ed column [h3c] from [hf]
p12=sum (sum (xr.*hr)); %2ed o/p convolved sample
xrr=xr (1:3, 2); %again omit column2 from previous matrix [xr]
hrr=hr (1:3, 1); %again omit column2 from previous matrix [hr]
p13=sum (sum (xrr.*hrr)); %3ed o/p convolved samples
X1= [p11 p12 p13]; %call results row1
xr2=x1(2:3, 1:3); %omit1st row [x1r] from original [x1]
hr2=hf (1:2, 1:3); %omit 3ed row [h3r] from original [hf]
p21=sum (sum (xr2.*hr2)); %1st o/p convolved sample of 2nd row
xr3=xr2 (1:2, 2:3); %omit 1st column [x1c] from previous matrix [xr2]
hr3=hr2 (1:2, 1:2); %omit 3ed column [h3c] from previous matrix [hr2]
p22=sum (sum (xr3.*hr3));
xr4=xr3 (1:2, 2);
hr4=hr3 (1:2, 1);
p23=sum (sum (xr4.*hr4));
X2= [p21 p22 p23]; %call results row2
xr5=xr2 (2, 1:3); %omit 2nd row [x2r] from previous matrix [xr2]
hr5=hr2 (1, 1:3); %omit 2nd row [h2r] from previous matrix [hr2]
p31=sum (sum (xr5.*hr5));
xr6=xr5 (1, 2:3); hr6=hr5 (1, 1:2);
p32=sum (sum (xr6.*hr6));
xr7=xr6 (1, 2); hr7=hr6 (1, 1);
p33=sum (sum (xr7.*hr7));
X3= [p31 p32 p33]; %call results row3
y1=[X1; X2; X3]; %O/P of 1st processor
End
```

```
Function [yo] =calloutO (y1, y2, y3, y4)
Yf=zeros (6, 6);
Yf (1:3, 1:3) =y1;
Yf (1:3, 4:6) =y2;
yf (4:6, 1:3) =y3;
yf (4:6, 4:6) =y4;
yf(1:6,2)=yf(1:6,2)+yf(1:6,6);%hor
Yf (1:6, 3) =yf (1:6, 3) +yf (1:6, 5);
yf(2,1:4)=yf(2,1:4)+yf(6,1:4);%ver.
Yf (3, 1:4) =yf (3, 1:4) +yf (5, 1:4);
Yo=yf (1:4, 1:4);
End
```

7. REFERENCES

[1] S. E Umbaugh. *Computer Imaging digital image analysis and processing*. CRC press Book, 2005, PP.659.

[2] A. R. Weeks. *Fundamentals of Electronic Image Processing*. New Delhi: University of Central Florida and Prentice-Hall of India, 2005, pp.570.

[3] A. K. Jain. *Fundamentals of Digital image processing*. New Delhi: Prentice-Hall of India, 1997, PP.569.

[4] R. C. Gonzalez, R. E. Woods , S. L. Eddins. *Digital image processing using Matlab*. Pearson Education. Inc, 2005(2002), PP.310.

[5] K. Berberidis. An efficient partitioning-based scheme for 2-D convolution and application to image restoration. *IEEE*, PP.843-846, 2002.

[6] L. Ismail , D. Guerchi. "Performance Evaluation of convolution on the cell broad band engine processor". *IEEE Transactions on parallel and distributed systems*, vol. 22, no.2, pp.337-351. 2011.

[7] T. C. Lu, S. R. Liao, P.L. Chen, C.C. Chang, Z.H. Wang. "Information hiding technology based on block-partitioning strategy". ISECS international colloquium computing, communication, control, and management, pp.500-505. 2009.

[8] E. D. Gelasca."Full-reference objective quality metrics for video watermarking, video partitioning and 3D model watermarking". PhD thesis, PP.187, 2005.

[9] R. Cuccchiara, M. Piccardi. "Exploiting image processing locality in cache pre-fetching .In High Performance Computing". 2002.

[10] Li Ma, Y. Wang, T. Tan**.** *"*Iris recognition based on multichannel Gabor filtering". The 5th Asian Conference on Computer Vision, pp.1-5, 2002.

[11] **D**. Agrawal, D. Ali, J.Singhai**.** "A modified partition fusion technique of mulltifocus images for improved image quality". Conference - *Bioinformatics and image*, special issue on ICIT, vol. 4 No. 3, pp.658-663, 2009.

[12] T. Bose. *Digital signal and image processing*. John Wiley & sons.inc, pp.706, 2004.

[13] M. Sonka, V. Hlavac, R. Boyle. *Image processing analysis and machine vision*. Chapman and Hall, PP-555, 1994.

[14] M. Holia, V.K.Thakar. **"**Image registration for recovering affine transformation using Nelder Mead Simplex method for optimization." *IJIP-Journal*, vol. 3, Iss.5, pp.218-228, 2011.

Edge Extraction With an Anisotropic Vector Field Using Divergence Map

Giuliani Donatella *giulianidonatella@libero.it*
Scientifc-Didactic Polo of Rimini
University of Bologna
Via Angherà 22, Rimini, Italy

Abstract

The aim of this work is edge detection by a deformable contour procedure, using an external force field derived from an anisotropic flow, with different external and initial conditions. By evaluating the divergence of the force field, we have generated a divergence map associated with it in order to analyze the field convergence. As we know, divergence measures the intensity of convergence or divergence of a vector field at a given point, so by means level curves of the divergence contour map, we have automatically selected an initial contour for the deformation process. The initial curve must include areas from which the vector field diverges pushing it towards edges. Furthermore the divergence map brings out the presence of curves pointing to the most significant geometric parts of boundaries corresponding to high curvature values, in this way it will result better defined geometrical shape of the extracted object.

Keywords: Edge Extraction, Active Contours, Anisotropic Flow, GGVF

1. INTRODUCTION

Image segmentation and boundary extraction are diffusely handled topics of research in image processing. These problems have been dealt with in various forms [1], [2] [3], [4],[5],[6],[7] nevertheless the initial conditions and the ability to change the topology of the evolving curve are the main limitations for most of these methods. Edge detection may be realized by a deformable contour process with active contours or surfaces embedded within an image domain that move under the influence of internal and external forces [1],[6]. Internal forces, which are defined within the curve or surface itself, are designed to smooth the model during deformation. External forces, which are computed from image data, are used to move the model toward researched features in the image [8],[9],[10]. In this paper the external force field is derived from an anisotropic flow, in a more general framework respect to the GGVF [10],[11], in order to come to an overall view of the field flow, analyzed through its divergence values. A careful overview of divergence map, showing the divergence of the external force field, gives us the opportunity to circumscribe areas within which we can place an initial contour that may be subsequently deformed.

2. PRELIMINARIES

In the traditional parametric models an active contour or snake, expressed explicitly by parametric equations $\vec{x}(s) = (x(s), y(s)), \quad s \in [0,1]$, is defined within a given image *I(x,y)* and subjected to modifications under the action of forces, until the evolving curve fits well into the final contour [1],[7]. The final shape of the contour to be extracted will be such as to minimize an energy functional associated with it, so given:

$$E(\vec{x}(s)) = \int E_{Int}(\vec{x}(s)) + E_{Ext}(\vec{x}(s))ds \qquad (1)$$

the first term is the *internal energy* that expresses a priori knowledge of the model in relation to the degree of flexibility of an active contour:

$$E_{Int}(\vec{x}(s)) = \int \frac{1}{2}\left[\alpha(s)\cdot\left|\frac{d\vec{x}}{ds}\right|^2 + \beta(s)\cdot\left|\frac{d^2\vec{x}}{ds^2}\right|^2\right]ds \qquad (2)$$

the term $\alpha(s)$ controls the contour tension, while $\beta(s)$ regularises its rigidity. The second term $E_{Ext}(\vec{x}(s))$ represents the *external energy* related to a potential energy function $P(\vec{x}(s))$ deriving from the image *I(x,y),* whose local minima correspond to edges of the features to be extracted. By using a variational approach [12], the contour that minimizes the total energy must satisfy the Euler-Lagrange equation:

$$-\frac{d}{ds}\left(\alpha\frac{d\vec{x}}{ds}\right) + \frac{d^2}{ds^2}\left(\beta\frac{d^2\vec{x}}{ds^2}\right) + \nabla P = 0 \qquad (3)$$

where ∇ is the gradient operator. Through energy minimization we derive a resolution of a static problem. We could construct a deformable model able to create a geometrical shape that evolves over time, by introducing a time-variable parametric equation $\vec{x}(s,t) = (x(s,t), y(s,t))$. To this end, indicating with $\mu(s)$ and $\gamma(s)$ the density of mass and the damping coefficient respectively, instead of equation (3) we obtain:

$$\mu\frac{\partial^2\vec{x}}{\partial t^2} + \gamma\frac{\partial\vec{x}}{\partial t} - \frac{\partial}{\partial s}\left(\alpha\frac{\partial\vec{x}}{\partial s}\right) + \frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2\vec{x}}{\partial s^2}\right) = -\nabla P \qquad (4)$$

The equilibrium is reached when internal and external forces are equal, this implies achieving a steady state in which time derivatives will become null.
When the solution $\vec{x}(s,t)$ of equation (4) stabilizes, we carry out the result of equation (3). Neglecting the inertial term and thus the second order derivatives, considering dumping, elasticity and rigidity as constant functions, the numerical solution of equation (4), can be reduced to that one of the dynamic equation:

$$\begin{cases} \gamma \dfrac{\partial \vec{x}(s,t)}{\partial t} = \alpha \dfrac{\partial^2 \vec{x}(s,t)}{\partial s^2} - \beta \dfrac{\partial^4 \vec{x}(s,t)}{\partial s^4} + \vec{F}_{Ext}(\vec{x}) \\ \vec{x}(s,0) = \vec{x}_0(s) \end{cases} \quad (5)$$

where $\vec{x}_0(s)$ is an initial contour. In order to reduce the sensitivity of this model to the initial conditions, edge extraction may be realized using a different class of external forces, the GGVF force field or Generalized Gradient Vector Flow [10],[11],[13],[14] obtained by solving a diffusion problem. In the GGVF contour generation, the external force field will be referred as:

$$\vec{F}_{Ext}^{GGVF} = \vec{v}(\vec{x}) \quad (6)$$

Using the calculus of variations [12], the GGVF force field $\vec{v}(x, y)$ can be found by solving the following diffusion equation:

$$\vec{v}_t = g(|\nabla f|) \cdot \nabla^2 \vec{v} - h(|\nabla f|) \cdot (\vec{v} - \nabla f) \quad (7)$$

where ∇^2 is the Laplacian operator, $\nabla f(x, y)$ is the gradient of the *edge map f(x,y),* derived from the gradient of brightness function *I*(x,y), that can be computed as $f(x, y) = |\nabla(I(x, y)|$ or using any other edge detector. The edge map gradient are vectors directed towards boundaries to detect with norms significantly different from zero in proximity of them. In equation (7) $g(|\nabla f|)$ and $h(|\nabla f|)$ are space varying weighting functions, being dependent on absolute gradient of edge map, generally not uniform. The function $g(|\nabla f|)$ will be monotonically non-increasing, since the vector field $\vec{v}(x, y)$ will be weakly variable far from edges to be extracted where image intensities are uniform. On the other hand, $h(|\nabla f|)$ should be monotonically non-decreasing, therefore, when $|\nabla f|$ is large, the vector field $\vec{v}(x, y)$ should have a trend nearly equal to ∇f .

The main shortcomings to be overcome in the generation of deformable contours are: 1) the initialization problem, i.e. the excessive sensitivity to shape and initial position of an initial curve 2) the weak convergence of models towards edges, especially in regions with highly variable concavities 3) the capture range, i.e. the size of area inside which an active contour can be initialized to be able achieving the desired boundary. We would like to explore a method that tries to overcome one of the main drawbacks for most of the approaches that were introduced until now: the initialization problem. The convergence of arbitrary initial contours for the GGVF leads only partially to the expected results, giving seemingly incomprehensible outcomes, as we can see in Fig.1a, Fig1b and Fig.2a, Fig2b for two test images. This paper is concerned with the problem of identifying automatically [15],[16],[17] an initial contour using the divergence of a force field derived from an anisotropic vector flow.

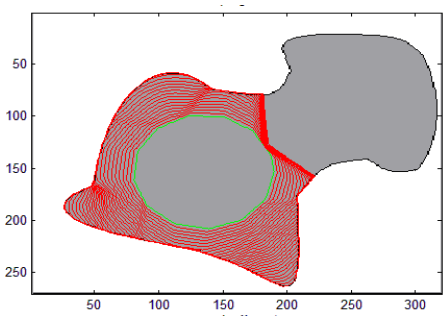


FIGURE 1a: Initialization problem

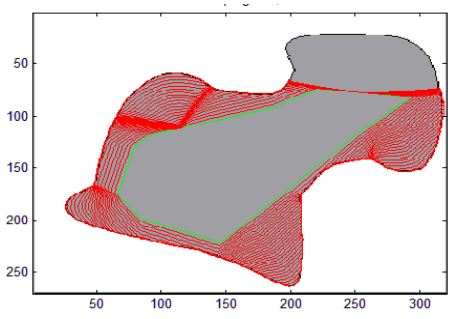


FIGURE 1b: Initialization problem

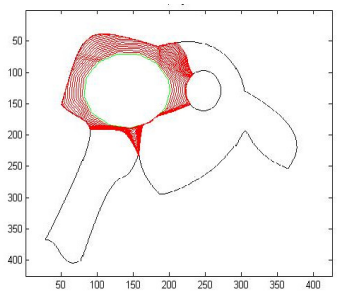


FIGURE 2a: Initialization problem

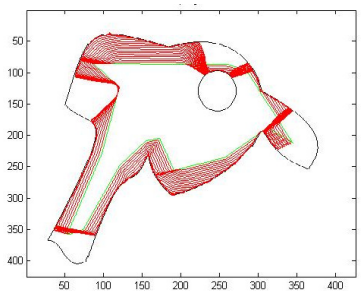


FIGURE 2b: Initialization problem

3. THE DIVERGENCE MAP OF AN ANISOTROPIC VECTOR FIELD

In this section we suggest a more general analysis of the diffusion process for an external force field. To this aim, we could see the solution of the GGVF equation (7) as the steady state of the following generalized parabolic equation:

$$\begin{cases} \vec{v}_t = div\left(g\left(|\nabla f|\right)\cdot\nabla\vec{v}\right)+\vec{F}(\vec{v}) \\ \vec{v}(x,\, y,0)=\vec{v}_0(x,\, y) \end{cases} \qquad (8)$$

where div is the divergence operator of the diffusion component and $\vec{F}(\vec{v})$ is a term generating an external force for the diffusion process of $\vec{v}(x,\, y)$, g(.) is the conduction coefficient that must be null or tend to zero at boundaries. So, in adiabatic condition hypothesis the vector flow is realized inside or outside the region. If the scalar-valued diffusivity function g(.) is monotonically non-increasing the diffusion has been stopping across the edges to be extracted. We obtain the GGVF field [11] if $\nabla g\left(|\nabla f|\right)\cdot\nabla\vec{v}=0$ $\vec{F}(\vec{v})=-h\left(|\nabla f|\right)\cdot(\vec{v}-\nabla f)$

$g\left(|\nabla f|\right)=e^{-\frac{|\nabla f|}{k}}$ $h\left(|\nabla f|\right)=1-g\left(|\nabla f|\right)$

with initial conditions $\vec{v}_0 = \nabla f$ and *k* as a constant positive value. If we consider an anisotropic flow for the vector field $\vec{v}(x, y)$ expressed as follows:

$$\begin{cases} \vec{v}_t = div\left(g\left(|\nabla f|\right) \cdot \nabla \vec{v}\right) \\ \vec{v}(x, y, 0) = \nabla f \end{cases} \quad (9)$$

where $\vec{F}(\vec{v}) = 0$, we carry out results rather similar to those of the GGVF field, given that the term $\vec{F}(\vec{v}) = -h(|\nabla f|) \cdot (\vec{v} - \nabla f)$ tends to zero whether near edges, for the chosen initial conditions, or far from them, since the function $h(|\nabla f|)$ is getting irrelevant, consequently the overall contribution of $\vec{F}(\vec{v})$ is not significant. But now considering the problem through the parabolic equation (9), we could interpret the process as a field flow from boundaries toward inside or outside without crossing edges and pointing to them because initially equal to the edge map gradient (Fig.3b). We could note from Fig.3a that the vector field can capture object boundaries from either sides. As we know, divergence is a measure of a field convergence or divergence at a given point by means signed scalar values. So if we compute the divergence of $\vec{v}(x, y)$, we would obtain negative values corresponding to object boundaries towards which the vector field converges, sinks of the flow, whereas positive values would define regions from which the field flow springs out, the flow sources.

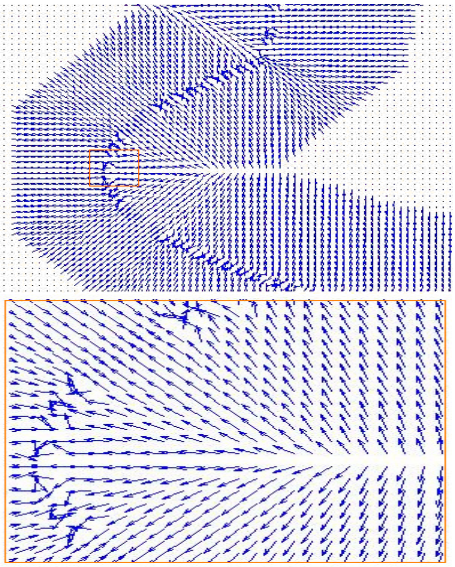


FIGURE 3a: The vector field $\vec{v}(x, y)$

FIGURE 3b: The vector field $\vec{v}(x, y)$

As a consequence, the analysis of divergence of the external force field $\vec{v}(x, y)$ may be very useful to feature detection, since it allows to demarcate regions from which the flow has been originated. Furthermore in correspondence of long and narrow boundary indentations, the field convergence towards opposite very near sides gives rise to zones of high positive divergence (Fig.3a,3b), hence divergence values may well highlight them. An efficient numerical scheme to resolve the anisotropic diffusion equation (9) may be that one proposed by Perona-Malik [18] and realized recurring to the 4-neighbors discretization of the Laplacian operator:

$$\vec{v}_{i,j}^{\,t+1} = \vec{v}_{i,j}^{\,t} + \lambda\left[g_N \nabla_N \vec{v}_{i,j}^{\,t} + g_S \nabla_S \vec{v}_{i,j}^{\,t} + g_E \nabla_E \vec{v}_{i,j}^{\,t} + g_W \nabla_W \vec{v}_{i,j}^{\,t}\right]$$
$$\nabla_N \vec{v}_{i,j} = \vec{v}_{i-1,j} - \vec{v}_{i,j}$$
$$\nabla_S \vec{v}_{i,j} = \vec{v}_{i+1,j} - \vec{v}_{i,j} \qquad (10)$$
$$\nabla_E \vec{v}_{i,j+1} = \vec{v}_{i,j+1} - \vec{v}_{i,j}$$
$$\nabla_W \vec{v}_{i,j} = \vec{v}_{i,j-1} - \vec{v}_{i,j}$$

with $\lambda = \frac{1}{4}$, and values of the edge-stopping function g(.) evaluated through a gradient approximation of $|\nabla f|$ along horizontal and vertical directions, such as:

$$g_N = g\left(\left|f_{i-1,j} - f_{i,j}\right|\right)$$
$$g_S = g\left(\left|f_{i+1,j} - f_{i,j}\right|\right)$$
$$g_E = g\left(\left|f_{i,j+1} - f_{i,j}\right|\right)$$
$$g_W = g\left(\left|f_{i,j-1} - f_{i,j}\right|\right)$$

The anisotropic vector field so obtained, from now on AVF, is partially shown in Fig.4a and Fig.4b for different numbers of time iterations, it results very similar to that one evaluated with the GGVF approach already seen. At first we could note that the size and the shape of capture range vary with the number of iterations performed in equation (9).

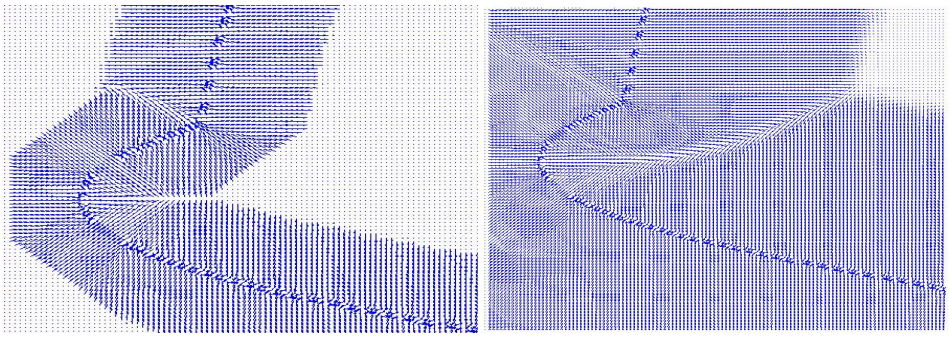


FIGURE 4a: AVF field from equation (9) after 20 iterations

FIGURE 4b: AVF field from equation (9) after 140 iterations

If now we consider equation (8) with an external force $\vec{F}(\vec{v})$ significantly different from zero, the convergence of $\vec{v}(x, y)$ to the steady state will result faster than the GGVF solution. By using the diffusion equation:

$$\begin{cases} \vec{v}_t = div\left(g\left(|\nabla f|\right)\cdot\nabla\vec{v}\right)+\vec{F}(\vec{v}) \\ \vec{v}(x, y, 0)=\nabla f \end{cases} \qquad (11)$$

with, for example, an the external force $\vec{F}(\vec{v})=g\left(|\nabla f|\right)\cdot(\vec{v}-\nabla f)$ we will be able to speed up the convergence process, because $\vec{F}(\vec{v})$ is approximately null near edges but increases as moving away from them. This implies an extension of capture range for an equal number of time iterations. In Fig.5a and Fig.5b we could compare the AVF field derived from equation (11) with GGVF field obtained through equation (7). With the same number of iterations it has been realized an increase of capture range, the region inside of which must be placed any initial contour. Consequently the evaluation of the AVF vector field will be less time consuming respect to the GGVF because we could choose a reduced number of iterations to achieve similar results. In Fig.5c and Fig.5d we have compared the results obtained by AVF field with those of GGVF using the same number of iterations (in the case in point 80) and the same initial polygonal. Recurring to 120 iterations to research the solution of equation (7), the GGVF field has been able to detect edges completely.

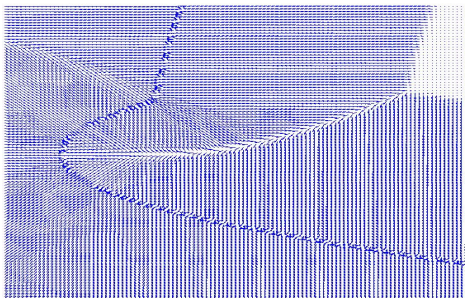


FIGURE 5a: AVF field from equation (11) after 80 iterations

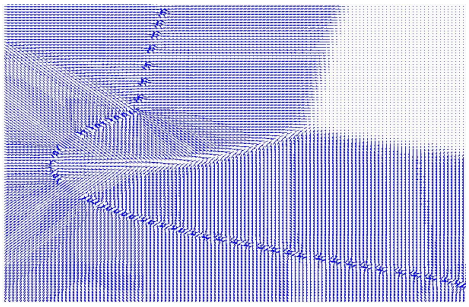
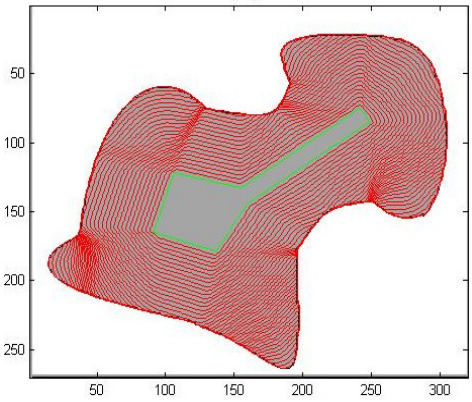


FIGURE 5b: GGVF field from equation (7) after 80 iterations



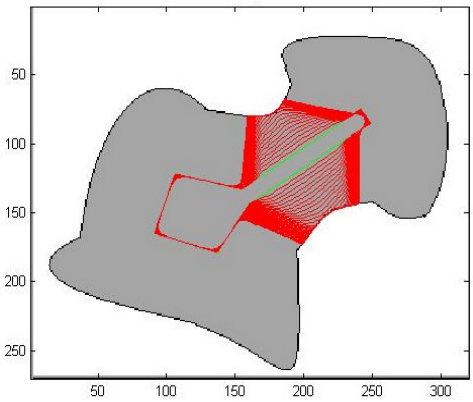
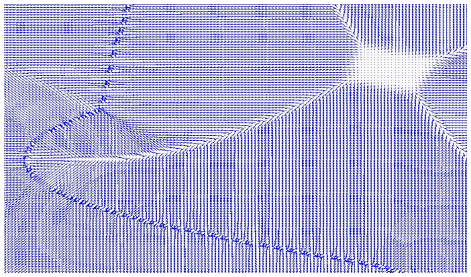


FIGURE 5c: AVF field from equation (11) **FIGURE 5d:** GGVF field from equation (7) with the same initial contour after 80 iterations

As an alternative approach we could evaluate the vector field $\vec{v}(x, y)$ using diffusion equation (11) but recurring to different external forces and initial conditions, so we have:

$$\begin{cases} \vec{v}_t = div\left(g\left(|\nabla f|\right)\cdot\nabla\vec{v}\right)+\vec{F}\left(\vec{v}\right) \\ \vec{v}(x, y,0)=0 \end{cases} \qquad (12)$$

where now $\vec{F}\left(\vec{v}\right)$ is an external attractive force directed towards edges, for example of the type $\vec{F}(\vec{v})=h(|\nabla f|)\cdot(\vec{v}-\nabla f)$ that stops acting when the vector field, initially null everywhere, tends to the edge map gradient. Again, by a comparison between the AVF field generated with equation (12) and the GGVF field after the same number of iterations, we could verify an acceleration of the convergence process (Fig.6a,6b). Therefore the area, within which the field is able to act on a deformable initial contour pushing it towards edges, is larger with the AVF field of equation (12) than GGVF. As a consequence, we need a smaller number of iterations to obtain a vector field as effective as the GGVF.



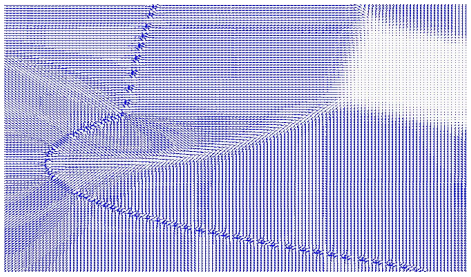


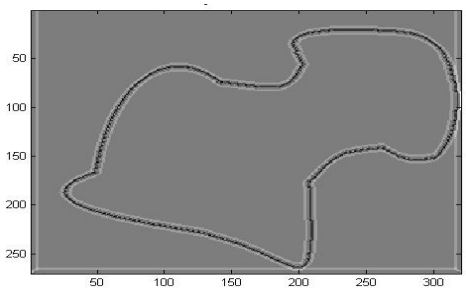
FIGURE 6a: AVF field from equation (12) after 140 iterations

FIGURE 6b: GGVF field from equation (7) after 140 iterations

After these considerations, we may suggest a method to generate automatically an initial curve, that could be able to adequately reconstruct the final contour. Indeed the effects due to initialization problem and capture range extension are reduced using the divergence map of the normalized force field, defined as:

$$I_D(x, y, t) = div\left(\frac{\vec{v}(x, y, t)}{|\vec{v}(x, y, t)|}\right)$$

In the following figures it has been shown divergence maps corresponding to vector fields resulted as numerical solutions of equation (9) at different times t, i.e. for different numbers of iterations, referred to the first test image (Fig.7a,7b,7c,7d) and to the second one (Fig.8a,8b,8c,8d) respectively:



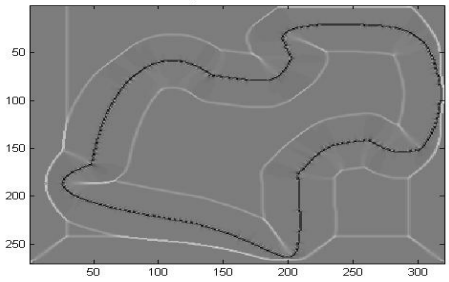


FIGURE 7a: Divergence Map of AVF from equation (9) after 0 iterations

FIGURE 7b: Divergence Map of AVF from equation (9) after 30 iterations

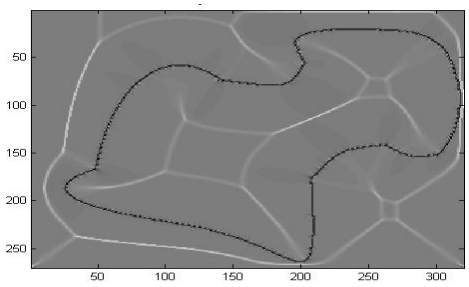


FIGURE 7c: Divergence Map of AVF from equation (9) after 100 iterations

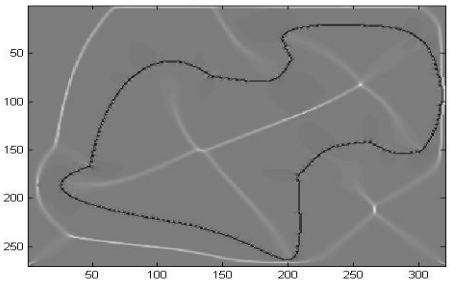


FIGURE 7d: Divergence Map of AVF from equation (9) after 240 iterations

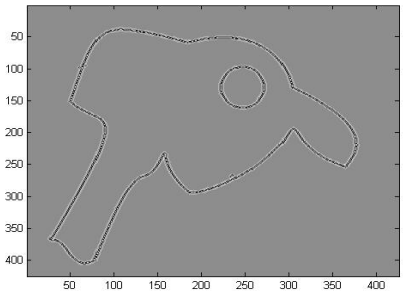


FIGURE 8a: Divergence Map of AVF from equation (9) after 0 iterations

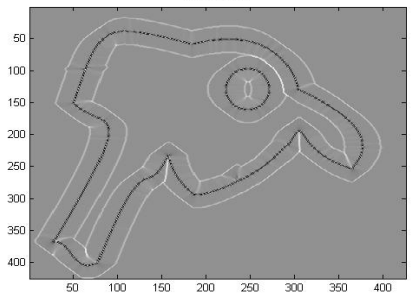


FIGURE 8b: Divergence Map of AVF from equation (9) after 20 iterations

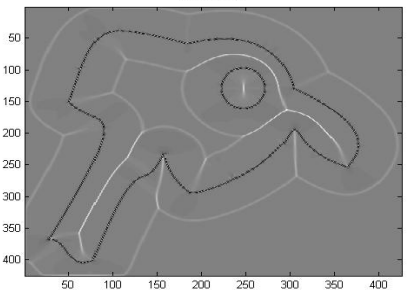


FIGURE 8c: Divergence Map of AVF from equation (9) after 80 iterations

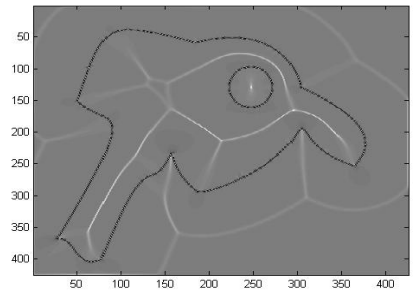


FIGURE 8b: Divergence Map of AVF from equation (9) after 200 iterations

By an accurate analysis of the shown figures, we may point out that each divergence map is characterized by a gray background with divergence values near zero, dark curves with negative divergence corresponding to edges, towards which the vector field converges, and a system of light curves with positive values, defining regions from which the vector field comes out. These regions are varying with the number of iterations after which the flow process has been interrupted.

The zones between opposite values of divergence define the capture range size. Moreover, by increasing the iteration number, the sides of areas that delimit parts of image inside which the field is null, collapse each other, especially in conjunction with deep and narrow concavities (see Fig.8b and Fig.8c). Thereby these curves, corresponding to positive divergence values, remain as traces of those edge sections with high curvatures and concur in forming the skeleton of the figure, as we can see in Fig.7d after 240 iterations for the solution of equation (9) or in Fig.9 only after 120 iterations for the solution of equation (11). In Fig.9 and Fig.10 we may compare divergence maps for AVF field evaluated by equation (11) and GGVF field respectively.

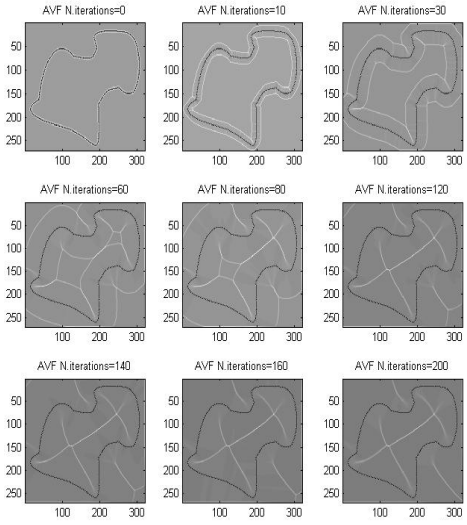


FIGURE 9: Divergence Maps of AVF field from equation (11) for different iteration numbers

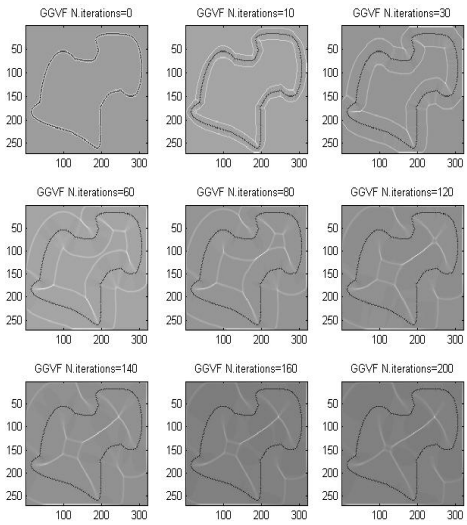


FIGURE 10: Divergence Maps of GGVF for different iteration numbers

As a consequence shape and position of any initial contour also depends on the number of iterations performed in the numerical resolution. We have to take into account boundaries of areas with positive divergence, because within them the vector field is null, so the sections of an evolving curve remain entrapped in their interiors, as we can see in Fig.11a, where an arbitrary initial contour is superimposed to the divergence map of our image.

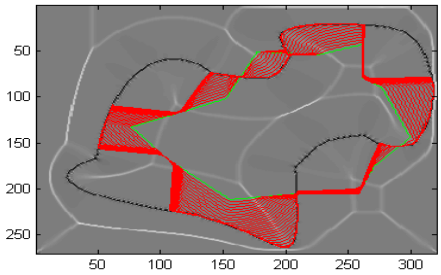


FIGURE 11a: Divergence Map of GGVF field with an arbitrary initial contour after 80 iterations

If we now review the results shown in Fig.5c and Fig.5d realized using AVF field and GGVF field respectively, after 80 iterations and we superimpose the initial polygonal with the corresponding divergence maps, we are able to understand the anomalous shape of the extracted edges of Fig.5d, apparently incomprehensible. Some portions of the initial contour fall within the areas where the GGVF field is null, thereby those traits remain trapped inside them (see Fig.11c). Instead after 120 iterations, the capture range of the GGVF field is become more extended, so the evolving curve converges to right results

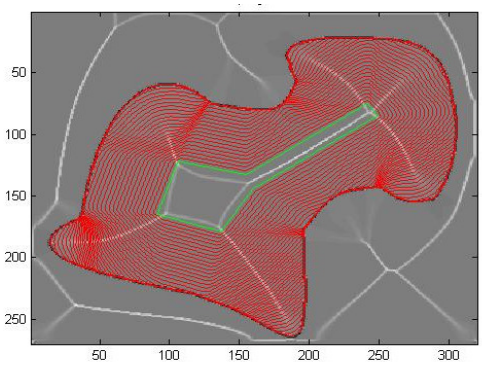


FIGURE 11b: Divergence Map of AVF field field from equation (11) after 80 iterations

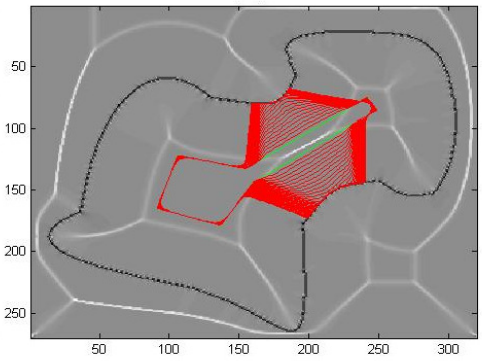


FIGURE 11c: Divergence Map of GGVF from equation (7) after 80 iterations

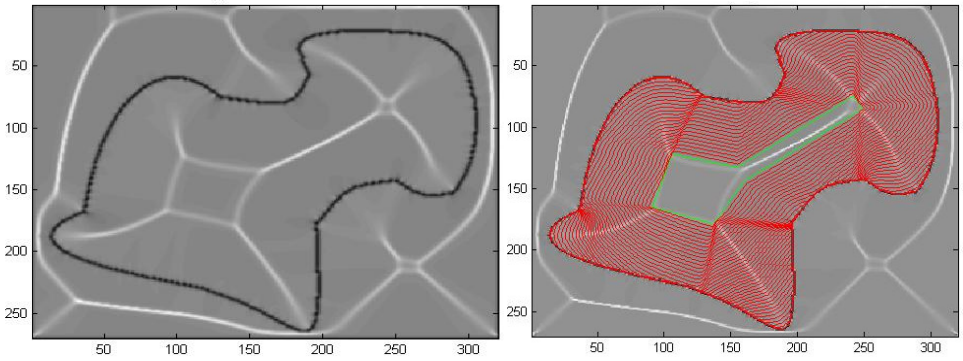


FIGURE 11d: Divergence Map and active contour with GGVF field after 120 iterations

We would remember that divergence will be zero either when the field is null or when the inward and the outward flux are equal. In the divergence map relative to the second test image, composed by two disjoint boundaries, we may well see two closed curves corresponding to positive divergence values. Within the first curve the field is zero, instead inside the second, divergence values are near zero because the flux is null, whereas the vector field is rather uniform and changes its direction converging to the inner circle, as we can see in Fig.12a. Then the traits of an initial contour that fall inside this region are pushed towards the inner circle rather than the outer edge (Fig.12b,112c).

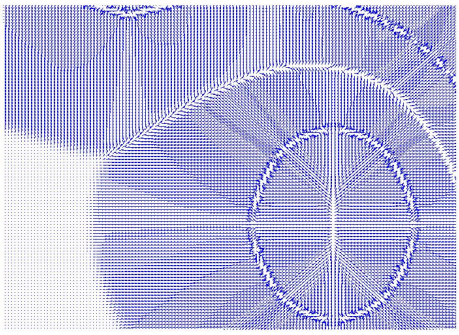


FIGURE 12a: AVF field after 80 iterations

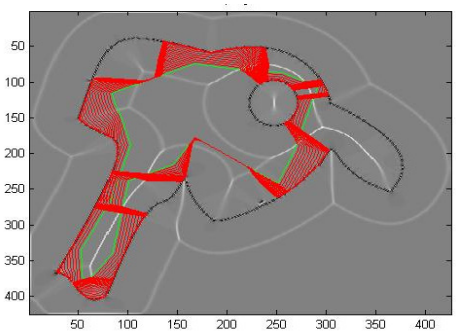


FIGURE 12b: Divergence Map of AVF field with an arbitrary initial contour

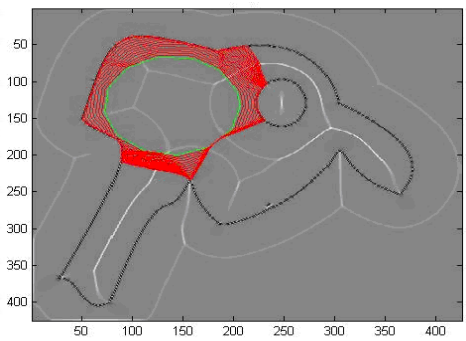
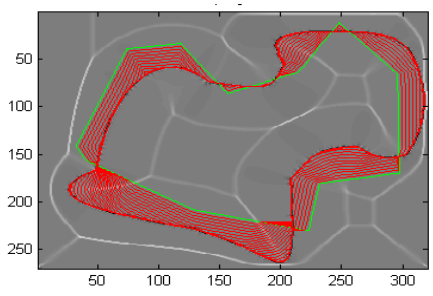


FIGURE 12c: Divergence Map of AVF field with an arbitrary initial contour

Thereby it results much easier to understand the anomalous behaviour in the convergence of each polygonal sections of Fig.11a, Fig.11b, Fig.11c or Fig.12b, Fig.12c using maps of divergence associated with the GGVF or AVF field. The traits of an initial contour are pushed along the field lines that, according to the Divergence Theorem, start at sources, i.e. areas of positive divergence, and end at sinks with negative divergence. In addition we would like to make some observations about the capture range extension. Looking closely at the divergence map, we could note that the area enclosed between the external or internal curve with positive divergence, and the curve of negative divergence, corresponding to the object edge, delimits the capture range. As we may well see in Fig.7a, 7b, 7c, 7d and in Fig.8a, 8b, 8c, 8d, shape and size of this area are varying with the number of iterations chosen. So we may conclude that divergence map, at any given time, is strictly related to the initialization problem and the capture range, because it allow us to clearly distinguish between sources and sinks for our image, enabling us to properly place an initial curve, as shown in Fig.13. Accordingly we suggest to generate a contour map of the field divergence in order to visualize it through level sets, i.e. the set of points in the image domain where divergence is constant. In this way we could automatically extract an appropriate initial contour using level curves corresponding to high divergence values.



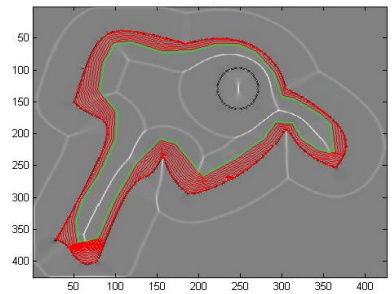


FIGURE13: Divergence Map with initial contours converging towards edges to detect

Given an image *I(x,y)*, it has been carried out $\vec{v}(x, y)$ resolving numerically equation (11). After evaluating its divergence map $I_D : D \subset R^2 \to C$ in the gray colour space C = [0;255], it has been generated a contour map concerning to it (Fig.14, Fig.15) and subsequently it has been selected as initial contour the level curve of intensity 153 for the two test images. In this way the chosen curve certainly encloses source areas from which the field diverges, pushing the active contour towards object edges. As we can see from Fig.16 to Fig.19, edges have been extracted correctly, establishing a quite good effectiveness of the proposed approach.

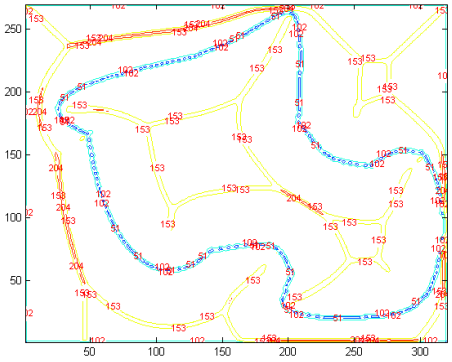


FIGURE 14: Level curves of Divergence Map

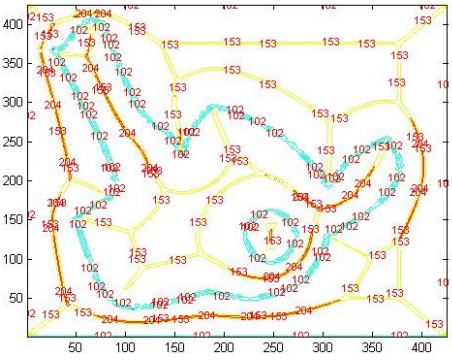


FIGURE15: Level curves of Divergence Map

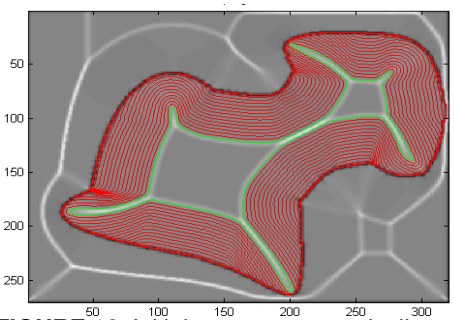


FIGURE.16: Initial contour automatically selected

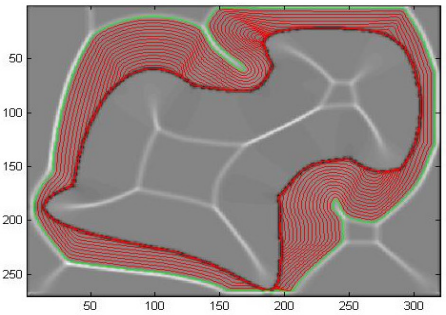
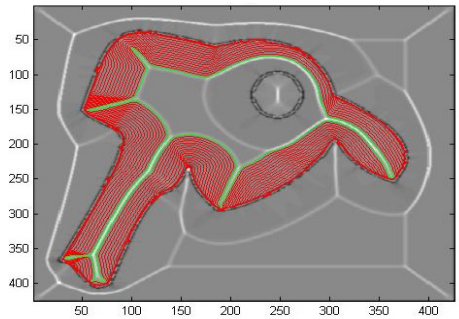


FIGURE.17: Initial contour automatically selected



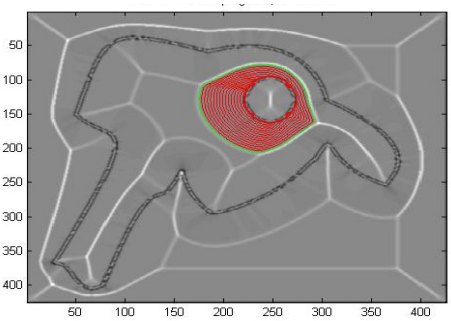


FIGURE. 18: Initial contour automatically selected **FIGURE.19**: Initial contour automatically selected

We have used real images for the validation of the proposed framework and promising experimental results were obtained with Gray Matter images derived from a data set of patients affected by Alzheimer disease (Fig.20) and subjects of a control group. This model doesn't require the user to input manually an initial curve close to the edges, that may be fastidious to do for an application dealing with a large number of images.

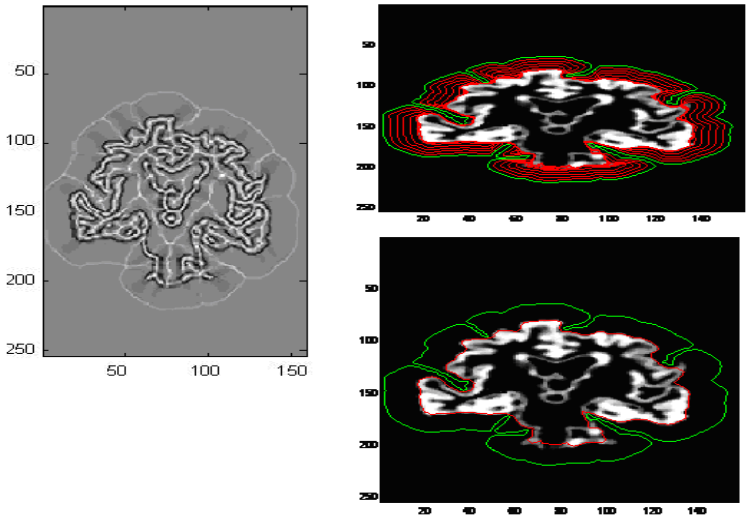


FIGURE 20: Initial contour automatically selected for a GM image after 10 Iterations

4. CONCLUSIONS

In this work, we have analyzed edge extraction by a deformable contour procedure, using an external force field derived from an anisotropic vector flow, with different diffusion equations and initial conditions. We have suggested a new

approach to overcome some limitations of traditional parametric snakes, specially regarding to the initialization problem. The external force field, resolution of an anisotropic diffusion problem, should be considered in relation to its divergence map, because the source areas, from which the vector field diverges pushing the deformable initial contour towards the edges to detect, are well defined and recognizable in it. We have observed that only if source areas are completely included inside the initial contour, the edge detection will turn out well. Moreover we have proposed as automatic initialization procedure the use of level curves of divergence contour map, in such a way it doesn't be required a specification of an initial curve by the user.

ACKNOWLEDGMENT

I sincerely would like to thank Prof. Naldi Giovanni (Department of Mathematics "Federigo Enriques", University of Milan – Italy) for the helpful contribution given for this study.

REFERENCES

[1] M. Kass, A. Witkin, D. Terzopoulos, " Snakes: active contour models" , Int. Jour. Comp. Vision, Vol. I, N.4, 1988, pp 321-331

[2] T. Mc Inerney, D. Terzopoulos, "T-snakes:Topology adaptive snakes" , Medical Image Analysis,4,2000,pp. 73-91.

[3] J. Liang,T.Mc Inerney, D. Terzopoulos, "United Snakes" , Medical Image Analysis,10,2006pp. 215-233

[4] S.J. Osher, J.A. Sethian, "Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations", J. of Computational Physics, 79, 1988, pp 12-49

[5] S.J. Osher, R.P. Fedkiw, "Level set methods: an overview and some recent results", J. of Computational Physics, 169, 2001, pp 463-502

[6] C. Xu, L.J. Prince, "Snakes ,shapes, and gradient vector flow", IEEE Trans. on Image Processing, 7, 1998

[7] C. Xu, A. Yezzi, and J. Prince, "On the Relationship between Parametric and Geometric Active Contours," Proc. Asilomar Conf. Signals, Systems, and Computers, pp. 483-489, 2000.

[8] B. Li, T.A Scott, "Active contour external force using vector field convolution for image segmentation",. IEEE Transactions on Image Processing, 16, 2007, pp.2096-2106

[9] A. Kovács, T. Szirányi, "Improved force field for vector field convolution method",
ICIP IEEE 2011, pp. 2909-2912.

[10] C. Xu, L.J. Prince, “Gradient Vector Flow: a New External Force for Snakes”, IEEE Proc. Conf. on Comp. Vis. Patt. Recogn., CVPR’97

[11] C. Xu, L.J. Prince, “Generalized gradient vector flow external forces for active contours”, Signal Proc., an Intern. Journal, 71, 1998

[12] R. Courant,D. Hilbert, “Methods of Mathematical Physics”, Vol I, Ed. Interscience, New York,1953

[13] J. Cheng, W.F. Say, “Dynamic Directional Gradient Vector Flow for Snakes”, IEEE Trans. on Image Processing, 15, 2006

[14] N. Paragios, O. Mellina-Gottardo, V. Ramesh, “Gradient Vector Flow Fast Geometric Active Contours”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 26, 2004

[15] C. Li, J. Li, M.D. Fox, “Segmentation of Edge Preserving Gradient Vector Flow: an Approach Toward Automatically Initializing and Splitting of Snakes”, IEEE Proc. Conf. on Comp. Vis. Patt. Recogn., CVPR’05

[16] C. Tauber, H. Batiata, A. Ayache, “A Robust Active Initialization and Gradient Vector Flow for Ultrasound Image Segmentation”, MVA2005 IAPR Conference on Machine Vision Applications, 2005

[17] F. Yabin, L. Caixia, Z. Bingsen, P. Zhenkuan, “An Improved Algorithm of Contour Initialization in Active Contour Model”,Image and Graphics, ICIG 2007

[18] P. Perona, J. Malik, “Scale-space and edge detection using anisotropic diffusion” IEEE Trans. on Pattern Analysis and Machine Intelligence, 12, 1990

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. Started with Volume 6, 2012, IJIP will be appearing with more focused issues. Besides normal publications, IJIP intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

LIST OF TOPICS

The realm of International Journal of Image Processing (IJIP) extends, but not limited, to the following:

- Architecture of imaging and vision systems
- Character and handwritten text recognition
- Chemistry of photosensitive materials
- Coding and transmission
- Color imaging
- Data fusion from multiple sensor inputs
- Document image understanding
- Holography
- Image capturing, databases
- Image processing applications
- Image representation, sensing
- Implementation and architectures
- Materials for electro-photography
- New visual services over ATM/packet network
- Object modeling and knowledge acquisition
- Autonomous vehicles
- Chemical and spectral sensitization
- Coating technologies
- Cognitive aspects of image understanding
- Communication of visual data
- Display and printing
- Generation and display
- Image analysis and interpretation
- Image generation, manipulation, permanence
- Image processing: coding analysis and recognition
- Imaging systems and image scanning
- Latent image
- Network architecture for real-time video transport
- Non-impact printing technologies
- Photoconductors

- Photographic emulsions
- Prepress and printing technologies
- Remote image sensing
- Storage and transmission
- Photopolymers
- Protocols for packet video
- Retrieval and multimedia
- Video coding algorithms and technologies for ATM/p

CALL FOR PAPERS

Volume: 6 - **Issue: 5**

i. Submission Deadline : August 31, 2012 **ii. Author Notification:** September 30, 2012

iii. Issue Publication: October 2012

CONTACT INFORMATION

Computer Science Journals Sdn BhD

B-5-8 Plaza Mont Kiara, Mont Kiara

50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6204 5627

Fax: 006 03 6204 5628

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2012
COMPUTER SCIENCE JOURNALS SDN BHD
B-5-8 PLAZA MONT KIARA
MONT KIARA
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6204 5627
FAX: 006 03 6204 5628
EMAIL: cscpress@cscjournals.org