



WHERE WISDOM SHARES

International Journal of Advanced Computer Science and Applications

Special Issue



# Image Processing and Analysis

ISSN 2156-5570(Online)  
ISSN 2158-107X(Print)



[www.ijacsa.thesai.org](http://www.ijacsa.thesai.org)

## **International Journal of Advanced Computer Science and Applications**

### **Special Issue on Image Processing and Analysis**

---

#### **Scope of this Issue:**

The range of topics covered by Special Issue on Image Processing and Analysis includes the following areas related to the generation, processing and communication of visual information:

**Generation and Display** - Imaging sensors and acquisition systems, illumination, sampling and scanning, quantization, colour reproduction, image rendering, display and printing systems, evaluation of image quality.

**Processing and Analysis** - Image enhancement, restoration, segmentation, registration, multispectral, colour and texture processing, multiresolution processing and wavelets, morphological operations, stereoscopic and 3-D processing, motion detection and estimation, video and image sequence processing.

**Implementations and Architectures** - Image and video processing hardware and software, design and construction, architectures and software, neural, adaptive, and fuzzy processing.

**Coding and Transmission** - Image and video compression and coding, compression standards, noise modelling, visual information networks, streamed video.

**Retrieval and Multimedia** - Storage of images and video, database design, image retrieval, video annotation and editing, mixed media incorporating visual information, multimedia systems and applications, image and video watermarking, stenography.

**Applications** - Innovative application of image and video processing technologies to any field, including life sciences, earth sciences, astronomy, document processing and security.

---

## IJACSA Special Issue Guest Editors



**Prof. T. V. Prasad, PhD**  
**Dean (R&D and Industrial Consultancy)**  
**Lingaya's University, India**

*Dr. T. V. Prasad has over 16 years of experience in industry and academics, has deep interest in planning and executing major IT projects, as well as in research in new areas of CS/IT and bioinformatics.*

*He has earned doctoral degree from Jamia Millia Islamia University, New Delhi in the area of computer sciences/bioinformatics. After having worked as Head of the Department of Computer Science & Engineering, he is currently working with Lingaya's University, Faridabad, as Dean of R&D and Industrial*

*Consultancy.*

*He is a trained auditor for implementation of Information Security Management System (ISMS) as per the ISO/IEC 27001:2005 standard. He also possesses quality system (ISO 9000) certification experience as technical expert. He has lectured at various international and national forums on subjects related to computing and is currently supervising 10 research scholars in various areas of computing.*

*His research interests include bioinformatics, consciousness studies, artificial intelligence (natural language processing, swarm intelligence, robotics, knowledge representation and retrieval). He has been Principal Investigator for some funded projects of the Govt. He has over 65 papers in different journals and conferences, and has authored four books and two chapters.*



**Dr. Rongrong Ji, PhD**  
**Research Scientist,**  
**Columbia University, USA**

*Dr. Rongrong Ji has deep interest in Mobile Visual Search and Geographical Multimedia Analysis, Visual Codebook Representation and Indexing, Iterative Video Search and Query Formulization, Social Multimedia Mining and Recommendation, Machine Learning in Vision Representation.*

*He has earned doctoral degree from Computer Science Department, Harbin Institute of Technology at November 2010 in the area of computer sciences/bioinformatics. After having worked with Institute of Digital Media, Peking University, he is currently working as PostDoc Researcher at DVMM Lab, working with Professor Shih-Fu Chang, Department of Electronic Engineering, Columbia University*

*He is a holder of 5 Chinese Patents and 2 International Patents and is the recipient of several international awards including 2010 CVPR Doctoral Consortium Award. He has also published over 40 referred journals and conferences, including Pattern Recognition, CVPR, and ACM Multimedia.*

*From the Desk of Guest Editor...*

I cannot forget the days when we had to work on character user interface (CUI). Thoroughly remembering commands of the operating system and syntax of programming languages were quite cumbersome. If slightest error is made in punctuation, for instance a period (".") is forgotten in COBOL, the compiler would generate at least 10 errors. Gone are the days when textual commands need to be remembered anymore. It is less than two decades ago that computer graphics were heavily employed in motion pictures.

Application of soft computing techniques (artificial neural networks, genetic algorithms and fuzzy logic, or combination thereof) in computer graphics has led to understanding of scenarios and recognition of objects possible. This leads to more "intelligent" or human like computing. Multimedia databases, spatial databases and other forms of data storage, and visual information network, all together are to be linked up with structures such as semantic web. We aren't happy with the existing arrangements and are more inquisitive to know how neural processing takes place for pattern recognition in the brain.

There are many challenging issues handling image data. Handling operations such as, but not limited to resolution, colour schemes, lighting, segmentation, storage, recognition, texture, morphing, retrieval, visualization, analysis, validation, compression, restoration, steganography, noise reduction, annotation, watermarking, motion detection, estimation, sequence processing, etc. can be serious problems or business.

Industrial applications of image processing have no bounds. Speed and accuracy can be enhanced greatly. So also in civil aviation industry, where luggage/baggage can be scanned automatically, which is done manually in current scenario Applications in the area of mining, wood industry, exploration/ geology, power sector (nuclear plants), security, space research, etc. are well known.

Medical field is also equally obsessed with the impact of image processing techniques. MRI scanning has become more accurate, meaningful and intelligent only after image analysis algorithms. We now talk of whole body scanning, wherein the entire human body is analyzed and predictions for the future are made by a group of doctors on the reports generated by computer.

Thanks to this field that has helped the astronomers unravel and understand the structure of universe. We now know where exactly are we placed and which galaxies are in our neighbourhood. We can compute the distances exactly between different galaxies without having visited there.

Areas such as fractal graphics, virtual reality, steganography in audio and video, computer gaming, telemedicine/tele-surgery, etc. are making a mark. Educational sector is yet to take up processing of knowledge and visualization of various disciplines at a large scale. Efforts are going on for medicine, architecture and engineering areas, other disciplines are yet to be considered.

I felt extremely glad to note that it is less than a year's time that the Chief Editor commenced this Journal, and the readership has increased tremendously. I thank the Chief Editor for inviting me to be Guest Editor for this Special Issue on Image Processing and Analysis, which is the first in a series of Special Issues being planned. The acceptance rate for the Special Issue is 24%. The purpose is bring to you good quality work of researchers, scientists and academicians.

My hearty greetings to all authors, reviewers, editors and the Journal staff for such an accomplishment.

---

**Thank You for Sharing Wisdom!**

**Dr. T.V Prasad**  
**Guest Editor**  
**IJACSA Special Issue on Image Processing and Applications**

(iii)

<http://ijacsa.thesai.org/>

## CONTENTS

### Paper 1: Sketch Recognition using Domain Classification

*Authors: Vasudha Vashisht, Tanupriya Choudhury, Dr. T. V. Prasad*

PAGE 1 – 9

### Paper 2: A new optimization based image segmentation method by particle swarm optimization

*Authors: Fahd M. A. Mohsen, Mohiy M. Hadhoud, Khalid Amin*

PAGE 10 – 18

### Paper 3: Image segmentation by adaptive distance based on EM algorithm

*Authors: Mohamed Ali Mahjoub, Karim Kalti*

PAGE 19 – 25

### Paper 4: Modeling of neural image compression using GA and BP a comparative approach

*Authors: G.G Rajput, Vrinda Shivashetty, Manoj Kumar singh*

PAGE 26 – 34

### Paper 5: Clustering and Bayesian network for image of faces classification

*Authors: Khlifia Jayech, Mohamed Ali Mahjoub*

PAGE 35 – 44

### Paper 6: Skew correction for Chinese character using Hough transform

*Authors: Tian Jipeng, G.Hemantha Kumar, H.K. Chethan*

PAGE 45 – 48

### Paper 7: Recombinant Skeleton Using Junction Points in Skeleton Based Images

*Authors: Mrs. Komala Lakshmi, Dr.M.Punithavalli*

PAGE 49 – 53

### Paper 8: ID Numbers Recognition by Local Similarity Voting

*Authors: Shen Lu, Yanyun Qu, Yanyun Cheng, Yi Xie*

PAGE 54 – 62

**Paper 9: Component Localization in Face Alignment**

*Authors: Yanyun Qu, Tianzhu Fang, Yanyun Cheng, Han Liu*

**PAGE 63 – 69**

**Paper 10: Fine Facet Digital Watermark (FFDW) Mining From The Color Image Using Neural Networks**

*Authors: N. Chenthalir Indra, Dr. E. Ramaraj*

**PAGE 70 - 74**

**Paper 11: Automatic Image Registration Using Mexican Hat Wavelet, Invariant Moment, and Radon Transform**

*Authors: Jignesh N Sarvaiya, Dr. Suprava Patnaik*

**PAGE 75 – 84**

**Paper 12: Human Face Detection under Complex Lighting Conditions**

*Authors: Md. Golam Moazzam, Ms. Rubayat Parveen, Md. Al-Amin Bhuiyan*

**PAGE 85 – 90**

**Paper 13: Ear Recognition using Dual Tree Complex Wavelet Transform**

*Authors: Rajesh M Bodade, Sanjay N Talbar*

**PAGE 91 – 97**

**Paper 14: SUCCESeR Simple and Useful Multi Color Concepts for Effective Search and Retrieval**

*Authors: Satishkumar L. Varma, Sanjay N. Talbar*

**PAGE 98 – 104**

**Paper 15: Automatic License Plate Localization Using Intrinsic Rules Saliency**

*Authors: Chirag N. Paunwala, Dr. Suprava Patnaik*

**PAGE 105 – 111**

**Paper 16: Performance Comparison of SVM and K-NN for Oriya Character Recognition**

*Authors: Sanghamitra Mohanty, Himadri Nandini Das Bebartta*

**PAGE 112 – 116**

# Sketch Recognition using Domain Classification

Vasudha Vashisht  
Assistant Professor  
Dept of Computer Sc & Engg  
Lingaya's University  
Faridabad, Haryana, INDIA

Tanupriya Choudhury  
Senior Lecturer  
Dept of Computer Sc & Engg  
Lingaya's University  
Faridabad, Haryana, INDIA

Dr. T. V. Prasad  
Dean (R&D)  
Lingaya's University  
Faridabad, Haryana, INDIA

**Abstract**— Conceptualizing away the sketch processing details in a user interface will enable general users and domain experts to create more complex sketches. There are many domains for which sketch recognition systems are being developed. But, they entail image-processing skill if they are to handle the details of each domain, and also they are lengthy to build. The implemented system's goal is to enable user interface designers and domain experts who may not have proficiency in sketch recognition to be able to construct these sketch systems. This sketch recognition system takes in rough sketches from user drawn with the help of mouse as its input. It then recognizes the sketch using segmentation and domain classification; the properties of the user drawn sketch and segments are searched heuristically in the domains and each figures of each domain, and finally it shows its domain, the figure name and properties. It also draws the sketch smoothly. The work is resulted through extensive research and study of many existing image processing and pattern matching algorithms.

**Keywords**- Sketch recognition; segmentation; domain classification.

## I. INTRODUCTION

As computers become an integral part of our lives, it becomes increasingly important to make working with them easier and more natural. It is visional to make human-computer interaction as easy and as natural as human-human interaction. As part of this vision, it is imperative that computers understand forms of human-human interaction, such as sketching. Computers should be able to understand the information encoded in diagrams drawn by and for scientists and engineers. A mechanical engineer, for example, can use a hand-sketched diagram to depict his design to another engineer. Sketching is a natural modality of human-computer interaction for a variety of tasks [7].

In an attempt to combine the freedom provided by a paper sketch with the powerful editing and processing capabilities of an interpreted diagram, sketch recognition systems have been developed for many domains, including Java GUI creation, UML class diagrams, and mechanical engineering. Sketch interfaces:

- 1) Interact more naturally than traditional mouse-and-palette tools by allowing users to hand-sketch diagrams,
- 2) Can connect to a back-end system (such as a CAD tool) to offer real-time design advice,

- 3) Recognize the shape as a whole to allow for more powerful editing,
- 4) Beautify diagrams, removing mess and clutter, and thereby
- 5) Notify the sketcher that the shapes have been recognized correctly[6].

Previous sketch systems required users to learn a particular stylized way of drawing, and used a feature-based recognition algorithm, such as a Rubine or a GraffitiTM-type algorithm. What these algorithms lose in natural interaction by requiring the sketcher to draw in a particular style, they gain in speed and accuracy. Rather than recognizing shapes, the algorithm recognizes sketched gestures, where each gesture represents a single shape. These sketched gestures focus more on how something was drawn than on how the drawn object looks. These recognition algorithms require that each gesture can be drawn in a single stroke in the same manner (i.e., same underlying stylistic features—stroke direction, speed, etc.) each time. Each gesture has recognized based on a number of features of that stroke, such as the initial angle of the stroke, end angle, speed, number of crosses, etc. Because of these requirements, the gesture representing the shape may look different from the shape itself.

Further it is noted that :

- Human generated descriptions contained syntactic and conceptual errors, and that
- It is more natural for a user to specify a shape by drawing it than by editing text[7].

When working in a closed domain such as this one, the computer knows exactly which conceptual uncertainties remain, and which hypotheses need to be tested and confirmed. The system builds a shape description language, using a modification of the version spaces algorithm that handles interrelated constraints.

To achieve the goals, the system is implemented for the segmentation of the sketch. This system takes the input from the user about the positions where the sketch has drawn and then it process this information to divide the sketch into a number of segments according to the position and the direction of the sketch that has drawn by the user.

To allow for natural drawing in proposed sketch recognition systems, shapes are described and recognized in terms of the subshapes that make up the shape and the geometric relationships (constraints) between the subshapes. Strokes are first broken down into a collection of primitive shapes using a SEGMENTATION technique which operates on the pixels and their orientation. A higher-level shape is then recognized by searching for possible subshapes in the domain and testing that the appropriate geometric constraints hold. The geometric constraints confirm orientation, angles, relative size, and relative location.

The remainder of this paper is organized as follows. Section 2 briefly reviews the basics of The User Interface Development Framework with Domain specific information. Section 3 gives the details of methodology by specifically talking about the language for describing drawing and display, its building blocks and limitations. Section 4 of the paper discusses the overall recognition system, the algorithm used along with its limitations and further research directions.

## II. USER INTERFACE DEVELOPMENT

### A. Domain-specific Information

When constructing a user interface, the domain-specific information is able to be obtained by asking the following questions [7]:

- What are the observable states to be recognized?
- How are these states to be recognized?
- What should happen when these states are recognized?
- How can we modify these states?

In sketch recognition user interfaces, the domain-specific information is obtained by asking these questions:

- What shapes are in the domain?
- How is each shape recognized?
- What should happen after each shape is recognized?[8]

Many domain-specific events can occur after a shape is recognized, but what is common in most domains is a change in display. Sketchers often prefer to have a change in display to confirm that their object was recognized correctly, as a form of positive feedback. Changes in display may also function as a way to remove clutter from the diagram. For example, the system may replace several messy hand-drawn strokes with a small representative image[4]. A change in the display may vary from a simple change in color, a moderate change of cleaning up the drawn strokes (e.g. straightening lines, joining edges), to a more drastic change of replacing the strokes with an entirely different image. Because display changes are so popular and so common to most domains, so they are included in the language.

This framework not only defines which shapes are in the domain and how they are to be recognized in the domain, it also recognizes the importance display in creating an effective user interface. Developers of different domains may want the same shape to be displayed differently: Compare a brainstorming sketch interface that develop a web page layout in which shapes may be left unrecognized, to a UML class

diagram sketch interface, where sketchers may want to replace box-shaped classes with an index card-like image.

### B. The Framework

Rather than build a separate recognition system for each domain, it should be possible to build a single, domain-independent recognition system that can be customized for each domain. In this approach, building a sketch recognition system for a new domain requires only writing a domain description, which describes how shapes are drawn and displayed. This description is then being transformed for use in the domain independent recognition system. The inspiration for such a framework stems from work in speech recognition and compiler-compilers, which have used this approach with some success.

In this framework, the recognition system translates the domain description into a recognizer of hand drawn shapes. This is analogous to work done on compiler-compilers, in particular, visual language compiler-compilers. A visual language compiler-compiler allows a user to specify a grammar for a visual language, then compiles it into a recognizer which can indicate whether an arrangement of icons is syntactically valid. One main difference between that work and this one is that the visual language compiler-compiler deals with the arrangement of completed icons, whereas this work includes three additional levels of reasoning:

- dealing with how strokes form primitive shapes (such as lines and ellipses),
- how these primitive shapes form higher-level shapes or icons, and
- how the higher-level shapes interact to form more complicated shapes or less formal shape groups.

To build a new sketch interface:

- 1) A developer writes a domain description language describing information specific to each domain, including: what shapes are included in the domain, and how each shape is to be recognized and displayed (providing feedback to the user).
- 2) The developer will write a Java file that functions as an interface between the existing back-end knowledge system (e.g., a CAD tool) and the recognition system.
- 3) The User Interface customizable recognition system translates the domain description language into shape recognizers, editors, and exhibitors.
- 4) The UI customizable recognition system now functions as a domain-specific sketch interface that recognizes and displays the shapes in the domain, as specified in the domain description. It also connects via the Java interface (listed in Step 2) to an existing back-end system.

### C. Implementation

The framework is implemented by building:

- 1) a symbolic language to describe domain-specific information, including how shapes are drawn, displayed, and edited in a domain; and



- 2) a customizable, multi-domain recognition system that transforms a domain description language into recognizers, and exhibitors to produce a domain-specific user interface

### III. A PERCEPTUAL LANGUAGE FOR DESCRIBING DRAWING AND DISPLAY IN RECOGNITION

In order to generate a sketch interface for a particular domain, the system needs domain-specific information, indicating what shapes are in the domain and how each shape in the domain is to be recognized and displayed. Domain information should provide a high level of abstraction to reduce the effort and the amount of sketch recognition knowledge that is needed by the developer. The domain information should be accessible, understandable, intuitive, and easy for the developer to specify [7].

A shape description needs to be able to describe a generalized instance of the shape, describing all acceptable variations, so that the recognition system can be properly recognized all allowable variations. A shape description should not include stylized mannerisms (such as the number, order, direction, or speed of the strokes used) [8] that would not be presented in other sketchers' drawings of a shape, as it would require all sketchers to draw in the same stylistic manner as the developer in order for their sketches to be recognized. Thus, it has chosen to describe shapes according to their user-independent visual properties.

A perceptual language for describing shapes is being developed, for use to specify the necessary domain information. The language consists of predefined primitive shapes, constraints, and display methods. Shape descriptions primarily concern shape, but may include information such as stroke order or stroke direction, if that information would prove useful to the recognition process. The specification of editing behavior allows the system to determine when a pen gesture is intended to indicate editing rather than a stroke, and what to do in response. Display information indicates what to display after strokes are recognized.

The difficulty in creating such a language involves ensuring that the language is broad enough to support a wide range of domains, yet narrow enough to remain comprehensible and intuitive in terms of vocabulary. To achieve sufficient broadness, it was used to describe several hundred shapes in a variety of domains. Relevant figure shows a sample of the shapes described. To achieve sufficient narrowness, only perceptually-important constraints are chosen.

The language also has a number of higher-level features that simplify the task of creating a domain description. Shapes can be built hierarchically. Shapes can extend abstract shapes, which describe shared shape properties, making it unnecessary for the application designer to define these properties numerous times. As an example, several shapes may share the same editing properties. Shapes with a variable number of components, such as poly-lines or polygons (which have a variable number of lines), can be described by specifying the minimum and maximum number of components (e.g., lines) allowed. Contextual information from neighboring shapes also can be used to improve recognition by defining shape groups;

for instance, contextual information can distinguish a pin joint from a circular body in mechanical engineering. Shape group information also can be used to perform chain reaction editing, such as having the movement of one shape cause the movement of another.

### IV. THE RECOGNITION SYSTEM

Recognition consists of two stages:

- stroke processing and segmentation
- shape recognition using domain classification

During stroke processing, each stroke is broken down into a collection of primitive shapes, including line, arc, circle, ellipse, curve, point, and spiral.

During shape recognition, the properties of the strokes and shape are searched for heuristically in each domain. If a stroke or shape has multiple interpretations, all interpretations are added to the pool of recognized shapes, but a single interpretation is chosen for display. The system chooses to display the interpretation that is composed of the largest number of primitive shapes or the first found interpretation, in the case of interpretations composed of the same number of primitive shapes.

#### A. Segmentation

Segmentation refers to the process of partitioning a digital image or sketch into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image or sketch segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images or sketches. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s).

Several general-purpose algorithms and techniques have been developed for image segmentation. Since there is no general solution to the image segmentation problem, these techniques often have to be combined with domain knowledge in order to effectively solve an image segmentation problem for a problem domain.

The existing algorithms like the Canny Algorithm have the disadvantage of working only over the frequency values that are present in the input matrix and have a uniform threshold value. The other algorithms works on the images, that are presented as the input, that is they check for the color frequency in the image, and then segment that image on the basis of that. Not to forget as the main disadvantage of the current segmentation algorithms is the complex computations they involve. Whereas the algorithm that is presented in this document works on the direction in which the sketch is proceeding at the run time. This direction is determined by

taking a pixel set (a set of 5 pixels) and processing those pixels to determine the direction. This direction determines the position or the pixel where segmentation needs to be done.

This Segmentation Algorithm is based upon the pixels or rather the pixel set (set of 5 pixels). The user draws a sketch on the console and its pixel values that is the x-coordinate and y-coordinate values (abscissa and ordinate values) are stored in the database. All the further working will be done by taking these pixel values from the database. The whole of the segmentation algorithm is divided into 8 phases. These phases are as mentioned below.

- 1) *Draw Sketch:* The user is given with a console on which he can draw any sketch that he requires to segment. He can draw any shape or figure with various input devices like a mouse or a light pen etc.
- 2) *Record Pixels:* When the user is drawing the sketch the system simultaneously records all the pixel values i.e. the abscissa and ordinate values of the drawn sketch, and stores these values into the database. When the input device is pressed or down/clicked, goes over any pixel in the consol or the canvas, highlights that pixel and the value of that pixel is stored in the database. This whole process has done at the run time, as soon as that specific pixel is highlighted and does not wait for the user to complete the sketch, thus saving processing time.
- 3) *Compare adjacent pixels:* After the value of the highlighted pixel is stored in the database, the system fetches these values and compares all the adjacent pixel values. This comparison is done so as to determine the flow or the direction of the sketch that is the direction in which the sketch is moving.

In order to get the direction there are 8 different categories. It means that the direction of two adjacent pixels can be one of the 8 categories. These cases are described below. Let us suppose the 2 pixels are P1 and P2, and there coordinate values are  $x_1, y_1$  and  $x_2, y_2$  respectively.

TABLE 1: Segmentation algorithm's 8 cases

Case	Sketch is moving	x	y
1	to the positive X direction keeping Y value as constant	$x_2 - x_1 > 0$	$y_2 - y_1 = 0$
2	in the negative X direction keeping Y value as constant	$x_2 - x_1 < 0$	$y_2 - y_1 = 0$
3	in positive Y direction keeping the X value as constant	$x_2 - x_1 = 0$	$y_2 - y_1 > 0$
4	in negative Y direction keeping X value as constant	$x_2 - x_1 = 0$	$y_2 - y_1 < 0$
5	in positive X direction and negative Y direction	$x_2 - x_1 > 0$	$y_2 - y_1 < 0$
6	in positive X direction and positive Y direction	$x_2 - x_1 > 0$	$y_2 - y_1 > 0$

Case	Sketch is moving	x	y
7	moving in negative X direction and positive Y direction	$x_2 - x_1 < 0$	$y_2 - y_1 > 0$
8	moving in negative X direction and negative Y direction	$x_2 - x_1 < 0$	$y_2 - y_1 < 0$

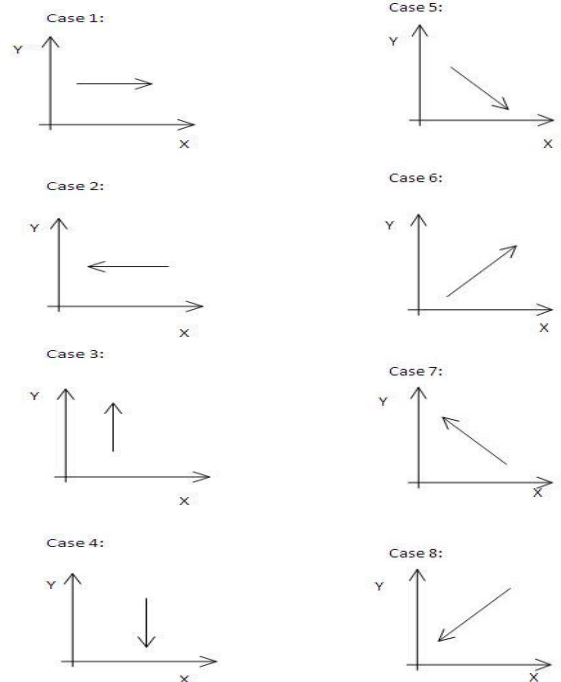


Figure 1: The various directions in which the sketch can flow and these are divided into 8 cases

### B. Domain Classification

The Sketch is recognized using Domain Classification by the method shown here. The properties of the user drawn sketch and segments are searched heuristically in the domains and each figures of each domain. The properties of the figure in the domain and the user sketch are mapped, and finally the sketch is recognized.

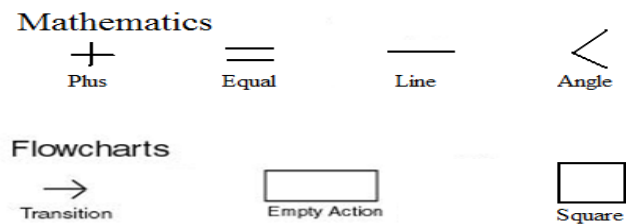


Figure 2: Various domains used by the system

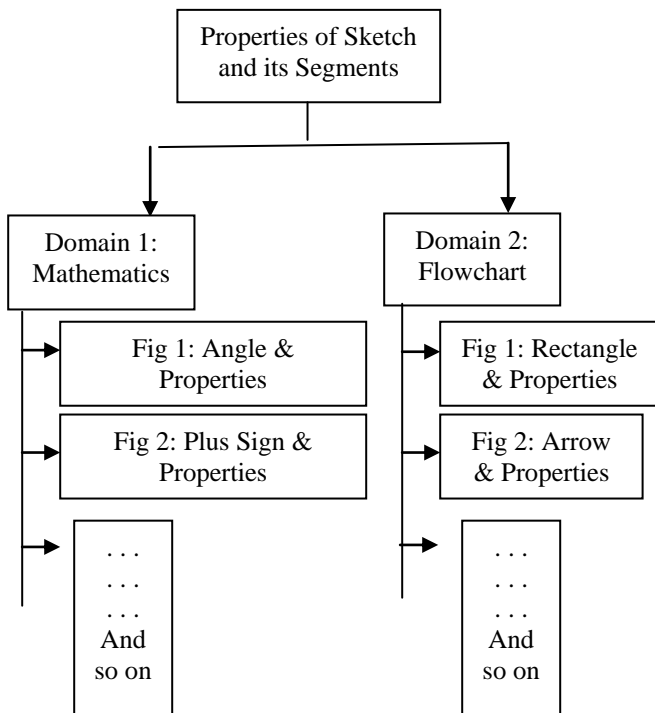


Figure 3: Domain Classification

### C. Experimental Results

If the user draws numerous strokes rapidly, the system can slow down because there is a steady amount of time necessary to preprocess every stroke. The running time of the recognition system is analyzed and determined that, with many unrecognized shapes on the amount of time in the number of shapes on the screen to compute the property values and a logarithmic amount time to insert in to the appropriate data structure. A very small portion of time was used to do the actual recognition, even though the last portion is exponential in the number of strokes on the screen. As a result of indexing, the recognition portion takes a small amount of time, with little to no constraint calculation, as the system was only performing list comparisons. As a result, the system still reacts in what can be considered close to real-time, even with 186 shapes on the screen.

### D. Flowchart

#### 1) First Module: Taking the Input

This module has the designing and coding of the user interface. The user interface is a panel divided into frames and Java AWT elements such as buttons, textboxes, divider etc. This interface takes input drawn by mouse by the user which is a rough sketch. The input is recorded and stored in the database as pixel values with the x coordinate and y coordinate values of the highlighted pixel.

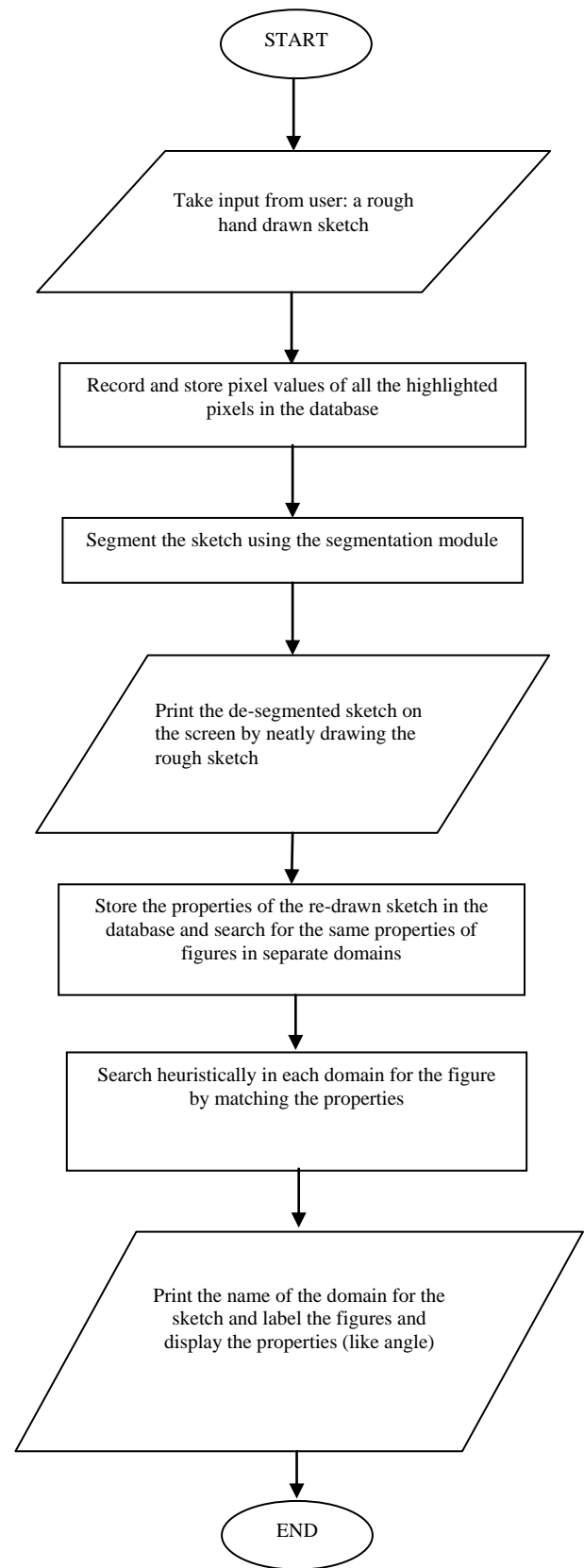


Figure 4: Flowchart for sketch recognition

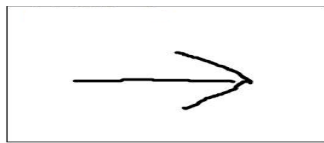


Figure 5: Hand drawn input

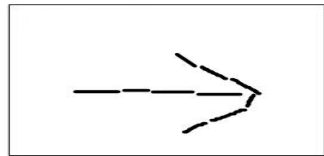


Figure 6: Processed segmented sketch

### 2) Second Module: Recording the Input

The storage of abscissa and ordinate value of the highlighted pixels starts immediately when the user clicks and drags the mouse on the interface to draw the sketch. Thus, the processing starts as user starts to draw and does not wait for the user to complete the sketch; saving the waiting time for the user.

### 3) Third Module: Segmenting the Sketch

This was one of the modules which were tough to design as the previous work done on sketch recognition does not disclose much on the segmentation algorithm.

This module also includes storing the properties of the sketch drawn and segmented so that these can be used further in recognizing the sketch.

### 4) Fourth Module: Searching the sketch in the domains and the figures

After segmentation is completed and the segmentation points are stored, the sketch has to be recognized. Domains are defined, such as mathematics, flowcharts, etc. These domains include properties of figures which lie in those respective domains.

### 5) Sn

### 6) Fifth Module: Re-Drawing the Recognized figure

After the sketch is recognized the de-segmented figure is re-drawn neatly with straight lines and proper curves. These replace the roughly drawn sketches in the user interface frame. This is done by looking at the properties of the figure such as distances and angles.

This module also includes displaying the domain name and the figure name if recognized. It may also display properties of the figure such as the angle measurements if the recognized figure is an angle. This module comes only if the sketch is recognized, if it is not recognized both domain and figure are undefined and the sketch is not re-drawn again.

Screen Shots:

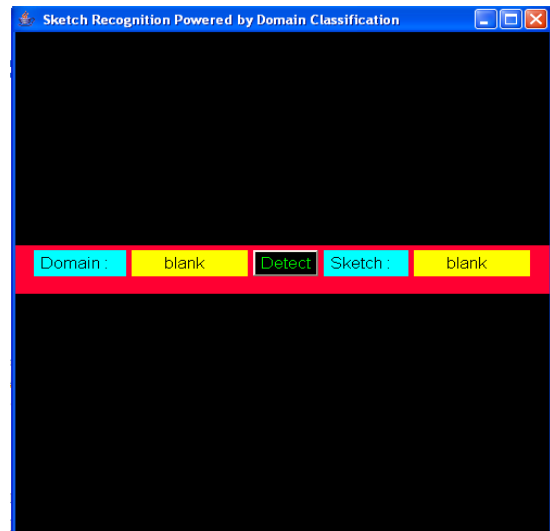


Figure 7: Basic User Interface

The Basic User Interface consists of 3 parts:

- input display
- output display
- information display

In the input display part, the user draws the sketch. In the output display the recognized sketch is displayed. Whereas in the information display shows the information like in which domain the sketch lies and what is the name of that sketch.

Following figure represents a sketch that is drawn by the user. When the user press the detect button, the system does its processing and generates an output.

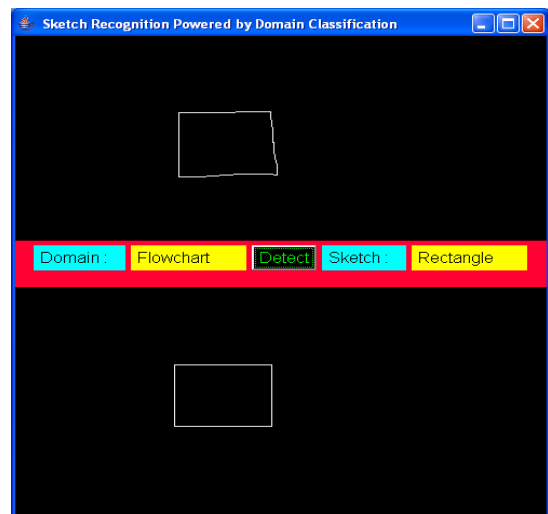


Figure 8: User Interface with input sketch and recognized output sketch as a Rectangle

This figure represents an input as submitted by the user in the previous figure. Along with that there is the recognized figure with proper orientation. Also there is the domain i.e. "Flowchart" in which this sketch lies and the name of this sketch is "Rectangle".

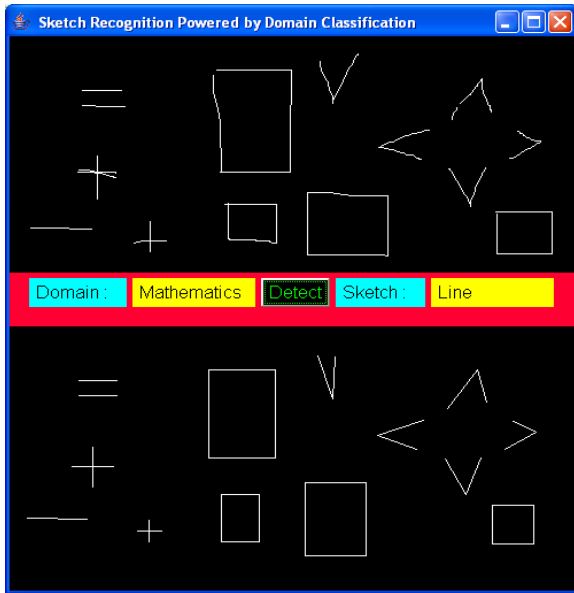


Figure 9: User Interface with input sketches and recognized output sketches in a single window

Figure 9 is a special figure. In this figure, it is represented that the user can draw any number of sketches in the input space and when he presses the Detect button, all of the input sketches are detected. Also for each input sketch a correct recognized sketch is displayed in the output space.

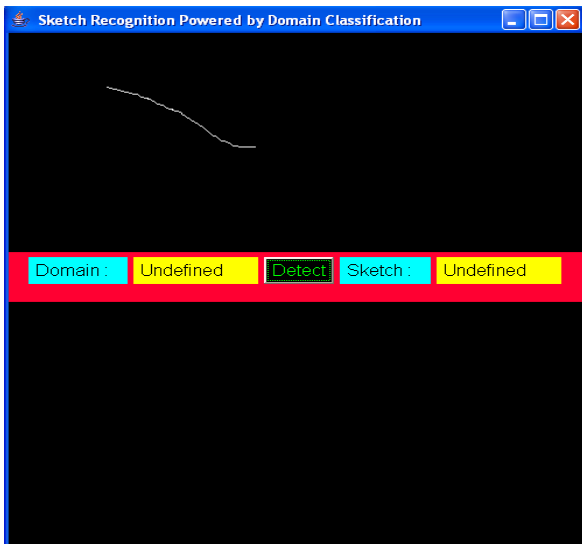


Figure 10: User Interface with input sketch only because this sketch is not recognized by the system.

The figure 10 represents a sketch that is drawn by the user. When the user press the detect button, the system does its processing and generates an output. In this figure no sketch is

recognized because the sketch which the user has drawn doesn't present in any of the domains. Thus the system returns a blank output space and domain and sketch value as "Undefined".

ID	X	Y
1	102	30
2	102	34
3	102	41
4	100	59
5	99	77
6	99	80
7	99	83
8	99	85
9	99	86
10	99	87
11	99	89
12	99	92
13	99	94
14	99	96
15	99	100
16	99	103
17	99	105
18	99	108
19	99	110
20	99	112
21	99	114

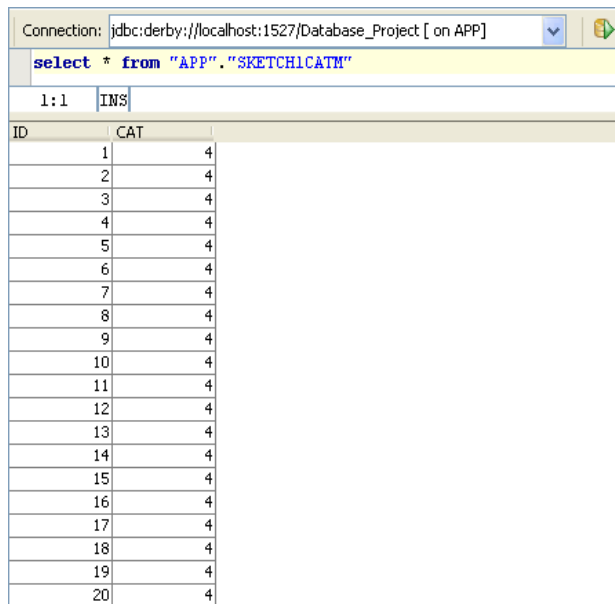
Figure 11: it shows the database table Sketch1

The figure 11 shows the values that are contained by the dynamically created table Sketch1. This table consists of the field ID, X and Y co-ordinates.

ID	CAT
1	4
2	4
3	6
4	6
5	4
6	4
7	4
8	4
9	4
10	4
11	4
12	4
13	4
14	4
15	4
16	4
17	4
18	4
19	4
20	4

Figure 12: it shows the database table Sketch1CAT

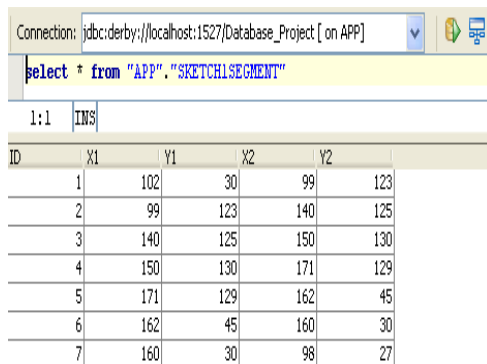
The figure 12 shows the values that are contained by the dynamically created table Sketch1CAT. This table contains the fields ID and the Cat which stores the category value.



ID	CAT
1	4
2	4
3	4
4	4
5	4
6	4
7	4
8	4
9	4
10	4
11	4
12	4
13	4
14	4
15	4
16	4
17	4
18	4
19	4
20	4

Figure 13: it shows the database table Sketch1CATM

Figure 13 shows the values that are contained by the dynamically created table Sketch1CATM. This table overwrites the field of Category in a pixelset.



ID	X1	Y1	X2	Y2
1	102	30	99	123
2	99	123	140	125
3	140	125	150	130
4	150	130	171	129
5	171	129	162	45
6	162	45	160	30
7	160	30	98	27

Figure 14: it shows the database table Sketch1CATM

Figure 14 shows the values that are contained by the dynamically created table Sketch1Segment. This table consists of all the values of the segments i.e. the segment starting and ending point along with the segment number.

## V. MERITS AND DEMERITS

### A. Merits

- This system can be used for multiple sketches at the same time.
- Segmentation Approach used removes the need of use any filtration algorithm, making it faster.
- Has an easy to use user interface.
- This interface can be connected to other applications which need recognition modules.
- It is platform independent.

### B. Demerits

- Currently the system cannot be used for figures with curves, however there is an algorithm designed by us which can be implemented further.

## VI. FUTURE WORK

It can be used to develop:

- Handwriting Recognition Systems.
- Fast Flowchart Designers.
- Other Architecture Designers.
- Higher accessibility tools for users to use the Interface more efficiently. (for OS etc.)
- Same approach can be used to implement similar projects for touch screen applications on various new upcoming OS(s) like nokia rim, apply iphone, windows mobile android, google garnet, palm web bada os , maemo os, meego os etc.

## VII CONCLUSION

The over-arching goal of this work is to make human-computer interaction as natural as human-human interaction. Part of this vision is to have computers understand a variety of forms of interaction that are commonly used between people, such as sketching. Computers should, for instance, be able to recognize the information encoded in diagrams drawn by humans, including mechanical engineering diagrams.

Ordinary systems offer one the freedom to sketch naturally, but it does not provide the benefits of a computer-interpreted diagram, such as more powerful design advice or simulation abilities. Sketch recognition systems bridge that gap by allowing users to hand-sketch their diagrams, while recognizing and interpreting these diagrams to provide the power of a computer-understood diagram. Many sketch systems have been built for a particular domain. Unfortunately, these sketch systems may not fill the needs of the sketcher, and building these sketch systems requires not only a great deal of time and effort, but also an expertise in sketch recognition at a signal level [7]. Thus, the barrier to building a sketch system is high. This researcher wants to empower user interface developers, including designers and educators, who are not experts in sketch recognition, to be able to build sketch recognition user interfaces for use in designing, brainstorming, and teaching.

In response to this need, the framework is developed for facilitating User Interface development. As part of the framework, this project has developed a perception-based sketching language for describing shapes, and a customizable recognition system that automatically generates a sketch recognition system from these shapes. In order to allow drawing freedom, shapes are recognized by what they look like, rather than by how they are drawn. This language provides the ability to describe how shapes in a domain are drawn and displayed within the user interface.

Because humans are naturally skilled at recognizing shapes, the system uses human perceptual rules as a guide for the constraints in the language and for recognition.

The sketch recognition system's accuracy can be improved by

- Combining user-dependent (feature-based) recognition with user-independent (geometric) recognition techniques
- Incorporating global and local context into the recognition system, including geometric, perceptual, functional, multi-modal, similarity, and common sense context.

#### REFERENCES

- [1] Christine Alvarado. Sketch Recognition User Interfaces: Guidelines for Design and Development. In *Making Pen-Based Interaction Intelligent and Natural*, pp.8-14. Menlo Park, California, October 21-24 2004.
- [2] Hammond. T.M., and Davis. R. (2005), "LADDER, a sketching language for user interface developers", *Computers & Graphics* 29.
- [3] Hammond, T.M., and Davis, R. (2002). Tahuti:a geometrical sketch recognition system for uml class diagrams. *AAAI Spring Symposium on Sketch Understanding*
- [4] Jacob Eisenstein. Book Review: Gesture in Human-Computer Interaction and Simulation. *Gesture* 7(1). 2007.
- [5] Landay, J. A., and Myers, B. A. 2001. Sketching interfaces: Toward more human interface design. *IEEE Computer*.
- [6] Liao. S.Z., Wang X.J., and Liang. J. "An incremental approach to sketch recognition". Proceedings of the 4<sup>th</sup> International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.
- [7] Hammond. T.M., LADDER: A Perceptually-based Language to Simplify Sketch Recognition User Interface Development. Ph.D Thesis for Massachusetts Institute of Technology. Cambridge, MA, January 2007.
- [8] Hammond. T.M., and Davis. R.. LADDER: A Sketch Recognition Language. In *MIT Computer Science and Artificial Intelligence Laboratory Annual Research Abstract*. September 2004.

- [9] Bharat Aggarwal, Vishi Segal, Disha Jaggi "Sketch Recognition", unpublished.
- [10] Hammond, Brandon Paulson, " Towards a framework for truly natural Low level Sketch Recognition"

#### AUTHORS' PROFILE

**Vasudha Vashisht** received her bachelor's and master's degree in Computer Science from M.D. University, Haryana, India. She has 6 years of experience in teaching. Currently, she is working as a Assistant Professor in the Dept. of Computer Sc. & Engg. at Lingaya's University, Faridabad, Haryana, India. She has authored 10 papers and her areas of interests include artificial intelligence, Cognitive Science, Brain Computer Interface, Image & Signal Processing. Currently she is pursuing her doctoral degree in Computer Science & Engg. She is a member of reputed bodies like IEEE, International Association of Engineers, International Neural Network Society, etc.

**Tanupriya Choudhury** received his bachelor's degree in CSE from West Bengal University of Technology, Kolkata, India, and master's Degree in CSE from Dr. M.G.R University, Chennai, India. He has one year experience in teaching. Currently he is working as a Senior Lecturer in dept. of CSE at Lingaya's University, Faridabad, India. His areas of interests include Cloud Computing, Network Security, Data mining and Warehousing, Image processing etc.

**Dr. T. V. Prasad** received his master's degree in Computer Science from Nagarjuna University, AP India and a doctoral degree from Jamia Milia Islamia University, New Delhi, India. With over 16 years of academic and Professional experience, he has a deep interest in planning and executing major IT projects, with deep interest in research in CS/IT and bioinformatics. He is the author of 60+ journal/conference/book chapter/white paper publications. He has also held respectable positions such as Deputy Director with Bureau of Indian Standards, New Delhi. His areas of interest include bioinformatics, artificial intelligence, consciousness studies, computer organization and architecture. He is a member of reputed bodies like ISRS, CSI, APBioNet, etc.

# A new Optimization-Based Image Segmentation method By Particle Swarm Optimization

Fahd M. A. Mohsen

Computer & Mathematics Dept.  
Faculty of Science, Ibb University  
Ibb, Yemen

Mohiy M. Hadhoud

Faculty of Computers & Information  
Minufiya University  
Shebin Elkom, Egypt

Khalid Amin

Faculty of Computers & Information  
Minufiya University  
Shebin Elkom, Egypt

**Abstract**— This paper proposes a new multilevel thresholding method segmenting images based on particle swarm optimization (PSO). In the proposed method, the thresholding problem is treated as an optimization problem, and solved by using the principle of PSO. The algorithm of PSO is used to find the best values of thresholds that can give us an appropriate partition for a target image according to a fitness function. In this paper, a new quantitative evaluation function is proposed based on the information theory. The new evaluation function is used as an objective function for the algorithm of PSO in the proposed method. Because quantitative evaluation functions deal with segmented images as a set of regions, the target image is divided into a set of regions and not to a set of classes during the different stages of our method (where a region is a group of connected pixels having the same range of gray levels). The proposed method has been tested on different images, and the experimental results demonstrate its effectiveness.

**Keywords**- *Thresholding-based segmentation; Particle swarm optimization; Quantitative image segmentation evaluation.*

## I. INTRODUCTION

Recently, swarm intelligence (SI) has been applied in numerous fields including optimization [1]. One of SI methods performing well in solving optimization problems is particle swarm optimization (PSO). PSO is a stochastic search method that was developed in 1995 [1] based on the sociological behavior of bird flocking. The algorithm of PSO is easy to implement and has been successfully applied to solve a wide range of optimization problems in many fields such as image processing fields including image segmentation. Image segmentation is a low-level image processing task aiming at partitioning an image into homogeneous regions [2]. The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image. All of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture [3]. Image segmentation methods have been classified into numerous categories of which region and thresholding based segmentations.

Thresholding based segmentation is definitely one of the most popular and effective approaches used in image segmentation [5]. Over the years a wide range of thresholding techniques has been developed and considerable research continues nowadays. Marcello et al. in [6] has classified the

thresholding techniques into two groups, local and global thresholding techniques, and the global ones again classified according to the information they exploit into histogram shape-based methods, clustering-based methods, entropy-based methods, object attribute-based methods and spatial methods. All the thresholding techniques involve a bi-level thresholding and a multilevel thresholding. The main objective of thresholding is to determine a threshold for bi-level thresholding or several thresholds for multilevel thresholding giving a suitable classification for pixels in an image. The simplest problem will be a bi-level thresholding one, where only one threshold, which separates the pixels into only two classes, is selected and the image able to be segmented by thresholding it at this value. This can facilitate to generate a binary image where all pixels having gray levels higher than the threshold are assigned to one class and pixels having gray levels lower than the threshold are assigned to another class. However the problem gets more and more complex when we try to achieve segmentation with greater detail by multilevel thresholding. Then the image segmentation problem becomes a multi-class classification problem where, based on the determined thresholds, pixels having gray levels within a specified range are grouped into one class. Determination of appropriate threshold values, that can segment the image efficiently, is the most important task involved in thresholding techniques. Over the years many method has been developed to solve this problem [5, 7]. The determination of appropriate thresholds is still the most difficult task in the thresholding techniques and it is still a challenge and a hot research topic for the researchers.

Now, the PSO technique has been used to solve the problem of thresholding based segmentation. Zahara et al. in [8] combined a hybrid Nelder–Mead simplex search method and the PSO technique to solve the objective functions of Gaussian curve fitting and the Otsu's method. This combination is applied to image thresholding with multi-modal histograms. In [9] a multilevel threshold selection based on PSO was proposed. The PSO technique was used to find near-optimal thresholds by minimizing the cross entropy between the original image and its thresholded version. Maitra et al. [10] proposed a new thresholding algorithm for histogram-based image segmentation using a hybrid cooperative-comprehensive learning based on the PSO algorithm. In their algorithm, the entropy criterion has been used as a fitness function and near



optimal threshold values have been searched out through maximizing the normal entropy function. Zhiwei et al. in [11] presented a new method to select image threshold automatically based on PSO. They employed the PSO technique to deal with the criteria of Otsu's for Bi-level thresholding image segmentation and Wei et al. in [12] combined the PSO technique with Otsu's for multilevel thresholding image segmentation. Sathya et al. [13] presented a new histogram-based method for multilevel thresholding segmentation based on PSO. This method used Tsallis entropy as a fitness function. The PSO algorithm is used to find the near optimal threshold values that maximize the Tsallis objective function. Hongmei et al. in [14] proposed a multilevel thresholding method segmenting images based on the maximum entropy and an improved PSO. First, the parameters and the evolutionary process of the basic PSO have been improved, and then the combinations of near optimal thresholds are searched out by combining the improved PSO with maximum entropy.

The goal of this paper is to propose a new multilevel thresholding image segmentation method based on the idea of PSO. The PSO algorithm will try to find the near-optimal threshold values that can give us a near optimal segmentation for a target image according to a fitness function. A new quantitative evaluation function will be proposed and used as a fitness function for the algorithm of PSO. The new function manipulates with a segmented image as a set of regions not a set of classes. So, in the proposed method, a given image is segmented into a set of regions, where a region is a group of connected pixels having a specific range of gray levels.

This paper is organized as follows. Section 2 gives an overview of the PSO method. Section 3.A gives an overview of image segmentation evaluation and section 3.B proposed a new quantitative evaluation function. Section 4 presents the proposed method. In Section 5, the experimental results are presented, and, finally, the conclusions are stated in Section 6.

## II. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is a population-based optimization algorithm modeled after the simulation of social behavior of birds in a flock [6, 15]. The algorithm of PSO is initialized with a group of random particles and then searches for optima by updating generations. Each particle is flown through the search space having its position adjusted based on its distance from its own personal best position and the distance from the best particle of the swarm. The performance of each particle, i.e. how close the particle is from the global optimum, is measured using a fitness function which depends on the optimization problem.

Each particle,  $i$ , flies through an  $n$ -dimensional search space,  $R^n$ , and maintains the following information:

- $x_i$ , the current position of  $i^{\text{th}}$  particle ( $x$  - vector),
- $p_i$ , the personal best position of  $i^{\text{th}}$  particle ( $p$  - vector), and
- $v_i$ , the current velocity of  $i^{\text{th}}$  particle ( $v$  - vector).

The personal best position associated with a particle,  $i$ , is the best position that the particle has visited so far. If  $f$  denotes the fitness function, then the personal best of  $i$  at a time step  $t$  is

updated as:

$$p_i(t+1) = \begin{cases} p_i(t) & \text{if } f(x_i(t+1)) \geq f(p_i(t)) \\ x_i(t+1) & \text{if } f(x_i(t+1)) < f(p_i(t)) \end{cases} \quad (1)$$

If the position of the global best particle is denoted by  $gbest$ , then :

$$gbest \in \{ p_1(t), p_2(t), \dots, p_m(t) \} \\ = \min \{ f(p_1(t)), f(p_2(t)), \dots, f(p_m(t)) \} \quad (2)$$

The velocity updates are calculated as a linear combination of position and velocity vectors. Thus, the velocity of particle  $i$  is updated using equation (3) and the position of particle  $i$  is updated using equation (4).

$$v_i(t+1) = w v_i(t) + c_1 r_1(p_i(t) - x_i(t)) + c_2 r_2(gbest - x_i(t)) \quad (3)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (4)$$

In the formula,  $w$  is the inertia weight [16],  $c_1$  and  $c_2$  are the acceleration constants,  $r_1$  and  $r_2$  are random numbers in the range  $[0,1]$  and  $V_i$  must be in the range  $[-V_{\max}, V_{\max}]$ , where  $V_{\max}$  is the maximum velocity.

## III. IMAGE SEGMENTATION EVALUATION

### A. Related Work

In order to measure the performance of image segmentation methods without human interactions, we need an evaluation criterion. There are many image segmentation evaluation functions have been presented in the literature. Those functions have been divided into numerous types including quantitative evaluation measures [17, 18, 19]. Borsotti et al. in [18] empirically proposed a quantitative evaluation function,  $Q$ , such as:

$$Q(I) = \frac{1}{1000 \times Na} \sqrt{n} \sum_{j=1}^n \left[ \frac{e_j^2}{1 + \log L(R_j)} + \left( \frac{M(L(R_j))}{L(R_j)} \right)^2 \right] \quad (5)$$

Where  $I$  is a given image,  $Na$  is the number of pixels in  $I$ ,  $n$  is the number of regions,  $R_j$  denotes to  $j^{\text{th}}$  region,  $L(R_j)$  is the number of pixels in  $j^{\text{th}}$  region and  $e_j$  is the color error of  $j^{\text{th}}$  region. The first term of Equation 5 is a normalization factor and the second term penalizes results with too many regions. The last term in  $Q$  function penalizes simultaneously regions with big color error and small regions.

Zhang et al. [19] proposed an information theoretic approach for segmentation evaluation,  $E$  function, based on entropy theory. Given a segmented image, they define  $V_j$  as the set of all possible values for the luminance in the region  $j$  and let  $L_m(R_j)$  denote the number of pixels in the region  $R_j$  that have luminance of  $m$ . The entropy for the  $j^{\text{th}}$  region is defined such as:

$$H_v(R_j) = - \sum_{m \in V_j} \frac{L_m(R_j)}{L(R_j)} \log \frac{L_m(R_j)}{L(R_j)} \quad (6)$$

Next, they define the expected region entropy of the segmented image,  $H_r(I)$ , such as:

$$H_r(I) = - \sum_{j=1}^n \left( \frac{L(R_j)}{Na} \right) H_v(R_j) \quad (7)$$

As it is seen,  $H_r(I)$  is simply the expected entropy across all regions where each regions has weight (or probability) proportional to its area. The expected region entropy serves in a similar capacity to the term involving the squared color error used in  $Q$  function. The expected region entropy must be combined with another term or factor that penalizes segmentations having a large number of regions since there would otherwise be a strong bias to over-segment the image. Instead of penalizing the expected region entropy, they instead introduced the layout entropy,  $H_l(I)$ , such as:

$$H_l(I) = - \sum_{j=1}^n \frac{L(R_j)}{Na} \log \frac{L(R_j)}{Na} \quad (8)$$

Finally, they define their evaluation function,  $E$ , where a lower entropy value means a better segmentation, as follows:

$$E = H_l(I) + H_v(I) \quad (9)$$

For more study about the image segmentation evaluation methods, the reader is advised to refer to Zhang [20] and Zhang et al. [21]. They have explained a lot of evaluation methods with detailed information.

### B. A new Quantitative Evaluation Function

This section presents a new quantitative evaluation function,  $Q^\backslash$ , based on the information theory. The new function is equivalent to  $Q$  function. The main problem of  $Q$  function is it was designed empirically. This makes  $Q$  function produces less balancing between the homogeneities and number of regions in a segmented image. Figure 1 illustrates this problem. Figures 1(a, b, c, d) show the segmentation results produced through multithresholding the image of House using  $k=1, k=10, k=15$  and  $k=30$  respectively (where  $k$  represents the number of thresholds – Note that the gray levels were divided into equal interval according to  $k$ ). Figure 1(a) shows a segmented image with many merged regions. Figure 1(b) shows a segmented image with fewer details. Figure 1(c) shows a segmented image with more details reflecting more homogenous regions. Figure 1(d) shows the segmentation with many noises. Each figure has its evaluation values computed using  $Q^\backslash$ ,  $Q$  and  $E$  functions respectively. From those results, we can see that the differences between the evaluations of  $Q$  function are very large especially between the evaluations of Figure 1(b) and Figure 1(d). These differences don't reflect the visual judgment of the segmentation quality. This is due to the fact that  $Q$  function produces less balancing between the number and homogeneities of regions where it was designed empirically. The results shown in Figure 1 illustrate that the evaluations obtained by  $Q^\backslash$  function, which will be described

below, are the same as that obtained by  $Q$ , but the order and scores given by  $Q^\backslash$  reflect the visual evaluation of segmentation quality. This is due to  $Q^\backslash$  function produces more balancing between the number of regions and their homogeneities. To show that, see the difference between the evaluations of  $Q^\backslash$  function shown in Figure 1(b) and Figure 1(d). The balancing of  $Q^\backslash$  isn't the optimal but it is acceptable so far.

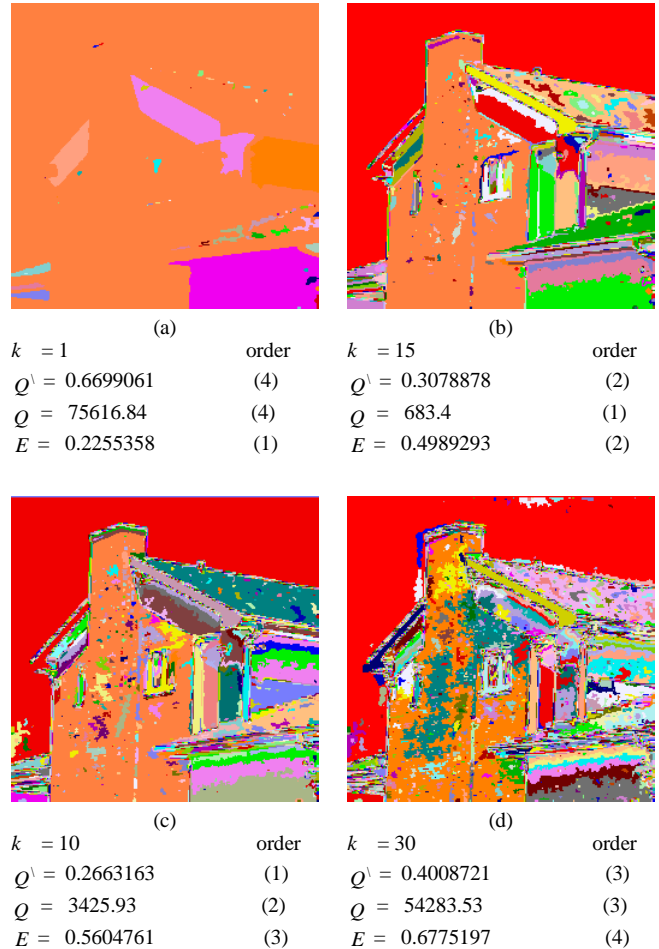


Figure 1: The segmentation results of House image though multithresholding it into regions using:

(a).  $k=1$ . (b).  $k=10$ . (c).  $k=28$ . (d).  $k=30$ .

The orders of segmented images produced by  $Q$  &  $Q^\backslash$  functions are both acceptable but the order of  $Q^\backslash$  function is the best. The order of the segmented images according to  $E$  function is  $k=1, k=10, k=15, k=30$ . This order gives Figure 1(a) the best segmentation. This is due to the fact that  $E$  function performs well in evaluating segmented images having explicit objects (real-world objects) [21] – Note:  $Q^\backslash$ ,  $Q$  and  $E$  functions are computed according to the gray levels and the values of  $Q^\backslash$  and  $E$  are scaled to be in the range  $[0,10]$ .

In describing the  $Q^\backslash$  evaluation function, considered here, we will use the following notations. Let  $I$  be a given image and let  $Na$  be the number of pixels in the given image. We will use

$R_j$  to denote  $j^{\text{th}}$  region, and use  $L(R_j)$  to denote the area of  $j^{\text{th}}$  region (as measured by the number of pixels). The homogeneity of the region  $R_j$  is computed according to the color errors. Let  $e(px_{i,j})$  be the color error of  $i^{\text{th}}$  pixel in  $j^{\text{th}}$  region, we define  $e(px_{i,j})$  according to the mean of a region as:

$$e(px_{i,j}) = \text{mean}_j - g(px_{i,j}) \quad (10)$$

Where  $g(px_{i,j})$  is the gray level of  $px_{i,j}$  and  $\text{mean}_j$  is the mean of  $j^{\text{th}}$  region computed using the gray levels of pixels being in this region. So, we will define a new group,  $\text{Re}_j$ , containing the integers of color errors of the pixels included in  $j^{\text{th}}$  region and it is defined such as:

$$\text{Re}_j = \{\text{int}(e(px_{i,j})) : px_{i,j} \in R_j\} \quad (11)$$

Let  $\text{Re}_j$  is split into two groups  $\text{Re}'_j$  &  $\text{Re}''_j$ , where  $\text{Re}'_j$  contains the integers of color errors of the near similar pixels (homogeneous) and  $\text{Re}''_j$  contains the integers of the rest color errors, we define  $\text{Re}'_j$  such as:

$$\text{Re}'_j = \{\text{int}(e(px_{i,j})) : px_{i,j} \in R_j \ \& \ \text{abs}(e(px_{i,j})) < d\} \quad (12)$$

Where  $d$  is a constant value. Empirically, we found that the proper value of  $d$  falls in the range  $[0, 0.3]$  and the value  $d=0.15$  gives us good evaluation results. Also, we found that the value of  $d$  can be detected mathematically according to the homogeneity of a given image - we will illustrate this principle later. Thus,  $d$  divides the pixels of regions in a segmented image into near similar and non-similar pixels according to the color errors. Consequently, we have  $I'_e$  &  $I''_e$ . If  $n$  denotes to the number of regions in a segmented image, then:

$$I'_e = \bigcup_{i=1}^n R'_e i \quad (13)$$

$$I''_e = \bigcup_{i=1}^n R''_e i \quad (14)$$

$Q^\setminus$  function as  $E$  function has two components,  $H_1^\setminus$  &  $H_e$ , where  $H_1^\setminus$  represents the layout entropy and  $H_e$  represents the entropy of color errors. The working principle of  $Q^\setminus$  function is a lower entropy value means a better segmentation result and vice versa, and it is computed such as:

$$Q^\setminus = w_1 H_1^\setminus(I) + w_2 H_e(I) \quad (15)$$

Where,  $w_1$  &  $w_2$  are weighting parameters. Empirically, we found that  $Q^\setminus$  function gives us a good evaluation if the value of  $w_1$  is greater than the value of  $w_2$ . Throughout this work we will set  $w_1 = 0.55$  and  $w_2 = 0.45$ . The users can tailor these parameters according to their objectives. The first term,  $H_1^\setminus$ , represents the layout entropy, and it is computed such as:

$$H_1^\setminus = - \sum_{j=1}^n \frac{L(R'_e j)}{Na} \log_2 \frac{L(R'_e j)}{Na} \quad (16)$$

From the information coding theory, the above function indicates the number of bits needed to specify a region id in a segmented image according to the number of near similar pixels in different regions. The second term,  $H_e$ , represents the entropy of color errors in the overall segmented image and it is computed such as:

$$H_e = - \sum_{z=\min_e}^{\max_e} \frac{L(z)}{Na} \log_2 \frac{L(z)}{Na} \quad (17)$$

Where,  $\min_e$  and  $\max_e$  represent the minimum and maximum color errors in  $I''_e$  respectively. From the information coding theory,  $H_e$  indicates the number of bits needed to encode the color errors of non-similar pixels. The entropy of color errors serves in a similar way to the term involving the squared color error used in  $Q$  function.

The two entropies of  $H_1^\setminus$  &  $H_e$  components are computed according to the number of pixels in the overall image,  $Na$ . This make  $H_1^\setminus$  and  $H_e$  are complements to each other making  $Q^\setminus$  function gives more balancing between the number of regions and their homogeneities. The value of  $H_1^\setminus$  is increased if the number of regions in a segmented image increases and vice versa and the value of  $H_e$  is increased if the homogeneities of regions decrease and vice versa. In the case of a segmented image with one pixel per region, the first term is maximized to a very large value and the second term is minimized to zero. This means that  $Q^\setminus$  function gives a large value in the case of non-segmented image.

- $d$  detection

The value of  $d$  splits the pixels in a region into near-similar (homogeneous) and non-similar (non-homogeneous) so this value can be extracted mathematically according to the homogeneity of the overall image,  $H(I)$ . Entropy theory is one of the ways used to measure the homogeneity of images. Let  $g_{\max}$  is the maximum gray level in a given image,  $I$ , then the homogeneity of  $I$  is computed depending on the image entropy as follows:

$$H(I) = - \sum_{z=0}^{g_{\max}} \frac{L(z)}{Na} \log_2 \frac{L(z)}{Na} \quad (18)$$

In the case of images having 256 gray levels, the number of bits needed to represent a pixel is 8 bits per pixel. So if the value of  $H(I)$  is less than 4 bits per pixels, this means that the pixels in the given image are near-similar. In this case, the value of  $d$  is equaled to zero. In other cases the value of  $d$  is equaled to  $(H(I) - 4)$ . Empirically, we found that the proper value of  $d$  falls in the range  $[0, 0.3]$ . So the value of  $d$  after it has been computed is scaled to be in that range.

#### IV. A NEW IMAGE SEGMENTATION METHOD BASED ON PSO

This section develops a new multilevel thresholding method segmenting images using the principle of PSO. The new method will be named PSOTH. The algorithm of PSO, in PSOTH, tries to find near-optimal values of thresholds that can give us a near-optimal segmentation. Consequently, the PSO algorithm initializes a random swarm of  $m$  particles, where each one has its  $k$  thresholds, and flies them on a search space to look after the target partition according to a fitness function. The PSO algorithm uses  $Q^{\setminus}$  function as a fitness function and the gray levels  $\{g_{\min}, \dots, g_{\max}\}$  as a search space, where  $g_{\min}$  and  $g_{\max}$  are the minimum and maximum gray levels in a given image respectively.

##### A. PSO Representation

One of the key issues in designing a successful PSO algorithm is the representation step, i.e. finding a suitable mapping between a problem and PSO particles. Figure 2 shows the PSO representation that is used in PSOTH. As it is shown, PSO has the following information:

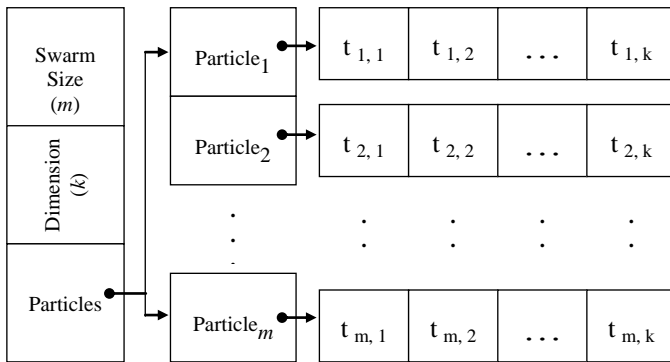


Figure 2: PSO Representation in the PSOTH Method

In PSO of PSOTH method, a single particle represents  $k$  thresholds. That is, each vector  $x_i$  is constructed such as  $x_i = (t_{i,1}, \dots, t_{i,j}, \dots, t_{i,k})$  where  $t_{i,j}$  refers to the  $j^{\text{th}}$  threshold value of the  $i^{\text{th}}$  particle. Therefore, in the PSOTH method, after applying the thresholding to each particle, a swarm represents a number of segmented images.

##### B. Fitness Function

In PSOTH, the algorithm of PSO uses the  $Q^{\setminus}$  evaluation function presented above as a fitness function. The idea behind using  $Q^{\setminus}$  function can be formulated as: the smaller the value of  $Q^{\setminus}$ , the better is the segmentation result. Consequently, The PSO algorithm, in the PSOTH method, looks after the threshold values that minimize the value of  $Q^{\setminus}$ . It should be noted that  $Q^{\setminus}$  function manipulates with a segmented image as a set of regions. So, in PSOTH, a given image is thresholded into a set of regions (where a region is a group of connected pixels located in the same rang of gray levels). Let there be  $L$  gray levels in the given image and these gray levels are in the range  $M=\{g_{\min}, \dots, g_{\max}\}$ , So the PSO algorithm tries to obtain a near optimal  $k$ -dimensional vector  $\{t_1, t_2, \dots, t_k\} \subset M$

that thresholds the given image into regions which can minimize the value of  $Q^{\setminus}$  function shown in Equation 15.

##### C. PSOTH Algorithm

Firstly, the algorithm of PSOTH starts by applying (3x3) low pass filter to remove a noise from a target image, after that, every particle in the swarm is initialized randomly to contain  $k$  thresholds. Secondly, the target image is thresholded and divided into regions using the threshold values of each particle separately, and the fitness function, which is the  $Q^{\setminus}$  evaluation function, is calculated for each particle. Once the fitness values have been found the global best solution,  $g_{best}$ , is computed and the updates of PSO velocities and vectors are then done. This procedure is repeated until the number of iterations has been satisfied. The algorithm of PSOTH is summarized in Algorithm 1.

Algorithm 1: PSOTH Algorithm	
1-	<b>Do</b> (3 x 3) Low pass filter.
2-	<b>Initialize</b> randomly each particle to contain $k$ threshold values.
3-	<b>For</b> each particle, $i$ , <b>Sort</b> the threshold values. // <i>Optional</i>
3-	<b>Repeat</b> the following Until the number of iterations has been satisfied
(a)	<b>For</b> each particle $i$ :
(i)	<b>Threshold</b> the target image into regions using the threshold values of particle $i$ .
(ii)	<b>Compute</b> the fitness of particle $i$ by using $Q^{\setminus}$ function.
(b)	<b>Find</b> the global best solution ( $g_{best}$ ).
(c)	<b>For</b> each particle $i$ , <b>Update</b> the threshold values.
4-	<b>Fix</b> the $g_{best}$ as a best solution.
5-	<b>Merge</b> small regions of the best solution.

The algorithm initializes the vectors of each particle randomly. The threshold values of each particle are initialized in the range  $[g_{\min}, g_{\max}]$ . The thresholds are sorted in our method optionally. The sorting step is added to increase the symmetric between the thresholds in different particles.

To threshold and group a target image into regions, the principle of normal region based segmentation is applied as follows. The image is scanned from left to right and from top to bottom and the current pixel,  $px$ , is used as a seed point growing with its neighboring pixels according to a connectivity and a similarity condition. Here, the 4-connectivity is used and the similarity condition is formulated using the range of gray levels as follows. If  $g(px)$  falls in a rang  $[t_{i,j}, t_{i,j+1}]$  then a neighbored pixel  $N(px)$  append to the region of  $px$ , if  $g(N(px))$  also falls in the rang  $[t_{i,j}, t_{i,j+1}]$ . Where,  $[t_{i,j}, t_{i,j+1}]$  is the rang of gray levels specified by two threshold values  $t_{i,j}$  and  $t_{i,j+1}$  of  $i^{\text{th}}$  particle,  $g(px)$  is the gray level of  $px$ , and  $g(N(px))$  refers to the gray level of a pixel neighboring to the region  $px$ , in 4-connectivity. The algorithm of thresholding and grouping the target image into regions is summarized in Algorithm 2.

The fitness of every particle is evaluated by using the quantitative evaluation function,  $Q^{\setminus}$ , shown in Equation 15.  $Q^{\setminus}$  function uses the information stored in the list of regions to

Algorithm 2: Grouping a target image into regions.

```
- For each pixel, px, in the target image.  
- If Pixel, px, is ungrouped Then  
  (i) Grow px with its neighboring pixels according to the  
  threshold values of particle i.  
  (ii) - Add the new region to the list of regions.  
- End if  
- End For
```

get the fitness of a particle. The gbest takes the position of the particle having the best fitness value. Once the gbest has been found, the threshold values of each particle are updated using Equation (3) and (4) in 2-dimension space ( $i, j$ ), where  $i$  represents  $i$ th particle and  $j$  represents  $j$ th threshold.

Step 3 of PSOTH algorithm is repeated until the stopping criterion has been satisfied. The segmented image of the global best solution,  $gbest$ , after finishing the final iteration, is chosen as a best solution. For the best segmentation, merging small regions must be done. In the merging step, the final list of regions is scanned to merge regions having pixels less than a

predefined number (e.g. 10 pixels), with their neighboring regions that satisfy a similarity condition. The similarity condition that is used in merging processes is the smallest difference between means. In the algorithm of PSOTH, the number of final regions is determined after merging small regions step.

## V. EXPERIMENTAL RESULTS

In order to prove the efficiency and accuracy of PSOTH, it has been applied to many different images. In this section we will present the results produced through applying the PSOTH method to the three images shown in Figure 3. These images have been selected to test the algorithm of PSOTH, and to compare it with other algorithms. All the test images are 256x256 pixels in size. This section is organized as follows. Section A presents the examination of  $Q^{\setminus}$  function. Section B shows the tracing steps of PSOTH algorithm and Section C shows the performance measuring of PSOTH method.

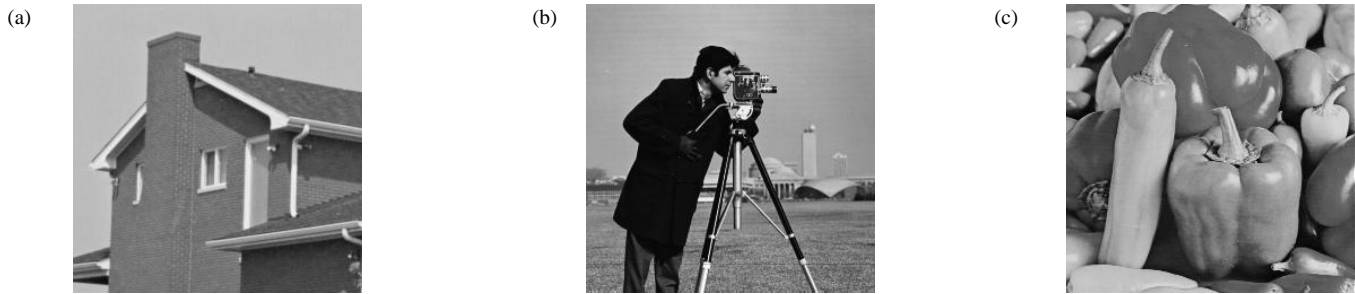


Figure 3: Test Images (a). House. (b). Cameraman. (c). Peppers.

### A. $Q^{\setminus}$ Function Examination

$Q^{\setminus}$  function was examined through applying it to evaluate the segmentation results produced through segmenting the test images using the principle of normal region-based segmentation. Here, the normal region based segmentation segments the test images through multithresholding them into a group of regions using  $k$  thresholds, where  $k$  thresholds divide the gray levels into a number of gray level ranges with equal interval. Consequently, a test image scanned from left to right and left to bottom, and the connected pixels that fall in the same range of gray level grouped into a single region. This action is repeated for  $n=40$  iterations. In each iteration, a

different segmentation is produced, where  $k$  is set to the number of iteration so  $k = \{1, 2, \dots, n\}$ .

Figure 4 shows the best segmentations that were chosen by  $Q^{\setminus}$  function. To show the efficiency of  $Q^{\setminus}$  function for choosing the best segmentation, the results produced by  $Q^{\setminus}$  will be compared with the results chosen by  $Q$  function. Figure 5 shows the best segmentations that were produced by  $Q$  function. With human visual perception, it can be seen that the results produced by  $Q^{\setminus}$  function are better than the results produced by  $Q$  function.

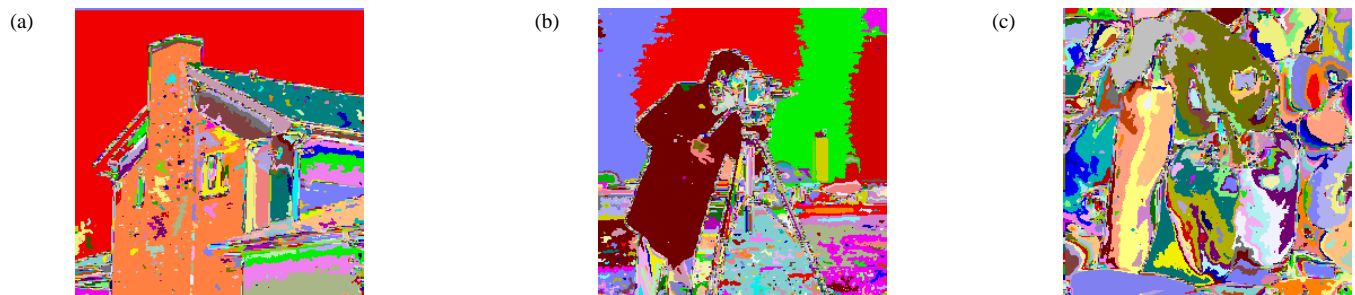


Figure 4: The best segmentations choosing by  $Q^{\setminus}$  function for (a). Cameraman image. (b). House Image (c).Peppers image.

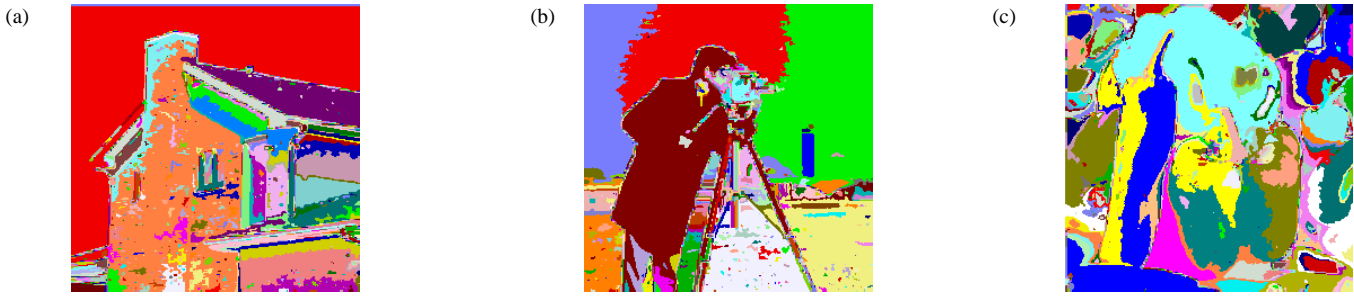


Figure 5: The best segmentations choosing by  $Q$  function for (a). Cameraman image. (b). House Image (c).Peppers image.

### B. Tracing Steps of PSOTH Algorithm

For tracing purpose, the algorithm of PSOTH is used to get a near optimal segmentation for Cameraman image. Firstly, the algorithm starts by applying (3x3) low pass filter, Figure 6(a) shows the filtered image, and, then, the particles are randomly initialized. The x-vector and v-vector values are initialized in the ranges  $[g_{min}, g_{max}]$  and  $[V_{max}, -V_{max}]$  respectively, where  $g_{min} = 13$ ,  $g_{max} = 233$  for Cameraman image.

For the PSO algorithm, in this experiment, 20 particles are trained for 50 iterations. The PSO parameters are initially set as follows.  $V_{max} = 3$ ,  $w = 1$  decreased over time to 0.4 and  $C_1 = C_2 = 1.49$ . The particles are trained with  $k=10$ . Empirically, we found that the proper value of  $k$  falls in the range  $[7, 12]$ , where if we increase the value of  $k$ , more details will be produced and if we decrease it, more flat regions will be produced.

The algorithm of PSO starts by segmenting the target image using the gray level ranges of each particle and computing the fitness of each one using the  $Q$  evaluation function. The flow chart shown in Figure 7 illustrates the values of  $Q$  related to each particle in the first iteration. In the flow chart, the best particle, 2<sup>nd</sup> particle, is white in color and the worst particle,

11<sup>th</sup> particle, is black in color. Figure 6(b) and Figure 6(c) show the best and worst segmentation in the first iteration, i.e. illustrating that the segmentation of 2<sup>nd</sup> particle is clearly better than the segmentation of 11<sup>th</sup> particle. After that, the particle having the best fitness value is taken as *gbest*. Finally the velocities and positions of each particle are updated respectively according to the updating functions. These steps are repeated until the number of iterations has been satisfied. Figure 8 illustrates for the target image how the fitness of PSO improves over time. The fitness value, as measured using  $Q$  function, improved from the initial 0.2720554 to 0.1326954.

After that, the segmented image of *gbest* is chosen as a near optimal segmentation (the best solution), Figure 6(d) shows the best solution choosing by PSO. For the best solution small regions having pixels less than 10 pixels are merged with their neighboring regions. Figure 6(e) shows the best solution after merging small regions. Clearly, we can see that the segmentation in Figure 6(e) is better than the segmentation shown in Figure 6(c). This means that the segmentation of House image was improved over item according to the values produced by  $Q$  function. Finally, Figure 6(f) shows the regions of the best solution after filling with their means.

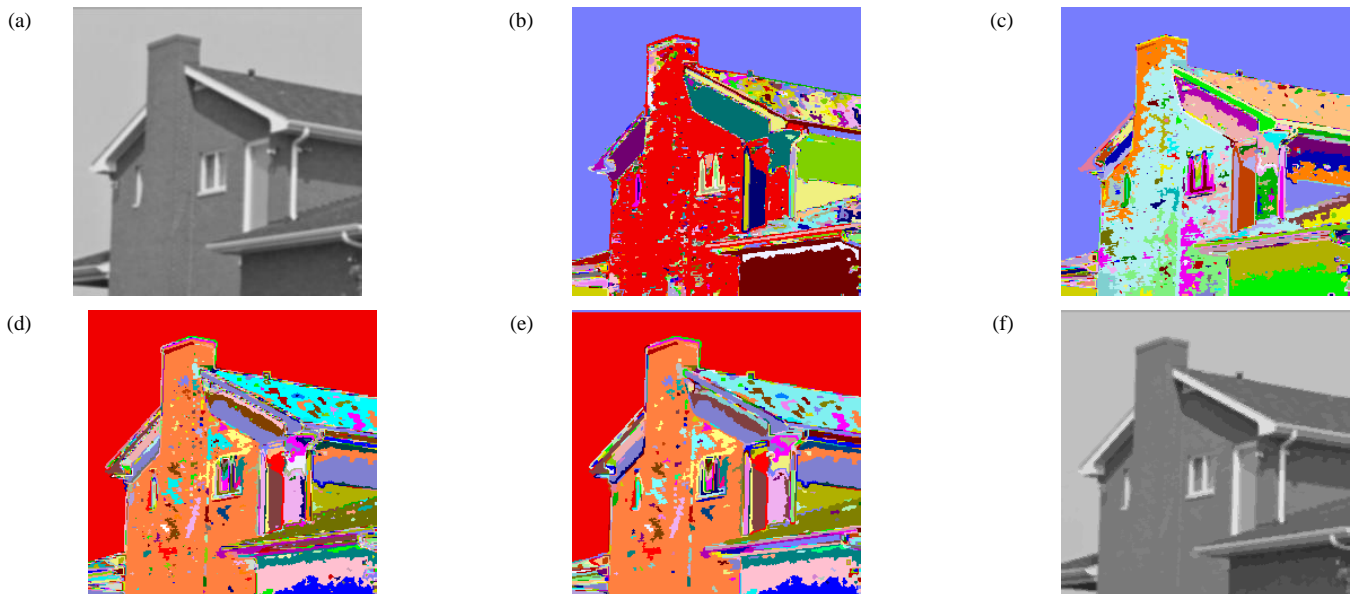


Figure 6: Tracing steps of PSOTH algorithm.

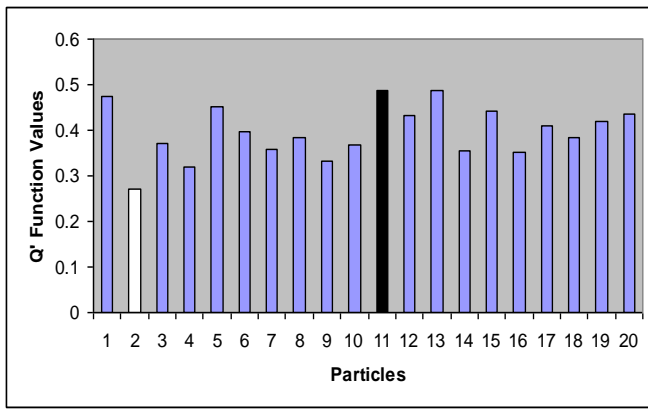


Figure 7: Fitness of Particles in First Iteration.

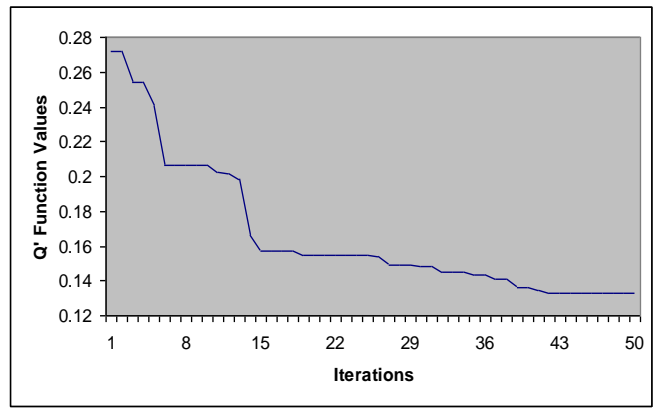


Figure 8: PSOTH Performance on House Image

### C. Performance Measuring

Comparing different segmentation algorithms with each other is difficult mainly because they differ in the properties they try to satisfy [33]. For the comparison of the PSOTH method, the segmentation results of it will be compared with the results produced by implementing the algorithm of  $k$ -means and the normal region-based segmentation as discussed in previous section.

In all the test images, for PSOTH method, 20 particles were trained for 50 iterations,  $k=10$ ,  $V_{max} = 3$ ,  $w = 1$  decreased over time to 0.4,  $c_1 = c_2 = 1.49$  and for the fitness function,  $Q'$  function,  $w_1 = 0.55$ ,  $w_2 = 0.45$ . For the  $k$ -means algorithm, 100 iterations were used and  $k=10$  centroids selected randomly in the range  $[g_{min}, g_{max}]$  of each image separately. For region

based method,  $n=40$  iterations was used and  $k= \{1, 2, \dots, n\}$  depending on the number of iteration.

Figures 9(a, b, c) illustrate the segmented images produced through applying the PSOTH method to the test images, while Figures 9(d, e, f) and Figures 9(g, h, i) illustrate the segmented images obtained from applying the  $k$ -mean algorithm and the region based segmentation to the test images respectively. With human visual perception, it can be seen that the results shown in the figures clearly illustrate that the segmented images obtained by the PSOTH method are better than the segmented images obtained by the region based segmentation and the  $k$ -means algorithm. This means that the PSO algorithm is a good optimization method can be used to segment images if it is complicated with a good fitness function such as  $Q'$  function.

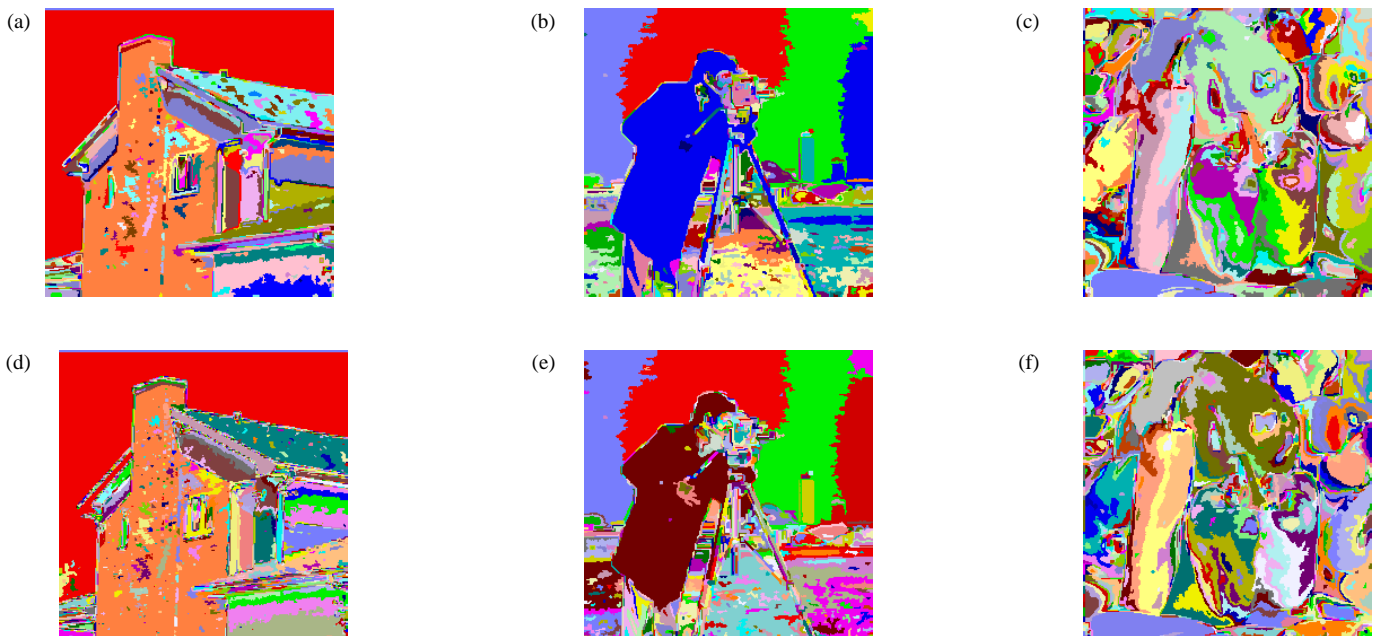


Figure 9: The best Segmentation of Test Images Produced by (a, b, c). PSOTH method. (d, e, f). Normal region based algorithm. (g, h, i)  $k$ -means algorithm.

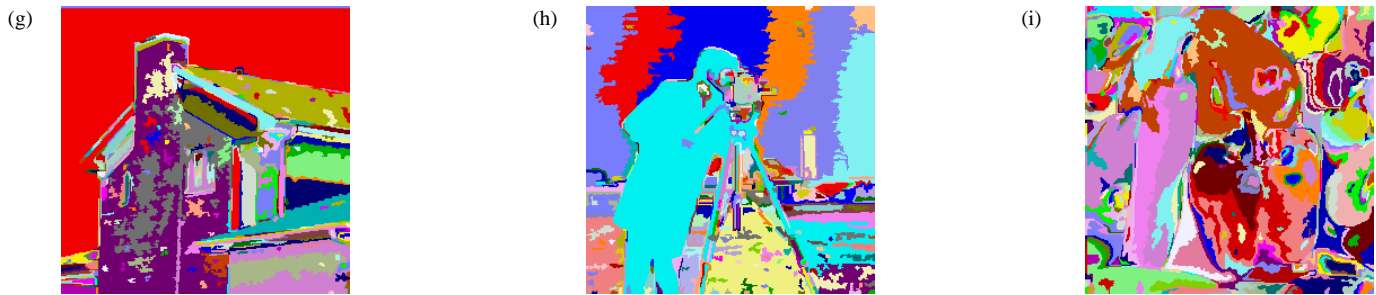


Figure 9: Continue;

### • Evaluation Measuring

Table 1 summarizes the values of  $Q^{\setminus}$  function produced by applying the normal region based segmentation,  $k$ -means algorithm and PSOTH method to the test images respectively (the values in Table 1 are the evaluations of segmented images in Figure 9). The results show that, in all cases, the  $k$ -means algorithm performed better than the region based segmentation method, and the PSOTH method performed better than both the region based method and  $k$ -means algorithm. Note that the values of  $Q^{\setminus}$  function were scaled to be in the range [0, 10].

TABLE I. COMPARISON BETWEEN REGION BASED, K-MEANS AND PSOTH

Images	Region Based	$k$ -means	PSOTH
House	0.2030503	0.1806134	0.1326954
Cameraman	0.1972899	0.1503304	0.1463228
Peppers	0.1692429	0.1587412	0.1372648

### VI. CONCLUSION

In this work, PSO has been used to produce a new optimization-based image segmentation method, PSOTH. In the PSOTH method, the algorithm of PSO tries to find a near optimal segmentation for a given image using a fitness function. PSO is a flexible optimization method, where many objective functions can be used. For this reason, a new quantitative evaluation function for segmented images has been proposed in this paper. So in the PSOTH method, the new evaluation function has been used as a fitness function for the algorithm of PSO. The experimental results have illustrated that the efficiencies of the PSOTH method and the new evaluation function.

### REFERENCES

- [1] E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, NY: Oxford University Press, 1999.
- [2] J. Kennedy and R. Eberhart, "Particle swarm optimization," in the Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, Vol. 4, pp. 1942-1948, 1995.
- [3] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New-York, 1973.
- [4] A. A. Younes, I. Truck, and H. Akdaj, "Color Image Profiling Using Fuzzy Sets," *Turk J Elec. Engin.*, Vol.13, No.3, 2005.
- [5] M. Sezgin, B. Sankur, "Survey over image thresholding techniques and

- quantitative performance evaluation," *J. Electron. Imaging*, Vol. 13, No. 1, pp. 146-165, 2004.
- [6] J. Marcello, F. Marques and F. Eugenio, "Evaluation of thresholding techniques applied to oceanographic remote sensing imagery," *SPIE*, 5573, pp. 96-103, 2004.
- [7] P. K. Sahoo, S. Soltani, A. K. C. Wong, Y. C. Chen, "A survey of thresholding techniques," *Computer Vision, Graphics and Image Processing*, Vol. 41, pp.233-260, 1988.
- [8] E. Zahara, S. S. Fan and D. Tsai, "Optimal multi-thresholding using a hybrid optimization approach," *Pattern Recognition Letters*, Elsevier, Vol. 26, pp. 1082-1095, 2005.
- [9] P. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," *Applied Mathematics and Computation*, Vol. 184, pp. 503-513, 2007.
- [10] M. Maitra and A. Chatterjee, "A hybrid cooperative-comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding," *Expert Systems with Applications*, Vol. 34, pp. 1341-1350, 2008.
- [11] Y. zhiwei, C. hongwei, L. wei and Z. jinping, "Automatic threshold selection based on Particle Swarm Optimization algorithm," in the proceedings International Conference on Intelligent Computation Technology and Automation, pp. 36-39, 2008
- [12] C. Wei and F. Kangling, "Multilevel Thresholding Algorithm Based on Particle Swarm Optimization for Image Segmentation," in the Proceedings of the 27th Chinese Control Conference, July 16-18, Kunming, Yunnan, China, pp. 348-351, 2008
- [13] P. D. Sathya, R. Kayalvizhi, "PSO-Based Tsallis Thresholding Selection Procedure for Image Segmentation," *International Journal of Computer Applications*, Vol. 5, No. 4, pp. 39-46, 2010.
- [14] T. Hongmei, W. Cuixia, H. Liying, and W. Xia, "Image Segmentation Based on Improved PSO," the proceedings of the International Conference on Computer and Communication Technologies in Agriculture Engineering(CCTAE2010), pp. 191-194, 2010.
- [15] J. Kennedy, and R. Eberhart, *Swarm Intelligence*, San Francisco: Morgan Kaufmann Publishers, 2001.
- [16] Y. Shi, and R. Eberhart, "A modified particle swarm optimizer," in the Proceedings of the IEEE International Conference on Evolutionary Computation, Piscataway, NJ, pp. 69-73, 1998.
- [17] Liu J., Yang Y. H., "Multiresolution Color Image Segmentation," *IEEE Trans. on PAMI*, Vol. 16, No. 7, pp. 689-700, 1994.
- [18] Borsotti M., Campadelli P., Schettini R., "Quantitative Evaluation of Color Image Segmentation Results," *Pattern Recognition Letters*, Elsevier, vol.19, pp. 741-747, 1998.
- [19] H. Zhang, J. Fritts, and S. Goldman, "An entropy-based objective evaluation method for image segmentation," in Proc. SPIE- Storage and Retrieval Methods and Applications for Multimedia, 2004.
- [20] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition*, Elsevier, Vol. 29, No. 8, pp. 1335-1346, 1996.
- [21] Hui Zhang, Jason E. Fritts, and Sally A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," *Computer Vision and Image Understanding*, Vol. 110, No. 2, pp.260-280, 2008



# Image segmentation by adaptive distance based on EM algorithm

Mohamed Ali Mahjoub<sup>1</sup>

<sup>1</sup>Preparatory Institute of Engineer of Monastir  
Street Ibn Eljazzar Monastir  
Tunisia

Karim Kalti<sup>2</sup>

<sup>2</sup>Faculty of science of Monastir  
Street Kairouan - Monastir  
Tunisia

**Abstract**— This paper introduces a Bayesian image segmentation algorithm based on finite mixtures. An EM algorithm is developed to estimate parameters of the Gaussian mixtures. The finite mixture is a flexible and powerful probabilistic modeling tool. It can be used to provide a model-based clustering in the field of pattern recognition. However, the application of finite mixtures to image segmentation presents some difficulties; especially it's sensible to noise. In this paper we propose a variant of this method which aims to resolve this problem. Our approach proceeds by the characterization of pixels by two features: the first one describes the intrinsic properties of the pixel and the second characterizes the neighborhood of pixel. Then the classification is made on the base on adaptive distance which privileges the one or the other features according to the spatial position of the pixel in the image. The obtained results have shown a significant improvement of our approach compared to the standard version of EM algorithm.

**Keywords**- EM algorithm, image segmentation, adaptive distance

## I. INTRODUCTION

Image segmentation is one of the major challenges in image processing and computer vision. When the problem of image segmentation is approached by a process of classification, several methods exist. Recently, finite mixture models have attracted considerable interest for image segmentation [2,8,16,17]. However, the application of finite mixtures model to image segmentation presents some difficulties. For the classical mixture statistical model each pixel must be associated with exactly one class. This assumption may be not realistic. Thus, several methods have been proposed to circumvent this problem. For example, the fuzzy-c mean algorithm has widely been used in image segmentation. Also, some methods mixing fuzzy and statistical model have been developed by Gath. Recently, the Markov Random Field (MRF) models were used with images in an important number of works to add spatial smoothness into the process of image segmentation [11,13,15]. This approach provide satisfactory results in many case, but most case the assumption of a single Gaussian distribution typically limits image segmentation accuracy. The segmentation algorithm developed in this paper is based on a parametric model in which the probability density function of the gray levels in the image is a mixture of Gaussian density functions. This model has received considerable attention in the development of segmentation algorithms and it has been noted that the performance is influenced by the shape of the image

histogram and the accuracy of the estimates of the model parameters. However, the model-based segmentation algorithms not allows a good results if the histogram of an image is a poor Gaussian approximation. The application of this model in image segmentation is, therefore, limited to the images which are a good approximations of Gaussian mixtures with well-defined modes.

Among the techniques of classification, we used in this work models of Gaussian mixtures. An EM algorithm is developed to estimate parameters of the Gaussian mixtures. The finite mixture is a flexible and powerful probabilistic modeling tool. It can be used to provide a model-based clustering in the field of pattern recognition. However, the application of finite mixtures to image segmentation presents some difficulties; especially it's sensible to noise. In this paper we propose a variant of this method which aims to resolve this problem.

The remainder of the paper is organized as follows. Section 2 describes the EM algorithm briefly and the parameter estimation of multi-variate normal densities by the EM algorithm. In Section 3 an unsupervised segmentation algorithm for grey level images is proposed on the basis of the ML estimation. Section 4 presents experimental results to show the performance of the proposed algorithm, and Section 5 concludes by summarizing the paper

## II. SEGMENTATION METHOD BASED ON GMM-EM

The expectation-maximization algorithm (in English Expectation-maximization algorithm), proposed by Dempster [19], is a class of algorithms that find the maximum likelihood parameters of probabilistic models where the model depends on unobserved variables. The finite mixture of distributions has provided a mathematical approach to the statistical modeling of a wide variety of random phenomena. In the past decades, the extent and the potential of the applications of finite mixture models have widened considerably. In the field of pattern recognition, the finite mixture can be used to provide a model-based clustering. A finite mixture density has the form :

$$f(x|\theta) = \sum_{i=1}^K p_i f_i(x|\theta_i)$$

Where :

- $\alpha_i$  is the proportion of the class  $i$  ( $\alpha_i \geq 0$  and  $\sum_i \alpha_i = 1$ )

$\theta_i = (\mu_i, \Sigma_i)$ ,  $\mu_i$  et  $\Sigma_i$  are respectively the center and the variance matrix of  $k^{\text{th}}$  normal component  $f(\cdot | \alpha_i)$ .

The evaluation of all these parameters can be calculated by maximizing the log-likelihood of the global parameter  $\theta$ :

$$\ln f(X|\theta) = \sum_{j=1}^n \ln f(x_j|\theta) \quad (1)$$

Where  $X=(x_1, \dots, x_n)$ , this maximization can be made by the EM algorithm of Dempster. Besides, the  $k$  number of components can also be estimated while keeping the value between  $k=1$  and  $k_{\text{sup}}$  ( $k_{\text{sup}}$  to choose a priori) that minimizes the Bayesian BIC criteria.

$$\text{BIC}(K) = -2 \ln f(X|\hat{\theta}_K) + v_K \ln n \quad (2)$$

Where  $\hat{\theta}_K$  and  $v_K$  are respectively the maximum likelihood estimator and the number of degrees of freedom model.

#### A. Classification

Any classification model is defined on the space  $N$  of maps from the image domain to the set  $N$  of classes (each class  $n$  corresponds to an entity of interest in the scene). Thus each classification  $v \in N$  assigns a class  $n = v(p) \in N$  to each pixel  $p$  giving the class of that pixel. By defining a posterior probability distribution on  $N$ , and using a suitable loss function, an optimal classification can be chosen. The loss function is more often than not taken to be the negative of a delta function, the resulting estimate then being a maximum a posteriori (MAP) estimate. The posterior distribution is expressed as the (normalized) product of a likelihood, such as the GMM (Gaussian Mixture Models) models that we will discuss in this paper, which gives the distribution of images corresponding to a given class, and a prior probability distribution on the classifications.

#### B. Parameters estimation

The EM (expectation-maximization) algorithm is an iterative approach to compute maximum-likelihood estimates when the observations are incomplete. In the mixture density estimation, the information that indicates the component from which the observable sample originates is unobservable. Expectation Maximization (EM) is one of the most common algorithms used for density estimation of data points in an unsupervised setting. In EM, alternating steps of Expectation (E) and Maximization (M) are performed iteratively till the results converge. The E step computes an expectation of the likelihood by including the latent variables as if they were observed, and a maximization (M) step, which computes the maximum likelihood estimates of the parameters by maximizing the expected likelihood found on the last E step. The parameters found on the M step are then used to begin another E step, and the process is repeated until convergence.

The EM algorithm is very used for the research of the parameter achieving the maximum likelihood [19,20]. The criteria of stop of the algorithm, is either a maximum number of iterations to limit the time of calculation, either a lower mistake. It is put easily in application because it leans on the calculation of the complete data. The EM algorithm requires the initialization of model parameters of Gaussian mixture.

The covariance matrices are initialized by the identity matrix, and  $K$  mean vectors are initialized by the various centers of Gaussian mixture estimated by the algorithm K-means.

Let's suppose that we have  $K$  components in the model of mixtures. The shape of the density of probability of this mixture is given by :

$$f(x|\theta) = \sum_{i=1}^k \alpha_i f_i(x|\theta_i) \quad (3)$$

where  $x$  is the characteristic vector,  $\alpha_i$  is the weight of the mixture as  $\sum_{i=1}^k \alpha_i = 1$ ,  $\theta$  represents the parameters  $(\alpha_1, \alpha_2, \dots, \alpha_k, \theta_1, \theta_2, \dots, \theta_k)$  and  $f_i$  the density of the gaussian parameterized by  $\theta_i$  that is to say  $(\mu_i, \sigma_i)$  :

$$P(x|\theta_i) = \frac{1}{\sqrt{2\pi\sigma_i}} \exp \left\{ -\frac{(x - \mu_i)^2}{2\sigma_i^2} \right\} \quad i = 1, 2, \dots, N$$

where  $\theta_i = (\mu_i, \sigma_i)$  is the Gaussian mixture distribution parameter.

Assume that the density is derived from a mixture of Gaussians. That is to say that :  $f(x) = \sum_{i=1}^k \alpha_i \varphi(x, \mu_i, \Sigma_i)$ . We will then estimate the parameters by maximizing the likelihood. For this, we must start with a number of  $K$  Gaussian fixed a priori and then seek a local maximum in the first order conditions. Thus, the EM algorithm is summarized as follows :

Input : H=histogram, k=gaussian number,  $\epsilon$  = Error  
 Output : model parameters  $(\alpha_1, \alpha_2, \dots, \alpha_k, \theta_1, \theta_2, \dots, \theta_k)$   
 Steps :

- 1- Evaluation expectancy (E) :
 
$$\varphi(i|x_j, \theta) = \frac{p_i f_i(x_j|\theta_i)}{\sum_{k=1}^K \alpha_k f_k(x_j|\theta_k)}$$
- 2- Maximization setp (M) : GMM parameters update
 
$$\alpha_i^{\text{new}} = \frac{1}{N} \sum_{j=1}^N \varphi(i|x_j, \theta^{\text{old}})$$

$$\mu_i^{\text{new}} = \frac{\sum_{j=1}^N x_j \varphi(i|x_j, \theta^{\text{old}})}{\sum_{j=1}^N \varphi(i|x_j, \theta^{\text{old}})}$$

$$\sum_i^{\text{new}} = \frac{\sum_{j=1}^N \varphi(i|x_j, \theta^{\text{old}}) (x_j - \mu_i^{\text{new}})(x_j - \mu_i^{\text{new}})^T}{\sum_{j=1}^N \varphi(i|x_j, \theta^{\text{old}})}$$

We stop when  $\|\theta^{\text{new}} - \theta^{\text{old}}\| \leq \epsilon$

Algorithm 1: EM Algorithm

However, during the classification phase, we can modify this algorithm to classify a given pixel by calculating the distance to the center of the class instead of a probability calculation. the modified algorithm is called DEM (Distance EM) algorithm (figure 1).

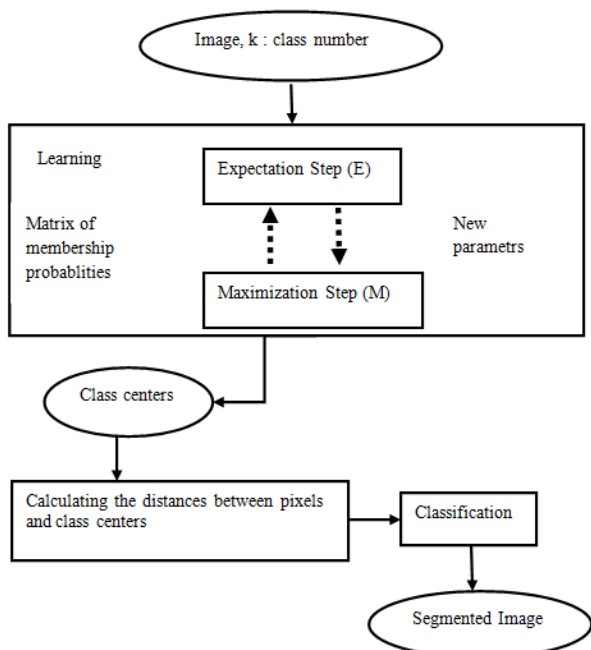


Figure 1: Pixels classification by EM algorithm

### III. LIMITS OF THE GMM SEGMENTATION AND POSSIBLE SOLUTION

The application of standard GMM (using a grey level as a single feature of pixel) yields a good segmentation of pixels inside regions and pixels of contours. However, it includes a bad clustering of noisy pixels of white region. This clustering drawback is essentially due to the only utilization of the intrinsic feature of pixel to be classified (gray level) without take into account the information relative to spatial position in the image. This information turns out to be important in the segmentation context. To overcome this limitation, we propose a solution consisting to include the neighborhood effect of pixel to be classified. To accomplish this effect, we have used in this work the arithmetic average estimator ( $\mu$ ) defined as follows:

$$\mu(x_j) = \frac{1}{N} \sum_{k=1}^N x_k$$

where  $x_j$  designates the pixel to classify,  $x_k$  designates a pixel belonging to the analysis window that determines the neighborhood and centered on  $x_j$ ,  $N$  designates the number of pixels of the analysis window.

The manner with which will be integrated the describer of neighborhood requires the knowledge of the effect of this last on the process of classification. We present in this paragraph a study of this effect on the noisy and contour pixels. The application of the GMM segmentation algorithm can generate a good classification of the internal pixels to the regions as well as the noisy pixels, but produces a deterioration of the contours between the two regions. Indeed, the effect of smoothing introduces by the average generated the attenuation of the noise since its value became enough close to those of its neighbors, thing that drove to its affectation to the same class that these last. This attenuation due to smoothing also affects

the pixels of the contours. The gap between the values of these last became less franc and drove toward a bad classification of some among them notably those that have in their setting more of pixels belonging at the neighboring region that to their own region. Of this study we present in figure 2 the advantage and the inconvenient of the use of the grey level and the spatial feature for the noise and the contours classification.

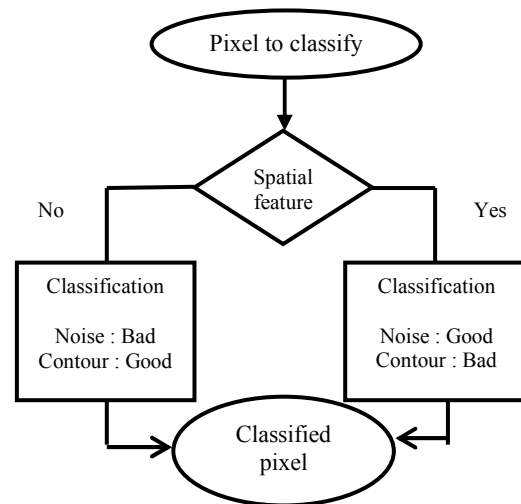


Figure 2 : Spatial feature and pixel classification

### IV. PROPOSED APPROACH

The complementarity of gray level and the average of the gray levels with regard to the classification can let consider an use joined of these two descriptors in the image segmentation with the help of the EM method. This method is inspired by the work of walid and kalti [1]. In this section we present a new version of the EM baptized Adaptive Distance EM (ADEM) that enrolls in this direction. This version tried to take advantage of the advantages of the two aforesaid featurures while avoiding their inconveniences and this while using an adaptive manner one or the other according to the spatial configuration of the pixel to classify.

In the setting of our work, we distinguished four possible spatial configurations for the pixels asking for each a specific choice of the criteria of classification (figure 3). These configurations are the following:

- PR : Pixel which belongs to a region
- PC : Pixel which belongs to a contour,
- NP : Noisy pixel,
- NNP : Neighbour of Noisy Pixel.

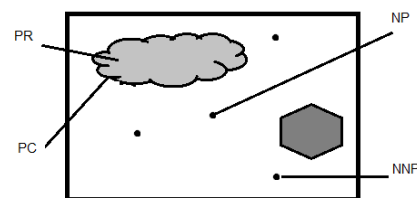


Figure 3 : Types of pixel noise/region/contour

Formally, the different spatial configurations are characterized by the two statistical descriptors presented as follows :

– The standard deviation ( $\sigma$ ) characterizes the dynamics of distribution around the pixel to be classified. This attribute is calculated as follows :

$$\sigma(x_j) = \sqrt{\frac{1}{N} \sum_{k=1}^N (x_k - \mu(x_j))^2} \quad (4)$$

– The  $NCN$  : represents the closest neighbours number in term of grey level regarding the considered pixel. It's defined as follows :

$$NCN(x_j) = \text{card}\{x_p \in \text{neighborhood}(x_j) / |x_p - x_j| < S\} \quad (5)$$

$S$  designates a threshold which is generally chosen in year empiric manner. From thesis two descriptors we can characterize the different spatial configurations possible of the pixels. In case of a PR the standard deviation  $\sigma$  is generally low, it is null for the constant regions. However the  $\sigma$  becomes high for gthe PC, NP and NNP. The distinction between these three configurations is made by using the  $NCN$  feature, which is generally low for a NP, moderate for a PC and high for a NNP.

The classification of a PR or a PB must privilege the spatial attribute because the decision must be taken on the basis of the information of its neighborhood. On the other hand the classification of a PC or a NCN must privilege the level of gray of the pixel (NG) respectively to preserve the contours and to avoid the influence of the noise. The choice of the criteria of classification as well as the characterizations of the spatial configurations are summarized in the figure 4.

#### A. Proposal of new parameter distance

The DEM algorithm generally uses to measure the similarity between an individual  $x_j$  and a class given by its center  $v_j$ , a distance that grants the same importance to the different attributes took in account in the process of classification. To introduce the adaptive effect as for the selection of the attributes, we propose to use a dynamic distance and weighted derivative of the Euclidian distance. This new distance is given by the equation 6.

$$D(x_j, v_j) = (1 - p_j)(x_j^{NG} - v_j^{NG})^2 + p_j(x_j^{Spatial} - v_j^{Spatial})^2 \quad (6)$$

$D$  is a bidimensional distance based on the two NG and Spatial attributs. In this equation, the weight  $p_j$  permits to control the importance of every attribute for the classification of the pixel  $x_j$ . So if the  $p_j$  is thigh we privileges the spatial attribute otherwise one privileges the gray level. The term  $p_j$  must be calculated for every pixel to classify according to its spatial configuration in the image. From the configurations

presented in figure 4, we can deduct that the weight  $p_j$  must be maximized (tender verse 1) when the pixel to classify is a PR or a PB, because the decision of  $\sigma$  of its adherence to the different

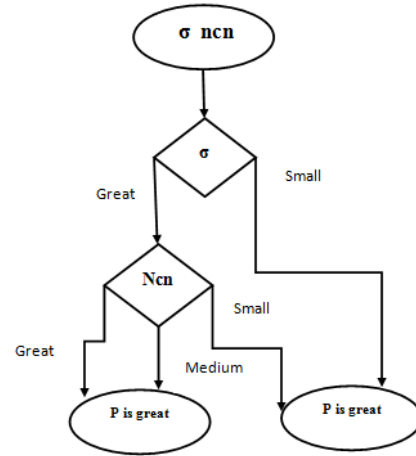


Figure 4 : Influence of  $\sigma$  and  $NCN$

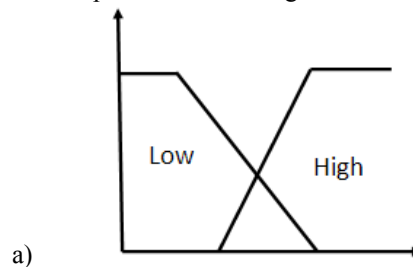
classes must be taken only on the basis of the spatial attribute. However this  $p_j$  must be minimized (tender verse 0) in case of a PC or a NNP because the gray level in these cases is going to constitute a good criteria of classification.

#### B. Estimation of the spatial weight

The choice of the spatial weight  $p_j$  is very important for the calculation of the new distance (6). We propose in this paragraph a method of fuzzy evaluation of this weight. For that we use a fuzzy system which processes as entries to linguistic variables of decision  $\sigma$  and  $NCN$  to give some results on the linguistic variable exit  $p$ . While considering the choices of the  $p$  in accordance with the spatial configurations, is it possible to define for each of these configurations, a fuzzy rule of the type SO THEN. We can deduct in this case four rules characterizing the relations between the classes of entry ( $\sigma$  and  $NCN$ ) with the corresponding exit class ( $p$ ) to determine all necessary consequences to calculate the value of  $p$ . The linguistic rules that we defined are the following :

- R1: IF  $\sigma$  is low THEN  $p$  is high
- R2: IF  $\sigma$  is high AND  $NCN$  is low THEN  $p$  is low
- R3: IF  $\sigma$  is high AND  $NCN$  is high THEN  $p$  is HIGH
- R4: IF  $\sigma$  is high AND  $NCN$  is moderate THEN  $p$  is low

The different membership functions of variables  $\sigma$ ,  $NCN$  and  $P$  are represented in the figure 5.



a)

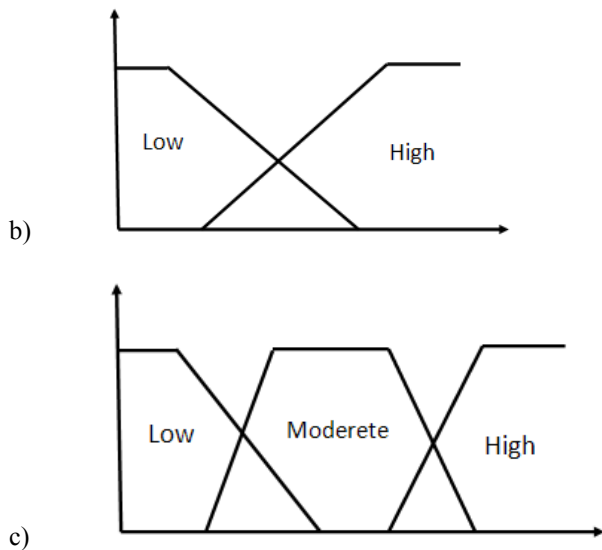


Figure 5 : Membership functions of the linguistic variables used for the estimation of the P

### C. Algorithmic description

Step 1 :

Definition of inputs and outputs:

In the Input there are two variables  $\sigma$  and NCN which characterizes each pixel and the output was the variable P.

Step 2 : Fuzzification or calculating membership degrees of input variables

- Calculating the membership degree of  $\sigma$  to the fuzzy set Small ( $d_{sig\_S}$ )
- Calculating the membership degree of fuzzy set  $\sigma$  to Great ( $d_{sig\_G}$ )
- Calculating the membership degree of the fuzzy NCN Small ( $d_{NCN\_S}$ )
- Calculating the membership degree of the fuzzy NCN Average ( $d_{NCN\_M}$ )
- Calculating the membership degree of the fuzzy NCN Great ( $d_{NCN\_G}$ )

Step 3 :

Using the basic rules already declared and using membership degrees computed already, we can calculate the membership degree of the output variable p to the fuzzy set Small ( $d_{p\_P}$ ) and its degree of membership to all blur Great ( $d_{p\_G}$ ) and the latter by replacing the logical AND of the basic rules by the MIN function and the logical OR of the MAX function.

$$d_{p\_S} \leftarrow \text{MAX} (\text{MIN} (d_{sig\_G}, d_{NCN\_G}), \text{MIN} (d_{sig\_G}, d_{NCN\_M}))$$

$$d_{p\_G} \leftarrow \text{MAX} (d_{sig\_S}, \text{MIN} (d_{sig\_G}, d_{NCN\_S}))$$

The intersection of two lines of equations  $y_1 = d_{p\_P}$ ,  $y_2 = d_{p\_G}$  and with the curve of the membership function p gives us two surfaces S1 and S2. These two surfaces are none other than a fuzzy value, therefore it must be transformed into real physical quantity.

Step 4 : Defuzzification

There are several methods to calculate the value of the output variable p, we chose the centroides method because it is the most accurate:

$$p \leftarrow (S1 * X1 + S2 * X2) / (S1 + S2)$$

where : X1 is the center of gravity of the surface S1

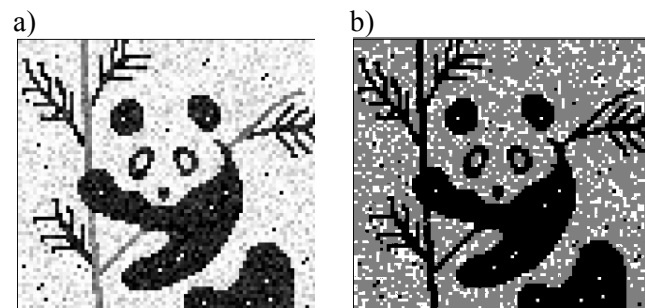
X2 is the center of gravity of the surface S2

## V. EXPERIMENT AND RESULTS

In this section, we present the results of the application of the ADEM algorithm. The performance of this algorithm is compared with the standard EM and DEM algorithm. The three techniques are tested on two images, the first is synthetic (panda image) and the second is the MRI cerebral image. The experiences are made in the same conditions (factor of fuzzification  $m = 2$  and mistake of convergence = 0.001). Concerning the spatial attributes, the ADEM algorithm uses the average calculated on a window of size 3x3 analysis.

The first test applies to the synthetic image named "Panda" (see Figure 6.a). This image contains 3 classes areas clearly identified. The image in Figure 6b shows the results of the application of the standard EM using as criterion classification gray level. This result shows the limits of this method on classification of noisy pixels. Compared to this, the application of DEM based on the average as attribute classification solves the problem noise as demonstrated in Figure 5.c but leads in cons part segmentation less accurate contours between regions and fine structures (Branches of the tree). The new variant that we proposed is given in figure 5.d. it confirms the good performance compared to DEM and standard EM.

The second test is about a MRI cerebral image. The segmentation consists in delimiting the three cerebral structures: gray matter (MG), white matter (WM) and the cerebrospinal fluid (CSF). It has particularities bound mainly to the noises and to the effect of the partial volume that one recovers when a pixel having a certain level of gray actually corresponds to a mixture of two or several cloths. This artifact exists mainly at the borders between the cloths MG and WM. The tests are achieved on the image of the corrupt face 6.a by 7% of Gaussian noise.



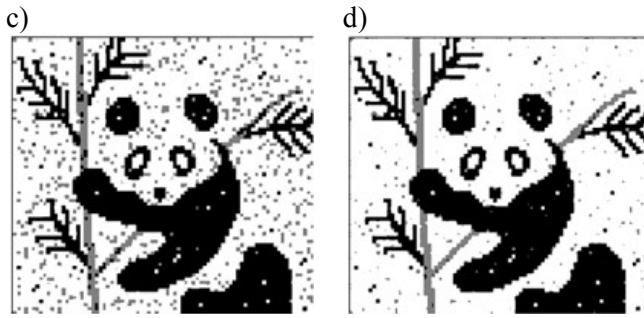


Figure 6 : a) original image b) Result of the standard EM c) Result of the DEM d) Result of the ADEM ( $\sigma=40$ )

Table I : Badly classified pixels number within contours and inside regions of MRI cerebral image

		EM	DEM	ADEM
Region	CSF	200	10	4
	GM	500	60	30
	WM	15	5	0
Contour	CSF	500	450	100
	GM	1120	1100	190
	WM	450	70	15

The application of the standard EM on MRI image gives some noisy classes and very ridden especially between the two classes MG and MB (figure 7.b). The use of the adaptive distance based EM on the spatial attribute implies a deterioration of the contours (figure 6.c). while the use of the ADEM ( $\sigma=40$ ) permits to reduce the noisy pixel significantly while getting the well identified regions and having continuous and near contours of the reality (figure 6.d). These good performances are confirmed well by the relative statistics to the pixels badly classified table I)..

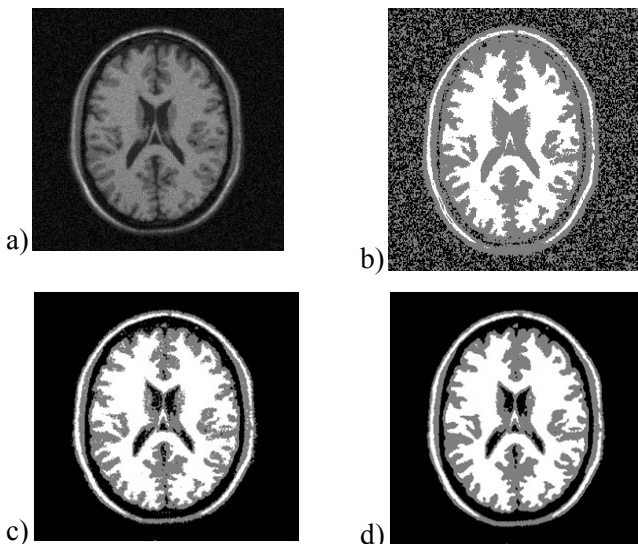


Figure 7 : a) original image b) Result of the standard EM c) Result of the DEM d) Result of the ADEM ( $\sigma=40$ )

On the other hand, experiments have shown that a value of sigma equal to 40 is quite sufficient for segmenting MRI images (figure 8).

## VI. CONCLUSION

A segmentation algorithm for grey level images was proposed. In the algorithm the image is considered as a mixture of multi-variate normal densities and segmentation is carried out by the ML estimation based on EM algorithm. It's a new approach that we proposed of pixels classification based on a dynamic and weighted similarity distance. The main idea is to use a manner conjoined and adaptive two attributes of classification: level of gray of the pixels and a spatial attribute (the local average). The adaptation must privilege one or the other of these attributes according to the spatial configuration of the pixel to classify. The originality of our approach resides in the manner with which is calculated these weights and that take as a basis on a mechanism of fuzzy inference. The new distance that we proposed thus permitted to get a new variant of the EM method that is adapted more to the segmentation of images. It has been confirmed by the tests achieved on a cerebral MRI image.

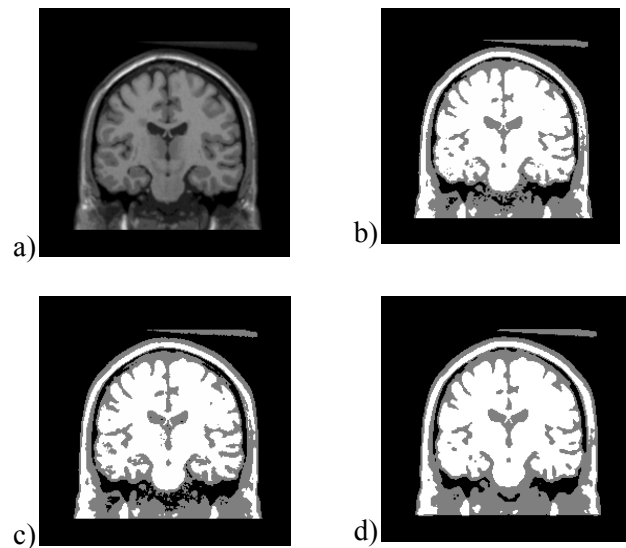


Figure 8 : a) original image b) Result of the ADEM ( $\sigma=15$ ) c) Result of the ADEM ( $\sigma=40$ ) d) Result of the ADEM ( $\sigma=70$ )

## REFERENCES

- [1] Mohamed Walid Ayeche, Karim El Kalti, Béchir el Ayebe: Image Segmentation Based on Adaptive Fuzzy-C-Means Clustering. ICPR 2010: 2306-2309
- [2] L.Xu, M.I.Jordan, On Convergence Properties of the EM Algorithm for Gaussian Mixture, *Neural Computation*, 8, 1996, pp. 129-151.
- [3] J-C. Bezdek, Pattern recognition with fuzzy objective functions algorithms, *Plenum Press*, New York, 1981.
- [4] T-N. Tran, R. Wehrens, L-M-C. Buydens, Clustering multispectral images: a tutorial, *Chemometric and intelligent laboratory systems*, pp. 211-218, 2004.
- [5] P-W-T. Krooshof, T-N. Tran, G-J. Postma, Effects of including spatial information in clustering of multivariate image data, *Trends in Analytical Chemistry*, Vol. 25, pp. 1067-1080, 2006.
- [6] K-S. Chuang, Fuzzy c means clustering with spatial information for image segmentation, *Elsevier Science*, Vol.30, pp. 9-15, 2006.

- [7] Y-A. Toliás, S-M. Panas, On applying spatial constraints in fuzzy image clustering using a fuzzy rule based system, *IEEE Signal Processing Letters*, Vol.5, pp. 245-247, 1998.
- [8] GU D. B, SUN J.-X EM image segmentation algorithm based on an inhomogeneous hidden MRF model IEEE proceedings, Vision, image and signal processing 2005, vol. 152, n°2, pp. 184-190.
- [9] S-C. Chen, D-Q. Zhang, Robust image segmentation using FCM with spatial constraints based in new kernel-induced distance measure, *IEEE Trans. Systems Man Cybernetics*, Vol. 34, pp. 1907-1916, 2004.
- [10] J. Kawa, E. Pietka, Image clustering with median and myriad spatial constraint enhanced FCM, *Springer-Verlag Berlin Heidelberg*, Vol. 30, pp. 211-218, 2005.
- [11] D-J. Hemanth, D. Selvathi, J. Anitha, Effective fuzzy clustering algorithm for abnormal MR brain image segmentation, *IEEE International Advance Computing Conference*, pp. 609-614, March 2009.
- [12] S. Shen, Preprocessing and segmentation of brain magnetic resonance images, *Proc of the 4th Annual IEEE Conference on Information Technology Application in Biomedicine*, UK, pp.149-152, 2003.
- [13] J. Wang, L. Dou, N. Che, D. Liu, B. Zhang, J. Kong, Local based fuzzy clustering for segmentation of MR brain images, *IEEE International Conference on Bioinformatics and Bioengineering*, pp. 1-5, October 2008.
- [15] L. Zadeh, Fuzzy sets, *Information and Control*, Vol. 8, pp. 338-353, 1965. Archambeau C., Vrins F., Verleysen M. (2001), Flexible and Robust Bayesian Classification by finite Mixture Models., Proceedings of the ESANN'2004 congress.
- [16] S. Sanjay-Gopal and T. J. Hebert, Bayesian Pixel Classification Using Spatially Variant Finite Mixtures and the Generalized EM algorithm", *IEEE trans. Image Processing*, Vol. 7, No.7, pp1014-1028, 1998.
- [17] R. A. Redner, H. F. Walker, Mixture densities, maximum vraisemblance and the EM algorithm, *SIAM Review*, 26, pp195-239, 1984.
- [18] C. Biernacki, R. Mohr. Indexation et appariement d'images par modèle de mélange gaussien des couleurs. Institut National de Recherche en Informatique et en Automatique, Rapport de recherche n\_3600\_Janvier 2000.
- [19] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from in- complete data via the em algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1-38, November 1977.
- [20] Zhi Kai Huang, Unsupervised image segmentation using EM algorithm by histogram, *Lecture Notes in Computer Science*, 2007, Volume 4681/2007, 1275-1282, DOI: 10.1007/978-3-540-74171-8\_130

#### AUTHORS PROFILE

**Dr Mohamed Ali Mahjoub** is an assistant professor in computer science at the Preparatory Institute of Engineering of Monastir. His research interests concern the areas of Bayesian Network, Computer Vision, Pattern Recognition, HMM, and image retrieval. His main results have been published in international journals and conferences.

**Dr Karim kalti** is an assistant professor in the department of computer science at the faculty of science of Monastir (Tunisia). His research interests concern the areas of Image processing, Pattern Recognition, Medical imaging and Data Retrieval. His main results have been published in international journals and conferences.

# Modeling of neural image compression using GA and BP: a comparative approach

G.G Rajput, Vrinda Shivashetty  
Department of Computer Science  
Gulbarga University  
Gulbarga,India

Manoj Kumar singh  
Manuro Tech Research,  
Bangalore,India

**Abstract**— It is well known that the classic image compression techniques such as JPEG and MPEG have serious limitations at high compression rate; the decompressed image gets really fuzzy or indistinguishable. To overcome problems associated with conventional methods, artificial neural networks based method can be used. Genetic algorithm is a very powerful method for solving real life problems and this has been proven by applying to number of different applications. There is lots of interest to involve the GA with ANN for various reasons at various levels. Trapping in the local minima is one of the well-known problems of gradient decent based learning in ANN. The problem can be addressed using GA algorithm. But no work has been done to evaluate the performance of both learning methods from the image compression point of view. In this paper, we investigate the performance of ANN with GA in the application of image compression for obtaining optimal set of weights. Direct method of compression has been applied with neural network to get the additive advantage for security of compressed data. The experiments reveal that the standard BP with proper parameters provide good generalize capability for compression and is much faster compared to earlier work in the literature, based on cumulative distribution function. Further, the results obtained shows that general concept about GA, it performs better over gradient decent based learning, is not applicable for image compression.

**Keywords**- Image compression; genetic algorithm; neural network; back propagation.

## I. INTRODUCTION

Artificial neural network (ANN) technique has been used successfully for image compression with various ways [10, 11,12,13,21]. A detail survey of about how ANN can be applied for compression purpose is reported in [1,14,15,16,17]. Broadly, two different categories for improving the compression methods and performance have been suggested. Firstly, develop the existence method of compression by use of ANN technology so that improvement in the design of existing method can be achieved. Secondly, apply neural network to develop the compression scheme itself, so that new methods can be developed and further research and possibilities can be explored for future. Statistical approaches are applied in integration with neural network for enhancement of compression performance. In [2,18], principal component analysis (PCA) is applied for this purpose. PCA is one of the famous statistical methods which eliminates the correlation between different data components and consequently decrease the size of data. In classical method, covariance matrix of input

data is used for extracting singular values and vectors. Neural networks are used for extracting principal value components in order to compress image data. First, different principal component analysis neural networks is presented and then a nonlinear PCA neural network is used which provides better results as shown in simulation results. Speed is one of the fundamental issues that always appear in the application of image compression. In [4,19,20,22], the problems associated with neural network for compression is discussed. Authors have given the concept of reduction of original feature space, which allows us to eliminate the image redundancy and accordingly leads to their compression. Two variants of neural network they have suggested: two layers neural network with self learning algorithm based on the weighted information criterion and auto-associative four layers feed forward network. In [5,23,24,25], a constructive One-Hidden-Layer feed forward Neural Network (OHL-FNN) architecture has been applied for image compression. The BPNN has taken as the simplest architecture of ANN that has been developed for image compression but its drawback is very slow convergence.

## II. FEED FORWARD ARTIFICIAL NEURAL NETWORKS

In feed forward architecture having multilayer perceptrons, the basic computational unit, often referred to as a “neuron,” consists of a set of “synaptic” weights, one for every input, plus a bias weight, a summer, and a nonlinear function referred to as the activation function . Each unit computes the weighted sum of the inputs plus the bias weight and passes this sum through the activation function to calculate the output value as  $y_j = f(\sum_i w_{ji}x_i + \theta_j)$  ,where  $x_i$  is the  $i$ th input value for the neuron and  $w_{ji}$  is the corresponding synaptic weight. The activation function  $f(\bullet)$  maps the potentially infinite range of the weighted sum to a limited, finite range. A common

activation function is a sigmoid defined as  $f(v) = \frac{1}{1 + e^{-v}}$

In a multilayer configuration, the outputs of the units in one layer form the inputs to the next layer. The inputs to the first layer are considered as the network inputs, and outputs of the last layer are the network outputs. The weights of the network are usually computed by training the network.

### A. Evolution of weights in neural network using GA

In recent times much research has been undertaken in the combination of two important and distinct areas: genetic algorithm and neural networks. Genetic algorithms attempt to



apply evolutionary concept to the field of problem solving, notably function optimization and have proven to be valuable in searching large, complex problem spaces. Neural networks are highly simplified models of the working of brain. These consist of a combination of neurons and synaptic connections, which are capable of passing data through multiple layers. The end result is a system which is capable of pattern and classification. In the past, algorithm such as back propagation have been developed which refine one of the principle components of the neural networks: connection weights. The system has worked well, but is prone to becoming trapped in local optima and is incapable of optimization where problems lie in a multi-model or non-differentiable problem space. Genetic algorithms and neural networks can be combined such that populations of neural networks compete with each other in a Darwinian 'survival of the fittest' setting. Networks which are deemed to fit are combined and passed onto the next generation producing an increasingly fit population, so that after a number of iterations for an optimized neural network can be obtained without resorting to a design by hand method. The primary motivation for using evolutionary technique to establish the weighting values rather than traditional gradient decent techniques such as back propagation lies in the inherent problems associated with gradient descent approaches.

The evolution of neural networks can be classified according to the goals behind such evolution. Some schemes have proposed by introducing the evolution of weights with the fixed architecture. Other level of evolution where improvement can be expected is in the architecture is the transfer function [yao].

#### B. Chromosome, Crossover & mutation operation to generate the offspring

Initially, a population of chromosomes created contains a uniformly distributed random number. Chromosomes directly representing the weights of neural network are shown in fig.2. Hence, there is no need of any encoding mechanism in result. Crossover here can be defined as node crossover. From picked up two parents for generating off springs, any one active node from the set of hidden and output layer, pick up randomly with equal probability. This node consider as a node of crossover. Values of all incoming weights for that particular node are exchanged with available other parent. Mutation can also be considered as node mutation, where in an offspring, all incoming weights for a randomly picked up active node added with Gaussian distributed random numbers. These two processes are shown in fig.3and fig.4, respectively.

#### C. Algorithm for weights evolution by GA in ANN

The following steps are performed for determining the optimal value of weights.

- (i)A population of  $\mu$  parent solution  $X_i$ ,  $i=1, \dots, \mu$ , is initialized over a region  $M \in R^n$ .
- (ii)Two parents are selected randomly with uniform distribution from population of  $\mu$  parents, and two offspring will created by crossover operator as shown in Fig.2.
- (iii)Mutation on newly generated offspring will be applied as shown in Fig .3.

- (iv)Repeat step (ii) until population of offspring  $\mu_o$  equal to  $\mu$ , otherwise move to step (v).
- (v)Each parent solution  $X_i$ ,  $i=1, \dots, \mu$  and offspring  $X_o$ ,  $o=1, \dots, \mu$ , is scored in light of the objective function  $f(X)$ .
- (vi)A mixture population  $X_m$ ,  $m = 1, \dots, 2\mu$  contains both parent population and offspring population created. This mixture population randomly shuffled so that parents and offspring could mix up properly.
- (vii)Each solution from  $X_m$ ,  $m = 1, \dots, 2\mu$  is evaluated against 10% of  $\mu$  other randomly chosen solutions from the mix population  $X_m$ . For each comparison a 'win' is assigned if the solution's score is less than or equal to that of its opponent.
- (viii)The  $\mu$  solutions with the greatest number of wins are retained to be parents of the next generation.
- (ix)If the difference in the best chromosome for N number of continuous generation are less than the defined threshold value k, terminate the process and the last generation best chromosome is the optimal weights, otherwise proceed to step (ii).

#### D. Weight optimization with back propagation algorithm.

Back propagation algorithm is a supervised learning algorithm which performs a gradient descent on a squared error energy surface to arrive at a minimum. The key to the use of this method on a multilayer perceptrons is the calculation of error values for the hidden units by propagating the error backwards through the network. The local gradient for the jth unit, in the output layer is calculated as (assuming a logistic function for the sigmoid nonlinearity)

$$\delta_j = y_j(1 - y_j)(d_j - y_j) \quad (1)$$

where  $y_j$  is the output of unit j and  $d_j$  is the desired response for the unit. For a hidden layer, the local gradient for neuron j is calculated as

$$\delta_j = y_j(1 - y_j) \sum_k \delta_k w_{jk} \quad (2)$$

where the summation k is taken over all the neurons in the next layer to which the neuron j serves as input. Once the local gradients are calculated, each weight  $w_{ji}$  is then modified according to the delta rule

$$w_{ji}(t + 1) = w_{ji}(t) + \eta \delta_j(t) y_i(t) \quad (3)$$

Where  $\eta$  a learning-rate parameter and t is time. Frequently modification is used that incorporates a momentum term that helps to accelerate the learning process

$$w_{ji}(t + 1) = w_{ji}(t) + \eta \delta_j(t) y_i(t) + \alpha [w_{ji}(t) - w_{ji}(t - 1)] \quad (4)$$

Where  $\alpha$  is a momentum term lying in the range  $0 < \alpha < 1$ .

### III. IMAGE COMPRESSION STRUCTURE USING PERCEPTRONS NEURAL NETWORK

The structure to compress images is the three layer perceptrons, depicted in Fig.11. In order to use structure, the input image is divided into blocks with pixels equal to the same

number of neurons in input layer, say N. It means that these blocks should be of order  $\sqrt{N} \times \sqrt{N}$  (in this paper this size is 8\*8) so that they can be expressed in N dimensional vector and fed into the input layer. The hidden layer in this structure is the compressed image which maps N pixels to K ( $K < N$ ) and finally the reconstructed image from compressed one is derived with the same number of pixels/neurons as the input. In this structure the input weights to the hidden layer are a transform matrix which scales the input vector of N-dimensional into a narrow channel of k-dimensional. Similarly, the weights of hidden to output layer are a transform matrix which scales the narrow vector of K-dimensional into a channel of N-dimensional. The input gray-level pixel values are normalized to the range [0, 1]. The reason for using normalized pixel values is due to the fact that neural networks can operate more effectively when both their inputs and outputs are limited to a range of [0, 1]. Learning is applied to train the architecture. All patterns in the input blocks of training set are also fed to output layer as the target. Once training is completed with proper performance, final weights are having the capability to map the input value of pixels into approximate same value at the output. Compression process is defined by taking the half of the trained architecture which has been utilize at the time of training ,i.e. input layer along with the hidden layer as shown in Fig.12. Remaining half of the trained architecture i.e. hidden layer along with output layer is utilized to setup the decompression, as shown in Fig.13.

#### IV. PERFORMANCE PARAMETERS

Evaluation criteria used for comparison in this paper, is compression ratio (CR) and the peak signal-to-noise ratio (PSNR). For an Image with R rows and C columns, PSNR is defined as follows:

$$PSNR = 10 \log_{10} \left( \frac{255^2}{\frac{1}{RC} \sum_{i=1}^R \sum_{j=1}^C (X_{ij} - \bar{X}_{ij})^2} \right)$$

Compression ratio (CR) which is a criterion in compression problems is defined as the number of bits in original image to number of bits in the compressed image. This criterion in the sense of using neural net structure is defined as follow:

$$CR = \frac{N \cdot B_I}{K \cdot B_H}$$

In this equation N and K are the neurons/pixels available in the input and hidden layer respectively and  $B_I$  and  $B_H$  are the number of bits needed to encode outputs of input and hidden layer. If the number of bits needed to encode the input layer and the number of bits needed to encode the hidden layer be the same, the compression ratio will be the number of neurons in the input layer to hidden layer. As an example for the gray level images which are 8 bits long if we encode the compressed image with the same number of bits in a block of 8x8 and the network of with 16 neurons at the hidden layer, the compression ratio will be 4:1. And for the same network with

floating point used to encode the hidden layer, the compression ratio will be 1:1 which means no compression.

To verify the developed design for evolving the weights of neural network two different experiments are considered as explained in the following section. This will give the confidence to apply the developed method for image compression.

#### V. PATTERN RECOGNITION AND THE XOR PROBLEM

The pattern recognition problem consists of designing algorithms that automatically classify feature vectors associated with specific patterns as belonging to one of a finite number of classes. A benchmark problem in the design of pattern recognition systems is the Boolean exclusive OR (XOR) problem. The standard XOR problem is shown in figure below. Here, the diagonally opposite corner-pairs of the unit square form two classes, A and B (or NOT A). From the figure, it is clear that it is not possible to draw a single straight line which will separate the two classes. This observation is crucial in explaining the inability of a single-layer perceptrons to solve this problem .This problem can be solved using multi-layer perceptrons (MLPs), or by using more elaborate single-layer ANNs.

##### A. Performance of GA with ANN for weight evolution over XOR problem.

A feed forward architecture of 2-2-1 designed and weights has evolved by above defined method of GA. Population size taken as 20 and condition of terminating criteria is, if the best chromosome error in 50 continuous generation is less than 0.00001.

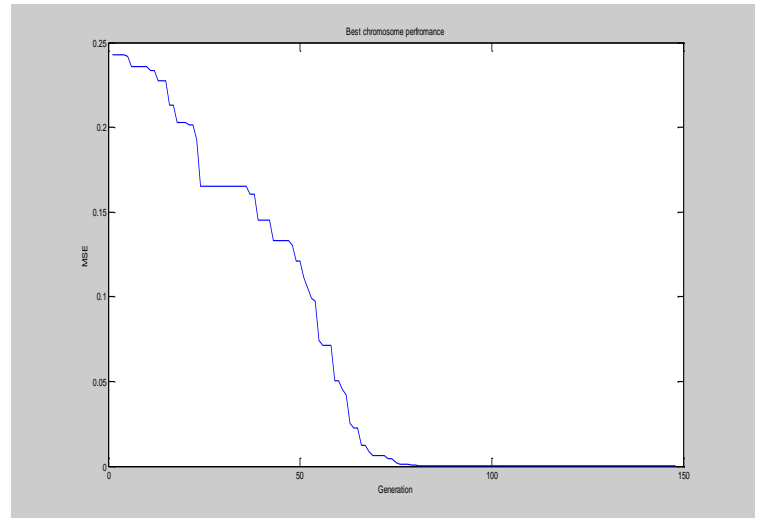


Figure.1. Error performance with generation for best chromosome.

Expected output	0	1	1	0
Output GA+ANN	0.000	1.000	1.000	0.000
Total generation	148			
Mean square error at last generation	6.61378207973114e-009			

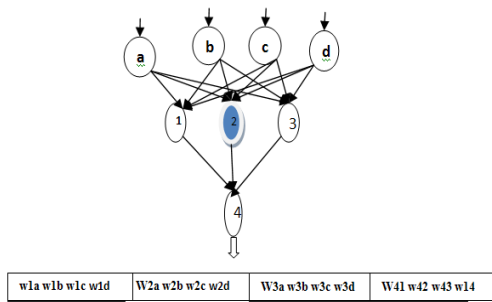


Figure 2. Chromosome development

(b) mapped output:

161	162	162	161	162	157	164	162
162	163	162	162	161	157	164	162
164	162	163	160	163	158	162	160
163	161	163	163	161	157	163	161
163	162	163	161	161	158	162	160
164	163	160	155	161	159	160	160
161	160	163	157	161	162	160	155
160	158	155	158	158	159	157	159

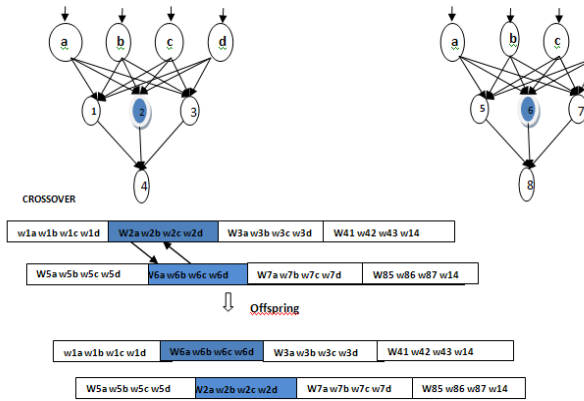


Figure 3. Crossover operation

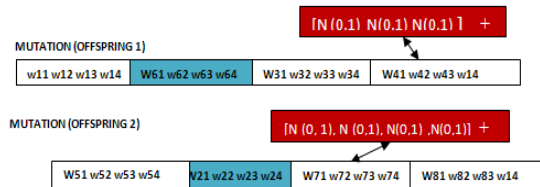


Figure 4. Mutation operation

**B. Performance of GA with ANN for weight evolution over one block of image**

To see the performance of GA based weights optimization design over real image, a block of image containing 8\*8 pixels has been taken. Population of 50 chromosomes is taken with 16 hidden nodes in ANN architecture. To see the proper convergence a large value of generation 2000 has defined for termination. Performance of experiment has shown in table (2). Mapping result is shown below. Convergence of learning for best chromosome is shown in Fig. 5.

(a) Input block:

162	162	162	161	162	157	163	161
162	162	162	161	162	157	163	161
162	162	162	161	162	157	163	161
162	162	162	161	162	157	163	161
164	164	158	155	161	159	159	160
160	160	163	158	160	162	159	156
159	159	155	157	158	159	156	157

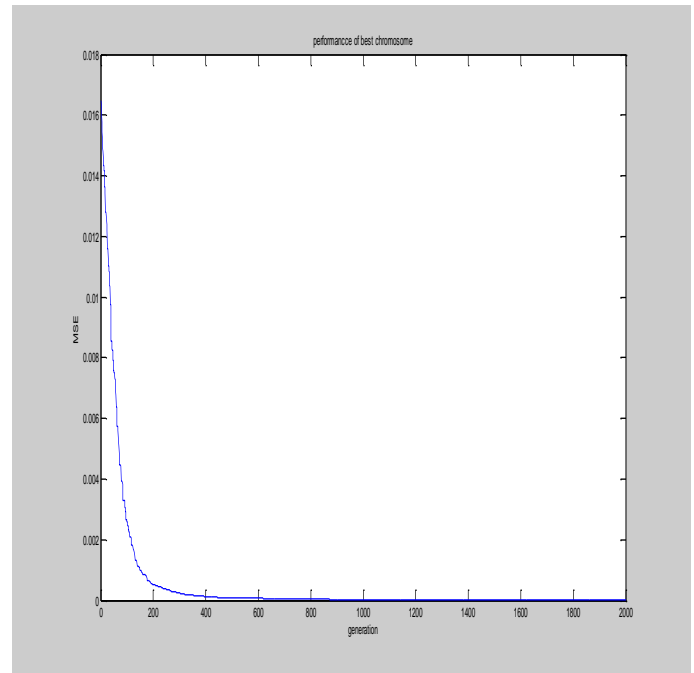


Figure 5. Error performance

With the above two different experiments, performance given by GA based weights optimization seems very impressive and results show the design of GA for neural learning is working better. This has given enough confidence to deploy the GA for image compression.

**VI. IMAGE COMPRESSION WITH GA AND ANN**

A population of 50 chromosomes is applied for evolving the weights up to 200 generations. Compression ratio for this experiment defined as 4:1. Performance plot for compression is shown in Fig.6 and in Fig.7 .Decompression result of Lena image is shown in Fig.8. Table (3) shows the parameter values. From the result it is very clear that the process is taking very long time for completing the cycle of generation. Even convergence is not proper and the result of compression performance is very poor and cannot be consider for practical purpose.

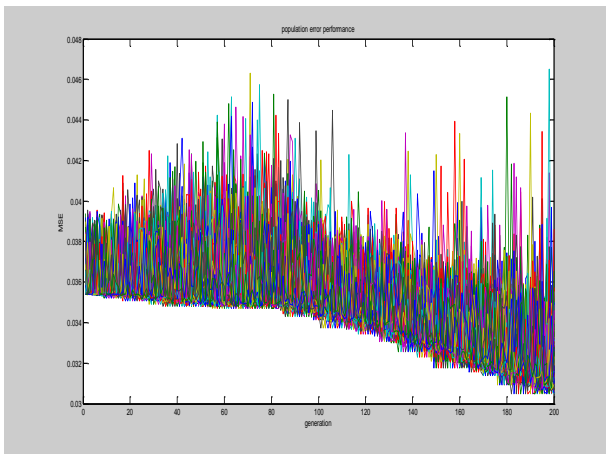


Figure 6. Population Error plot

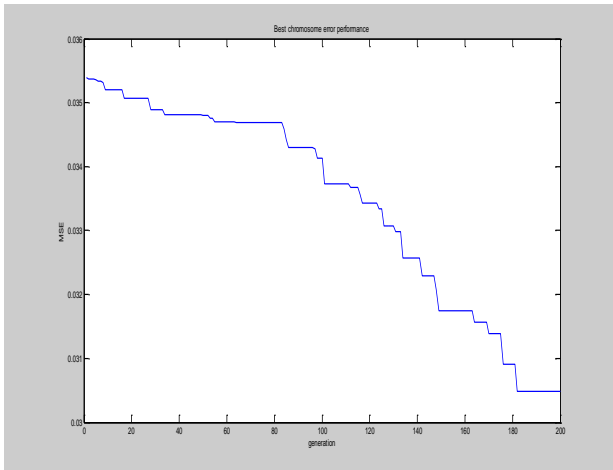


Figure 7. Best chromosome Error plot

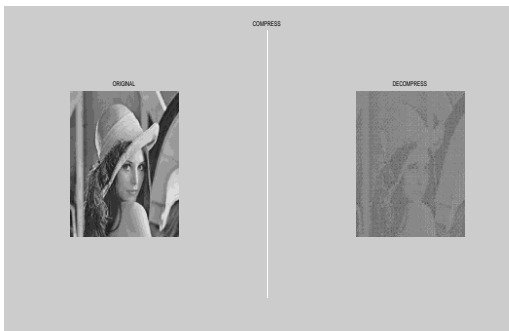
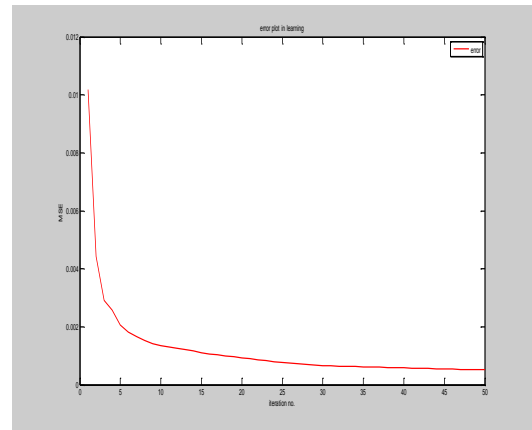


Figure8: Decompression by GA with ANN

## VII. IMAGE COMPRESSION USING ANN AND STANDARD BACK PROPAGATION ALGORITHM

Standard back propagation is one of the most widely used learning algorithms for finding the optimal set of weights in learning. A single image “Lena” is taken as training data. The error curve of learning is shown in Fig.9 for below defined set of parameters. Further, different images are tested to generalize the capability of compression. The processes repeated for two different compression ratio by changing the number of hidden

nodes in neural architecture. The performance observed during the time of training and testing is shown in table 4, table 5 for compression ratio 4:1 and in table 6, table 7 or 8:1, respectively. Table 8 given the comparison with [9]



**Parameter setting for back propagation learning:**  
Initial random weights value taken from uniform distribution in range of  $[-0.0005 +0.0005]$ . Learning rate: 0.1 ; Momentum constant: 0.5; Bias applied to hidden and output layer nodes with fixed input as (+1). Allowed number of iteration : 50

Figure 9. Error plot in back propagation

Compression ratio: 4:1



Figure 10. Performance by gradient descent

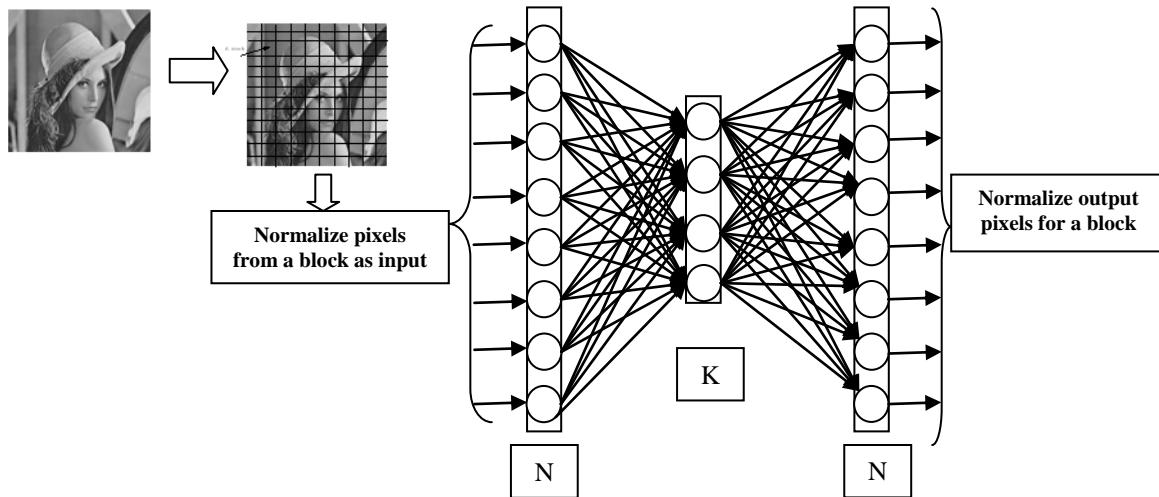


Figure 11. Architecture of neural network at the time of training

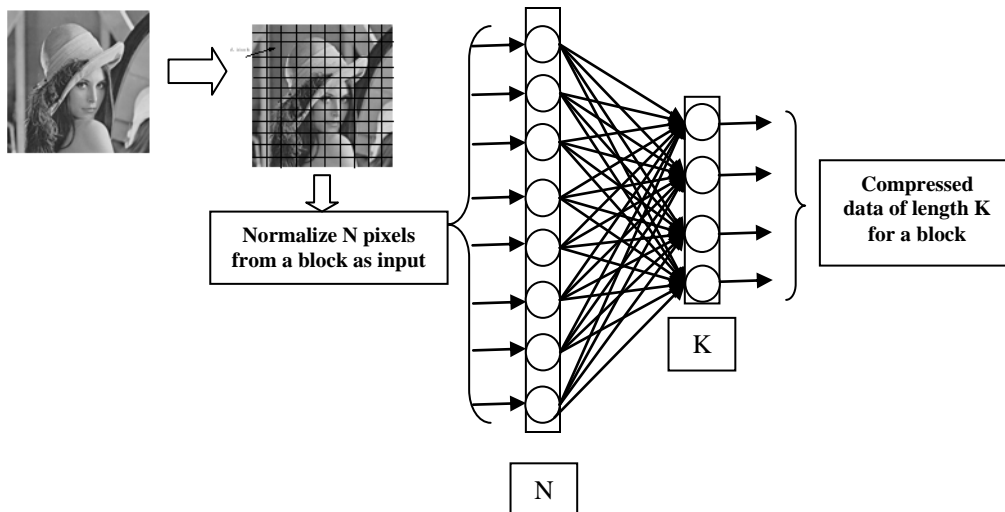


Figure 12. Compression module

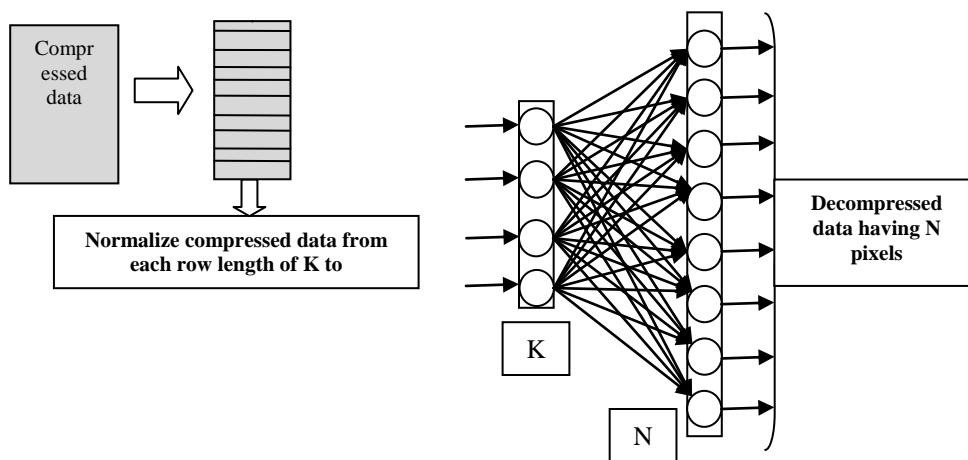


Figure 13. Decompress module

Table 3

Training image	Image format	Learning Time	No. of epoch.	Error	CR	PSNR(db)	Comp. Time(sec.)	Decom. Time(sec.)
Lena	BMP	1.771	200	0.035	4:1	15.15	0.0692	0.1665

Table 4

Training image	Image format	Learning Time	No. of epoch.	Error	CR	PSNR(db)	Comp. Time(sec.)	Decom. Time(sec.)
Lena	BMP	66.69	50	0.005	4:1	32.6017	0.0686	0.1881

Test image	Image format	CR	PSNR(db)	Comp. Time	Decom. Time
Elaine	TIF	4:1	31.0652	0.0691	0.1754
Boat	TIF	4:1	28.5801	0.0700	0.1844
Pepper	TIF	4:1	29.5953	0.0696	0.1875

Table 5

Training image	Image format	Learning Time	No. of epoch.	Error	CR	PSNR(db)	Comp. Time(sec.)	Decom. Time(sec.)
Pepper	TIF	61.32	50	0.005	4:1	31.4338	0.0692	0.1845

Test image	Image format	CR	PSNR(db)	Comp. Time	Decom. Time
Elaine	TIF	4:1	30.3149	0.0689	0.1834
Boat	TIF	4:1	28.4153	0.0693	0.1864
Pepper	BMP	4:1	31.4238	0.0696	0.1928

Table 6

Training image	Image format	Learning Time	No. of epoch.	Error	CR	PSNR(db)	Comp. Time(sec.)	Decom. Time(sec.)
Lena	BMP	43.03	50	0.0010	8:1	29.7739	0.0665	0.1799

Test image	Image format	CR	PSNR(db)	Comp. Time	Decom. Time
Elaine	TIF	8:1	29.4417	0.0336	0.1833
Boat	TIF	8:1	26.0333	0.0578	0.1790
Pepper	TIF	8:1	27.3020	0.0576	0.1821

Table 7

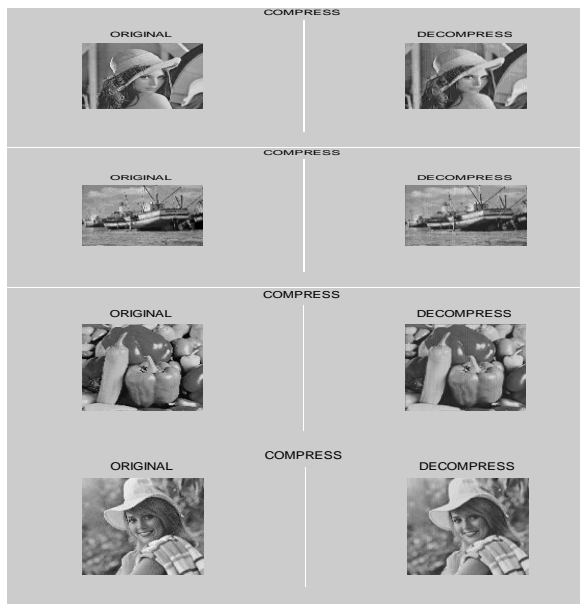
Training image	Image format	Learning Time	No. of epoch.	Error	CR	PSNR(db)	Comp. Time(sec.)	Decom. Time(sec.)
Pepper	TIF	47.4454	50	0.0012	8:1	28.6450	0.0675	0.1857

Test image	Image format	CR	PSNR(db)	Comp. Time	Decom. Time
Elaine	TIF	8:1	28.2311	0.0581	0.1795
Boat	TIF	8:1	26.2294	0.0583	0.1855
Lena	BMP	8:1	28.5439	0.0570	0.1871

Table 8

[Durai & Sarao]				Proposed method	
Image	CR	PSNR(db)	Time(sec)	PSNR(db Time(sec))	
Lena	4:1	28.91	182	<b>32.60</b>	66.69
Pepper	4:1	29.04	188	<b>31.43</b>	61.32
Boat	4:1	29.12	178	<b>29.68</b>	64.75

Compression ratio: 8:1



Figure(14). Performance by gradient decent

## VIII. CONCLUSION

There is increasing demand of image compression based processing in various applications. Numbers of methods are available and up to some extent they are generating satisfactory results. However, with changing technology there is still a wider scope to work in this area. New techniques may be proposed which could either replace or provide the support of existing methods. Compression techniques based on neural network have good scope in both ways. The general perception about GA is that, it performs better over back propagation based learning has proven wrong in the present work. Even though same algorithm of GA performs very well for XOR classification and mapping of small data, for image compression. GA based learning for neural network is suffering from curse of very slow convergence and poor quality of compression. Whereas, back propagation has shown high converging speed with good quality of compression. The method is also applicable to a wide range of different image file types. Security of compressed data is another inherent advantage available if compression happen by neural network in direct approach (i.e. until weights are not available it is nearly impossible to find the contents of the image from compressed data).

## ACKNOWLEDGMENT

I would like to convey my thanks to my guide Dr. G.G. Rajput for providing helpful guidance and encouragement. It is my pleasure to express deep sense of gratitude and profound to Mr. Manoj Kumar Singh for guiding & giving me a strength of a solution.

I would like to thank my family and friends for the splendid support and encouragement, without whom this paper would have been just a dream.

Last but not the least I extend my sincere thanks to all the teaching and non-teaching staff of the computer science dept. Gulbarga University, Gulbarga.

## REFERENCES

- [1] J.Jiang, "Image compression with neural networks :A survey ".Signal Processing: Image Communication 14 (1999) ,737-760
- [2] Moghadam, R.A. Eslamifar, M. "Image compression using linear and nonlinear principal component neural networks" .Digital Information and Web Technologies, 2009. ICADIWT '09 . Second International Conference on The London, Issue Date: 4-6 Aug. 2009 .pp: 855 – 860.
- [3] Palocio, Crespo, Novelle, "image /video compression with artificial neural network".Springer-Verlag, IWANN 2009, part ii, LNCS 5518, pp:330-337.2009
- [4]Bodyanskiy, grimm, mashtalir, vinariski, "fast training of neural network for image compression".Springer-Verlag ,ICDM 2010, LNAI 6171, PP:165-173,2010.
- [5] Liying Ma, Khashayar Khorasani, " Adaptive Constructive Neural Networks Using Hermite Polynomials for Image Compression " ,Advances in Neural Networks, springer, Volume 3497/2005, 713-722, DOI: 10.1007/11427445\_115
- [6] Dipta Pratim Dutta, Samrat Deb Choudhury, Md Anwar Hussain, Swanirbhar Majumder, "Digital Image Compression Using Neural Networks," act, pp.116-120, 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies, 2009
- [7] B. Karlik, "Medical Image Compression by Using Vector Quantization Neural Network", ACAD Sciences press in Computer Science, vol. 16, no. 4, 2006 pp., 341-348.
- [8] Y. Zhou., C. Zhang, and Z. Zhang, "Improved Variance-Based Fractal Image Compression Using Neural Networks", Lecture Notes in Computer Science, Springer-Verlag, vol. 3972, 2006, pp. 575-580
- [9] Durai S.A. and E.A. Saro., Image compression with back-propagation neural network using cumulative distribution function. World Acad. Sci. eng. Technol:60-64, 2006.
- [10] Rao, P.V. Madhusudana, S. Nachiketh, S.S. Keerthi, K. "image compression using artificial neural network".EEE,ICMLC 2010, PP:121-124.
- [11] Dutta, D.P.; Choudhury, S.D.; Hussain, M.A.; Majumder, S.; "Digital image compression using neural network".IEEE,international Conference on Advances in Computing, Control,Telecommunication .....Technologies, 2009. ACT '09.
- [12] N.M.Rahim, T.Yahagi, "Image Compression by new sub-image bloc Classification techniques using Neural Networks", IEICE Trans. On Fundamentals, Vol. E83-A, No.10, pp 2040-2043, 2000
- [13] M. S. Rahim, "Image compression by new sub- image block Classification techniques using neural network. IEICE Trans. on Fundamentals of Electronics, Communications, and Computer Sciences, E83-A (10), (2000),pp. 2040- 2043.
- [14] D. Anthony, E. Hines, D. Taylor, and J. Barham, "A study of data compression using neural networks and principal component analysis," in *Colloquium on Biomedical Applications of Digital Signal Processing*, 1989, pp. 1-5.
- [15] G. L. Sicuranzi, G. Ramponi, and S. Marsi, "Artificial neural network for image compression," *Electronics Letters*, vol. 26, no. 7, pp. 477-479, March 29 1990.
- [16] M.Egmont-Petersen, D.de.Ridder, Handels, "Image Processing with Neural Networks – a review", Pattern Recognition 35(2002) 2279-2301, www.elsevier.com/locate/patcog
- [17] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT Press, Cambridge, MA, 1995.
- [18] K.I. Diamantaras, S.Y. Kung, *Principal Component Neural Networks: theory and applications* , John Wiley & Sons, 1996
- [19] S. Rizvi, L. Wang, " Nonlinear vector prediction using feedforward Neural network", IEEE Trans. Image Process., 6(10), (1997), pp. 1431-1436.
- [20] A. K. Ashikur, C. Mofizur, "A new approach for compressing images Using neural network", CIMCA Proceeding, (February 2003), Vienna – Austria.
- [21] M. Egmont-Ptersen, D. de Ridder, H. Handels, "Image processing with Neural networks- a review", Pattern Recognition , 35 (2002) pp. 2279- 2301.

- [22] R. D. Dony, S. Haykin, "Neural network approaches to image compression", Proc. IEEE 83 (2) (February 1995) pp. 288- 303.
- [23] B.Verma, B.Blumenstin and S. Kulkarni, Griggith University, Australia, "A new Compression technique using an artificial neural network".
- [24] M.A.Otair, W.A.Salameh (Jordan), "An Improved Back-Propagation Neural Networks using a Modified Non-linear function", The IASTED Conference on Artificial Intelligence and Applications, Innsbruck, Austria, February 2006.
- [25] T. M. Nabhan and A. Y. Zomaya, "Toward generating neural network structures for function approximation," *Neural Networks*, vol. 7, no. 1,

pp. 89–99, 1994.

#### AUTHORS' PROFILE



Vrinda Shivashetty pursuing Ph.D in image compression from the University of Gulbarga University, Gulbarga, Karnataka. Field of interest includes the intelligent image processing, evolutionary computation.

G.G Rajput presently working under the Dept. of studies & Research in computer science, Gulbarga University, Gulbarga.

Manoj kr. Singh is having background of R&D in Nanotechnology, Evolutionary computation, Neural network etc. Currently he is holding the Post of director in Manuro Tech. Research. He is also actively associated with Industry as a consultant & guiding number of research scholars.



# Clustering and Bayesian network for image of faces classification

Khelifia Jayech<sup>1</sup>

<sup>1</sup>SID Laboratory, National Engineering School of Sousse  
Technology Park 4054 Sahloul, Sousse  
Tunisia

Mohamed Ali Mahjoub<sup>2</sup>

<sup>2</sup>Preparatory Institute of Engineer of Monastir  
Street Ibn Eljazzar Monastir  
Tunisia

**Abstract**— In a content based image classification system, target images are sorted by feature similarities with respect to the query (CBIR). In this paper, we propose to use new approach combining distance tangent, k-means algorithm and Bayesian network for image classification. First, we use the technique of tangent distance to calculate several tangent spaces representing the same image. The objective is to reduce the error in the classification phase. Second, we cut the image in a whole of blocks. For each block, we compute a vector of descriptors. Then, we use K-means to cluster the low-level features including color and texture information to build a vector of labels for each image. Finally, we apply five variants of Bayesian networks classifiers (Naïve Bayes, Global Tree Augmented Naïve Bayes (GTAN), Global Forest Augmented Naïve Bayes (GFAN), Tree Augmented Naïve Bayes for each class (TAN), and Forest Augmented Naïve Bayes for each class (FAN) to classify the image of faces using the vector of labels. In order to validate the feasibility and effectively, we compare the results of GFAN to FAN and to the others classifiers (NB, GTAN, TAN). The results demonstrate FAN outperforms than GFAN, NB, GTAN and TAN in the overall classification accuracy.

**Keywords**- face recognition; clustering; Bayesian network; Naïve Bayes; TAN; FAN.

## I. INTRODUCTION

Classification is a basic task in data mining and pattern recognition that requires the construction of a classifier, that is, a function that assigns a class label to instances described by a set of features (or attributes) [10]. Learning accurate classifiers from pre-classified data has been a very active research topic in the machine learning. In recent years, numerous approaches have been proposed in face recognition for classification such as Fuzzy sets, Rough sets, Hidden Markov Model (HMM), Neural Network, Support Vector Machine and Genetic Algorithms, Ant Behavior Simulation, Case-based Reasoning, Bayesian Networks etc. Much of the related work on image classification for indexing, classifying and retrieval has focused on the definition of low-level descriptors and the generation of metrics in the descriptor space [1]. These descriptors are extremely useful in some generic image classification tasks or when classification based on query by example. However, if the aim is to classify the image using the descriptors of the object content this image.

Several methods have been proposed for face recognition and classification, we quote: structural methods and global techniques. Structural techniques deal with local or analytical characteristics. It is to extract the geometric and structural features that constitute the local structure of face of image.

Thus, analysis of the human face is achieved by the individual description of its different parts (eyes, nose, mouth...) and by measuring the relative positions of one by another. The class of global methods includes methods that enhance the overall properties of the face. Among the most important approaches, we quote Correlation Technique used by Alexandre in [23] is based on a simple comparison between a test image and face learning, Principal component analysis approach (eigenfaces) based on principal component analysis (PCA) [25-26-17-18], discrete cosine transform technique (DCT) which based on computing the discrete cosine transform [28], Technique using Neural Networks [27] and support vector machine (SVM) in [24].

There are two questions to be answered in order to solve difficulties that are hampering the progress of research in this direction. First, how to link semantically objects in images with high-level features? That's mean how to learn the dependence between objects that reflected better the data? Second, how to classify the image using the structure of dependence finding? Our paper presents a work which uses three variants of naïve Bayesian Networks to classify image of faces using the structure of dependence finding between objects. This paper is divided as follows:

Section 2, presents an overview of distance tangent; Section 3 describes the developed approach based in Naïve Bayesian Network, we describe how the feature space is extracted; and we introduce the method of building the Naïve Bayesian network, Global Tree Augmented Naïve Bayes (GTAN), Global Forest Augmented Naïve Bayes (GFAN), Tree Augmented Naïve Bayes (TAN) and Forest Augmented Naïve Bayes (FAN) and inferring posterior probabilities out of the network; Section 4 presents some experiments; finally, Section 5 presents the discussion and conclusions.

## II. TANGENT DISTANCE

The tangent distance is a mathematical tool that can compare two images taking into account small transformations (rotations, translations, etc.). Introduced in the early 90s by Simard [30] it was combined with different classifiers for character recognition, detection and recognition of faces and recognition of speech. It is still not widely used. The distance of an image to another image ( $I_1$ ,  $I_2$ ) is calculated by measuring the distance between the parameter spaces via  $I_1$  and  $I_2$  respectively. These spaces locally model all forms generated by the possible transformations between two images.

When an image  $x \in R^{I \times J}$  is transformed (e.g. scaled and rotated) by a transformation  $t(x, \alpha)$  which depends on  $L$  parameters  $\alpha \in R^L$  (e.g. the scaling factor and rotation angle), the set of all transformed patterns

$$M_x = \{t(x, \alpha) : \alpha \in R^L\} \in R^{I \times J}$$

is a manifold of at most dimension  $L$  in pattern space. The distance between two patterns can now be defined as the minimum distance between their respective manifolds, being truly invariant with respect to the  $L$  regarded transformations. Unfortunately, computation of this distance is a hard non-linear optimization problem and the manifolds concerned generally do not have an analytic expression. Therefore, small transformations of the pattern  $x$  are approximated by a tangent subspace  $\bar{M}_x$  to the manifold  $M_x$  at the point  $x$ . This subspace is obtained by adding to  $x$  a linear combination of the vectors  $(x), l = 1, \dots, L$  that span the tangent subspace and are the partial derivatives of  $t(x, \alpha)$  with respect to  $\alpha_l$ . We obtain a first-order approximation of  $M_x$

$$M_x = \{x + \sum_{l=1}^L \alpha_l T_l(x) : \alpha \in R^L\} \in R^{I \times J}$$

The single-sided (SS) TD is defined as:

$$D_{DT}(x, \mu) = \min_{\alpha} \{ \|x + \sum_{l=1}^L \alpha_l T_l(x) - \mu\| \}$$

The tangent vectors  $T_l(x)$  can be computed using finite differences between the original image  $x$  and a reasonably small transformation of  $x$ . Example images that were computed using equation above are shown in Figure 1 (with the original image on the left).



Figure 1. Examples for tangent approximation

### III. BAYESIAN NETWORK

#### A. Definition

Bayesian networks represent a set of variables in the form of nodes on a directed acyclic graph (DAG). It maps the conditional independencies of these variables. Bayesian networks bring us four advantages as a data modeling tool [13-15-16]. Firstly, Bayesian networks are able to handle incomplete or noisy data which is very frequently in image analysis. Secondly, Bayesian networks are able to ascertain causal relationships through conditional independencies, allowing the modeling of relationships between variables. The last advantage is that Bayesian networks are able to incorporate existing knowledge, or pre-known data into its learning, allowing more accurate results by using what we already know.

Bayesian network is defined by:

- A directed acyclic graph (DAG)  $G = (V, E)$ , where  $V$  is a set of nodes of  $G$ , and  $E$  of  $G$ ;
- A finite probabilistic space  $(\Omega, Z, p)$ ;

- A set of random variables associated with graph nodes and defined on  $(\Omega, Z, p)$  as :

$$p(V_1, V_2, \dots, V_n) = \prod_{i=1}^n p(V_i | C(V_i))$$

Where  $C(V_i)$  is a set cause (parents)  $V_i$  in graph  $G$ .

#### B. Inference in bayesian network

Suppose we have a Bayesian network defined by a graph and the probability distribution associated with  $(G, P)$ . Suppose that the graph is composed of  $n$  nodes, denoted  $X = \{X_1, X_2, \dots, X_n\}$ . The general problem of inference is to compute  $p(X_i | Y)$  where  $Y \subseteq X$  and  $X_i \notin Y$ . To calculate these conditional probabilities we can use methods of exact or approximate inferences. The first gives an accurate result, but is extremely costly in time and memory. The second turn requires less resources but the result is an approximation of the exact solution.

A BN is usually transformed into a decomposable Markov network [29] for inference. During this transformation, two graphical operations are performed on the DAG of a BN, namely, moralization and triangulation.

#### C. Parameters learning

In this case the structure is completely known a priori and all variables are observable from the data, the learning of conditional probabilities associated with variables (network nodes) may be from either a randomly or according to a Bayesian approach. The statistical learning calculation value frequencies in the training data is based on the maximum likelihood (ML) defined as follows [11,12]:

$$\hat{p}(X_i = x_k | pa(X_i) = x_j) = \hat{\theta}_{i,j,k}^{MV} = \frac{N_{i,j,k}}{\sum_k N_{i,j,k}}$$

where  $N_{i,j,k}$  is the number of events in the data base for which the variable  $x_i$  in  $x_k$  the state of his parents are in the configuration  $x_j$ .

The Bayesian approach for learning from complete data consists to find the most likely  $\theta$  given the data observed using the method of maximum a posteriori (MAP) where:

$$\theta^{MAP} = \text{argmax } p(\theta | D) = \text{argmax } p(D | \theta) p(\theta)$$

With the conjugate priori distribution;  $p(\theta)$  is a Dirichlet distribution:

$$p(\theta) = \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{i,j,k})^{\alpha_{i,j,k}-1}$$

And the posterior parameter  $(D | \theta)$  :

$$p(\theta | D) = \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{i,j,k})^{N_{i,j,k} + \alpha_{i,j,k} - 1}$$

Thus

$$\hat{p}(X_i = x_k | pa(X_i) = x_j) = \hat{\theta}_{i,j,k}^{MAP} = \frac{N_{i,j,k} + \alpha_{i,j,k} - 1}{\sum_k (N_{i,j,k} + \alpha_{i,j,k} - 1)}$$

D. Structure learning

Structure learning is the act of finding a plausible structure for a graph based on data input. However, it has been proven that this is an NP-Hard problem, and therefore any learning algorithm that would be appropriate for use on such a large dataset such as microarray data would require some form of modification for it to be feasible. It is explained by Spirtes *et al* (2001) that finding the most appropriate DAG from sample data is large problem as the number of possible DAGs grows super-exponentially with the number of nodes present. The number of possible structures is super-exponential in the number of variables. The number of possible combinations G of DAGs of  $n$  variables can be calculated by the recursive formula [20]

$$G(n) = \sum_{k=1}^n (-1)^{k+1} \binom{n}{k} 2^{k(n-k)} G(n-k).$$

Learning algorithms are numerous structures. It identifies the algorithms that are based on causality such that IC and PC. Secondly, we note the algorithms which uses a score in the search for structure and finally the algorithms that make a research project such as K2 and structural EM.

E. Bayesian network as a classifier

1) Naïve bayes

A variant of Bayesian Network is called Naïve Bayes. Naïve Bayes is one of the most effective and efficient classification algorithms.

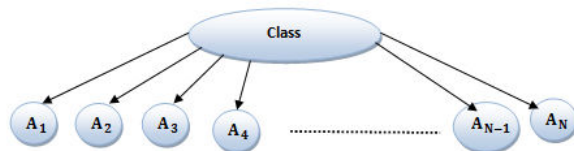


Figure 2. Naive Bayes structure

The conditional independence assumption in naïve Bayes is rarely true in reality. Indeed, naïve Bayes has been found to work poorly for regression problems (Frank *et al.*, 2000), and produces poor probability estimates (Bennett, 2000). One way to alleviate the conditional independence assumption is to extend the structure of naïve Bayes to represent explicitly attribute dependencies by adding arcs between attributes.

2) TAN

An extended tree-like naïve Bayes (called Tree augmented naïve Bayes (TAN) was developed, in which the class node directly points to all attribute nodes and an attribute node can have only one parent from another attribute node (in addition to the class node). Figure 3 shows an example of TAN.

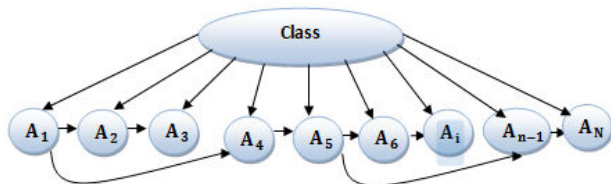


Figure 3. Tree Augmented Naïve Bayes structure (example)

Construct-TAN procedure is described in our previous work [4].

3) FAN

To improve the performance of TAN, we have added a premature stop criterion based on a minimum value of increase in the score in the search algorithms of maximum weight spanning tree. (For more details view [4]).

IV. PROPOSED SYSTEM

In this section, we present two architectures of the classification system developed face. We recall the architecture developed in our article [4] and the new architecture developed in this work. In the latter, we proposed a Bayesian network for each class. So we all structures as classes in the training set. Each structure models the dependencies between different objects in the face image. The proposed system comprises three main modules: a module for extracting primitive blocks from the local cutting of facial images, a classification module of primitives in the cluster by using the method of k-means, and a classification module the overall picture based on the structures of Bayesian networks developed for each class.

The developed system for faces classification is shown in the following figure:

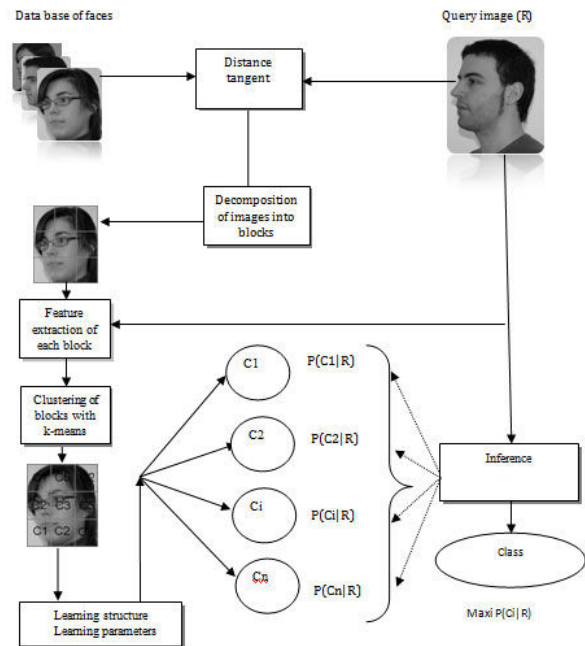


Figure 4: Global system structure

Emphasize at this level we will adopt two types of Bayesian networks. On the one hand, a global network (figure 5) for all classes and other one network for every class.

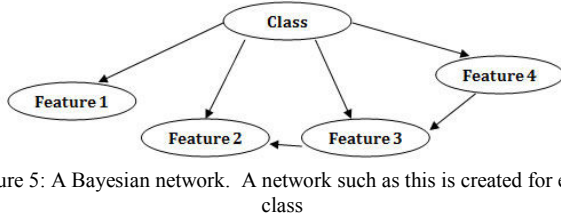


Figure 5: A Bayesian network. A network such as this is created for each class

This sub-section describes the different step of the proposed system:

#### A. Step 1: decomposition of images into blocks

Classifying an image depends on classifying the objects within the image. The studied objects in an image are the following analysis represented by blocks. An example of one image being divided into elementary building blocks that contain different concepts is illustrated in figure 6.

#### B. Step 2: Features Extraction

The feature vector was constructed on the basis of the average, as in (1), the standard deviation  $\sigma$ , as in (2), calculated from model mixtures of Gaussians, and energy, as in (3), the entropy, as in (4), contrast, as in (5), and homogeneity, as in (6), derived from the co-occurrence matrix of gray levels used to cut blocks of the image. These characteristics are very used in the synthesis of texture. The average gives the average value (or moderate) levels of gray from all the pixels in the region of interest (ROI). This parameter represents the location of the histogram on the scale of grays. The images that have a higher average appear brighter. The standard deviation expresses the clustering (or dispersion) around the mean value. Values are more scattered, the greater the standard deviation is large. Most values are clustered around the mean, minus the standard deviation is high. Energy is a parameter that measures the homogeneity of the image. The energy has a value far lower than a few homogeneous areas. In this case, there are a lot of gray level transitions. Homogeneity is a parameter that has a behavior opposite contrast. Has more texture and more homogeneous regions of the parameter is high.

The mean, standard deviation, energy, entropy, contrast, and homogeneity are computed as follow:

The means is defined as:

$$\mu_{mn} = \frac{\sum_{x=1}^M \sum_{y=1}^N I_i(x,y)}{M \times N} \quad (1)$$

The standard deviation is defined as:

$$\sigma^2 = \frac{\sum_{x=1}^M \sum_{y=1}^N [I_i(x,y) - \mu]^2}{M \times N} \quad (2)$$

The energy is defined as:

$$E = \sum_{i,j} (p(i,j))^2 \quad (3)$$

The entropy is defined as :

$$ENT = - \sum_i \sum_j p(i,j) \log p(i,j) \quad (4)$$

The contrast is defined as

$$CONT = \sum_i \sum_j (i - j)^2 p(i,j) \quad (5)$$

The homogeneity is defined as

$$HOM = \sum_{i,j} \frac{1}{1+(i-j)^2} p(i,j) \quad (6)$$

#### C. Step 3: Clustering of blocs with K-means:

Our approach is based on modeling of the image by a 3x3 grid reflecting a local description of the image. So the image to be processed is considered a grid of nine blocks (figure 6). At each block we will apply the descriptors presented in the previous section.

To generate the label vector from vector descriptor, we used the k-means algorithm. Each vector will undergo a clustering attribute, and replace the labels generated vector components descriptors. We use the method of k-means to cluster the descriptor as shown in figure 6.



Figure 6. The result of clustering with k-means into five clusters

#### D. Step 4: Structure Learning

The originality of this work is the development of a Bayesian network for each class. Then, we have compared the results of this network to a global Bayesian network. We have utilized naïve Bayes, Global Structure Tree Augmented Naïve Bayes, Global Structure Forest Augmented Naïve Bayes, Tree Augmented Naïve Bayes for each class (TAN), and Forest Augmented Naïve Bayes for each class (FAN) classifiers. This sub-section describes the implementation of these methods

#### E. Step 5: Parameters learning

NB, TAN and FAN classifiers parameters were obtained by using the procedure as follows.

In implementation of NB, TAN and FAN, we used the Laplace estimation to avoid the zero-frequency problem. More precisely, we estimated the probabilities  $P(c)$ ,  $P(a_i|c)$  and  $P(a_i|a_j, c)$  using Laplace estimation as follows:[4]

$$P(c) = \frac{N(c) + 1}{N + k} \quad P(a_i|c) = \frac{N(c, a_i) + 1}{N(c) + v_i}$$

$$P(a_i|a_j, c) = \frac{N(c, a_i, a_j) + 1}{N(c, a_j) + v_i}$$

Where - N: is the total number of training instances.

- k: is the number of classes,
- $v_i$ : is the number of values of attribute  $A_i$ ,
- $N(c)$ : is the number of instances in class c,
- $N(c, a_i)$ : is the number of instances in class c and with  $A_i = a_i$ ,

- $N(c, a_j)$ : is the number of instances in class  $c$  and with  $A_j = a_j$ ,
- $N(c, a_i, a_j)$ : is the number of instances in class  $c$  and with  $A_i = a_i$  and  $A_j = a_j$ .

F. Step 6: Classification

In this work the decisions are inferred using Bayesian Networks. Class of an example is decided by calculating posterior probabilities of classes using Bayes rule. This is described for both classifiers.

➤ NB classifier

In NB classifier, class variable maximizing equation (7) is assigned to a given example.

$$P(C|A) = P(C)P(A|C) = P(C) \prod_{i=1}^n P(A_i|C) \quad (7)$$

➤ TAN and FAN classifiers

In TAN and FAN classifiers, the class probability  $P(C|A)$  is estimated by the following equation defined as:

$$P(C|A) = P(C) \prod_{i=1}^n P(A_i|A_j, C)$$

Where  $A_j$  is the parent of  $A_i$  and

$$\begin{cases} P(A_i|A_j, C) = \frac{N(c, a_i, a_j)}{N(c, a_j)} & \text{si } A_j \text{ existe} \\ P(A_i|A_j, C) = \frac{N(c, a_i)}{N(c)} & \text{si } A_j \text{ n'existe pas} \end{cases}$$

The classification criterion used is the most common maximum a posteriori (MAP) in Bayesian Classification problems. It is given by:

$$\begin{aligned} d(A) &= \operatorname{argmax}_{\text{classe}} P(\text{classe}|A) \\ &= \operatorname{argmax}_{\text{classe}} P(A|\text{classe}) \times P(\text{classe}) \\ &= \operatorname{argmax}_{\text{classe}} \prod_i P(A_i|\text{classe}) \times P(\text{classe}) \end{aligned}$$

V. EXPERIMENTS AND RESULTS

In this section, we present the results obtained using a database of images. We start by presenting the database with which we conducted our tests, then we present our results according to the used structure (global or one for each class).

A. Data base presentation

Now, we present the results of the contribution of our approach to classify images of some examples of classes from the database used 'Database of Faces'. We have used the GTAV face database found at <sup>1</sup> and the ORL (figure 7) face database found at <sup>2</sup>. The databases are used to evaluate systems classification in the presence of facial variations in lighting conditions, facial expressions (smiling, open/closed eyes) and facial details (glasses / no glasses).



Figure 7. Example of classes of database

B. Structure learning

We have used Matlab, and more exactly Bayes Net Toolbox of Murphy (Murphy, on 2004) and Structure Learning Package described in [11] to learn structure. Indeed, by applying the two architectures developed and the algorithm of TAN and FAN detailed in our previous work [4], we obtained the structures as follows:

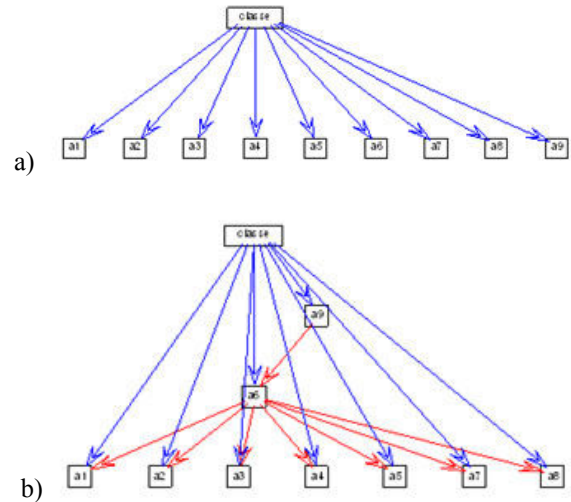


Figure 8. Global network structure

a) Structure of TAN b) Structure of FAN

The application of structure learning algorithm based on global image and on the basis of images from each class gave us the following results:

- The structure of GFAN (Global FAN) returns to the structure of NB, that is to say, the attributes are independent of each other. In fact, the variability of face images and the choice of very high mutual information will decrease the degree of dependency between attributes.
- On the other hand, we note that there is a dependency between the attributes of face images in the same class as shows the structures of class 1, 2, 3,4, and 5.
- By comparing the structure with those of FAN TAN, we see that the choice of a very high dependency criterion will eliminate weak dependencies between attributes and

<sup>1</sup> <http://gpstsc.upc.es/GTAV/ResearchAreas/UPCFaceDatabase/GTAVFaceDatabase.htm>

<sup>2</sup> <http://www.cl.cam.ac.uk/research/dtg/attarchive/face/database.html>

keeps only the bonds that represent a strong dependency between attributes and which will influence positively on the classification results.

estimation to avoid the zero-frequency problem; we obtained the results as follows (Tables I, II):

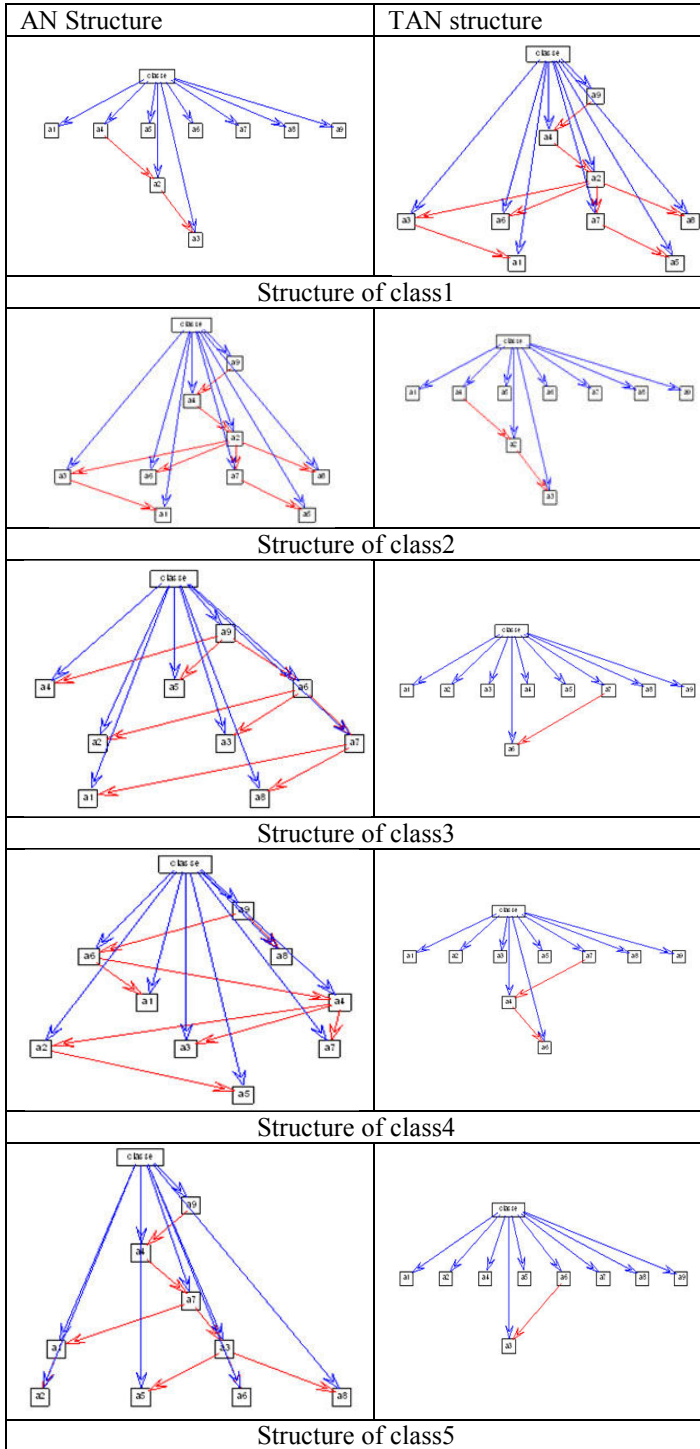


Figure 9. Bayesian Network structure for each class

C. Parameter learning

After estimating the global structure of GTAN, GFAN and the structure of TAN and FAN of each class. We have used those structures to estimate the conditional and a priori probability  $P(c)$ ,  $P(F_i|c)$  and  $P(F_i|F_j, c)$  using Laplace

TABLE I. A PRIORI PROBABILITY OF ATTRIBUTE CLASS  $P(\text{CLASS})$

	P (Class)
Class 1	0.2
Class 2	0.2
Class 3	0.2
Class 4	0.2
Class 5	0.2

TABLE II. A PRIORI PROBABILITY OF  $P(\text{FEATURE}_i|\text{CLASS})$  WITH NUMBER OF CLUSTER  $k=5$

	P (Feature Class)				
	Class 1	Class 2	Class 3	Class 4	Class 5
Feature 1	0,80	0,84	0,28	0,84	0,76
Feature 2	0,04	0,04	0,04	0,04	0,04
Feature 3	0,04	0,04	0,04	0,04	0,04
Feature 4	0,08	0,04	0,60	0,04	0,12
Feature 5	0,04	0,04	0,04	0,04	0,04

D. Results

For each experiment, we used the percentage of correct classification (PCC) to evaluate the classification accuracy defined as:

$$PCC = \frac{\text{number of images correctly classified}}{\text{total number of images classified}}$$

The results of experiments are summarized in Tables IV, V and figure 10 with number of cluster  $k(k=8)^3$ , Naive Bayes, Global TAN, Global FAN, TAN and FAN use the same training set and are exactly evaluated on the same test set.

E. Naïve Bayes

From the table III, we note that the naive Bayesian networks, despite their simple construction gave a very good classification rates. However, this rate is improving as we see that the classification rate obtained by class 3 is 40%. So we conclude that there is a slight dependency between the attributes that must be determined by performing structure learning.

TABLE III. NAÏVE BAYES CLASSIFICATION RATES

Network	class	Number of cluster k	Training data's classification rate	Test data's classification rate
Naïve Bayes	class 1	8	0,95	1,00
	class 2	8	1,00	0,90
	class 3	8	0,75	0,40
	class 4	8	0,90	0,90
	class 5	8	0,80	0,90

<sup>3</sup> The choice of k value discussed in our previous work [4]

F. Tree augmented Naïve Bayes

From the table IV (number of clusters k=8), we note that the rate of correct classification decreases. However, there is a slight improvement in rate of correct classification using a class structure.

G. Forest augmented Naïve Bayes

From the table V, we find that the rate of correct classification GFAN is the same as that obtained by naïve Bayes. Since learning of GFAN structure with a strict threshold equivalent to 0.8 gave the same structure as Naïve Bayes. However, using the structure of class 1, we note that the classification rate was slightly improved to class 1. Also for the Class 3 classification rate increased from 0.4 to 0.9 using the structure of FAN for class 3.

H. Discussion

According to our experiments, we observe that the naïve Bayesian network gave a good result as TAN. Two factors may cause:

- The directions of links are crucial in a TAN. According to the TAN algorithm detailed in our article [4] an attribute is randomly selected as the root of the tree and the directions of all links are made thereafter. We note that the selection of the root attribute actually determines the structure of the TAN result, since TAN is a directed graph. Thus the selection of the root attribute is important to build a TAN.
- Of unnecessary links can exist in a TAN. According to the TAN algorithm, a spanning tree of maximum weight

is constructed. Thus, the number of links is set to n-1. Sometimes it could be a possible bad fit of the data, since some links may be unnecessary to exist in the TAN.

It is observed that NB and Global Fan gave the same classification rate, since they have the same structure.

We note also that the rate of correct classification given by FAN is very high that TAN. Several factors are involved:

- 1- According to the FAN algorithm illustrated in our article [4], the choice of the attribute A root is defined by the equation below, the maximum mutual root has the information with the class.,
 
$$A_{\text{racine}} = \text{argmax}_{A_i} I_p(A_i; C)$$
 when  $i = 1, \dots, n$ . It is obvious to use this strategy, ie the attribute that has the greatest influence on the classification should be the root of the tree.
- 2- Filtering of links that have less than a conditional mutual information threshold. These links are at high risk for a possible bad fit of the data which could distort the calculation of conditional probabilities. Specifically, the use of a conditional average mutual information defined in the equation below as a threshold. All links that have conditional mutual information unless  $I_{\text{avg}}$  are removed.

$$I_{\text{avg}} = \frac{\sum_i \sum_{j, j \neq i} I_p(A_i; A_j | C)}{n(n-1)}$$

where n is the number of attributes.

TABLE IV. RESULTS FOR TREE AUGMENTED NAÏVE BAYES

Network	Structure	class	Training data's classification accuracy	Test data's classification accuracy
GTAN	Global Structure (0,84 seconds)	Class 1	0,65	0,60
		Class 2	0,50	0,40
		Class 3	0,50	0,70
		Class 4	0,40	0,40
		Class 5	0,40	0,50
TAN (Tree Augmented Naive Bayes)	Structure class 1 (0,75 seconds)	Class 1	0,85	0,65
		Class 2	0,60	0,50
		Class 3	0,40	0,40
		Class 4	0,50	0,30
		Class 5	0,45	0,40
TAN (Tree Augmented Naive Bayes)	structure class 2 (0,72 seconds)	Class 1	0,40	0,30
		Class 2	0,60	0,55
		Class 3	0,45	0,40
		Class 4	0,50	0,40
		Class 5	0,55	0,50
	structure class	Class 1	0,45	0,4

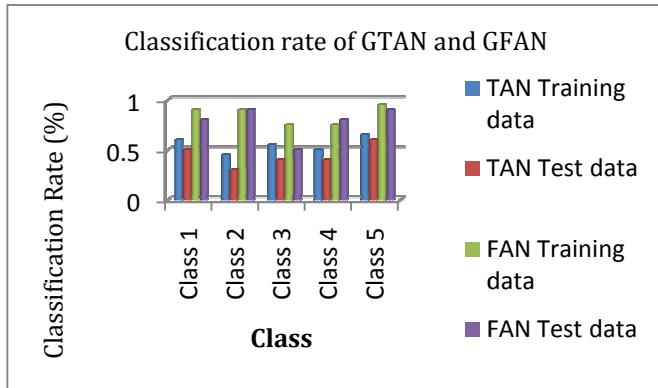
3 (0,81 seconds)	Class 2	0,5	0,4
	Class 3	0,55	0,50
	Class 4	0,45	0,30
	Class 5	0,40	0,3
structure class 4 (0,95 seconds)	Class 1	0,55	0,50
	Class 2	0,55	0,40
	Class 3	0,60	0,50
	Class 4	0,70	0,60
structure class 5 (0,88 seconds)	Class 5	0,55	0,50
	Class 1	0,60	0,50
	Class 2	0,45	0,30
	Class 3	0,55	0,40
Class 4	0,50	0,40	
	Class 5	0,65	0,60

TABLE V. RESULTS FOR FOREST AUGMENTED NAÏVE BAYES

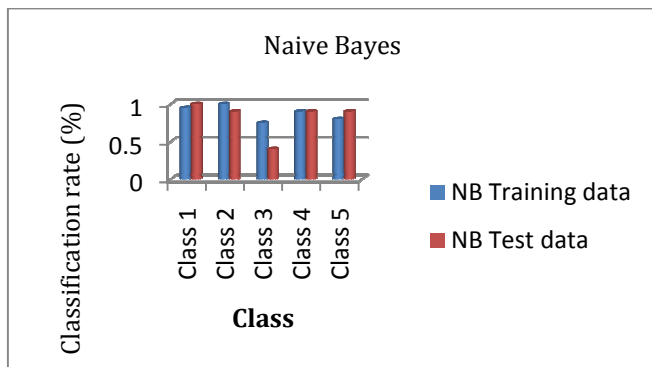
Network	Structure	class	Number of cluster k	Training data's classification accuracy	Test data's classification accuracy
GFAN	Global Structure (0,84 seconds)	Class 1	8	0,95	1
		Class 2	8	1	0,9
		Class 3	8	0,75	0,4
		Class 4	8	0,9	0,9
		Class 5	8	0,8	0,9
FAN with threshold S=0,8	structure of class 1 (0,75 seconds)	Class 1	8	1,00	1,00
		Class 2	8	1,00	0,80
		Class 3	8	0,70	0,80
		Class 4	8	0,70	0,70
		Class 5	8	0,75	0,80
	structure of class 2 (0,72 seconds)	Class 1	8	0,85	0,80
		Class 2	8	1,00	0,90
		Class 3	8	0,70	1,00
		Class 4	8	0,75	0,90
		Class 5	8	0,70	0,90
	structure of class 3 (0,81 seconds)	Class 1	8	0,80	0,80
		Class 2	8	0,90	0,70
Class 3		8	0,90	0,90	
Class 4		8	0,80	0,90	
Class 5		8	0,80	0,30	
structure of class 4 (0,95 seconds)	Class 1	8	0,75	0,80	
	Class 2	8	1,00	0,90	
	Class 3	8	0,75	0,80	
	Class 4	8	1,00	0,90	



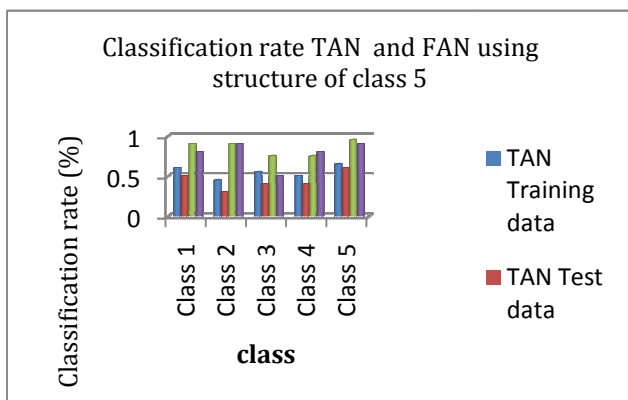
	Class 5	8	0,75	0,70
	Class 1	8	0.90	0,80
structure of class 5 (0,88 seconds)	Class 2	8	0,90	0,90
	Class 3	8	0,75	0,50
	Class 4	8	0,75	0,80
	Class 5	8	0,95	0,90



a)



b)



c)

Figure 10. Classification rates

a) GTAN and GFAN b) Naive Bayes

c) TAN and FAN using structure of class 5

## VI. CONCLUSION

This study is an extension of our previous work [4]. We have developed a new approach for classifying image of faces using distance tangent and the method of k-means to cluster the vectors descriptors of the images. First, we use the technique of tangent distance to calculate several tangent spaces representing the same image. The objective is to reduce the error in the research phase.

Then we have used Bayesian network as classifier to classify the whole image into five classes. We have implemented and compared three classifiers: NB TAN and FAN using two types of structure, a global structure and structure per class presenting respectively the dependence between the attributes inter and intra-class. The goal was to compare the results obtained by these structures and apply algorithms that can produce useful information from a high dimensional data. In particular, the aim is to improve Naïve Bayes by removing some of the unwarranted independence relations among features and hence we extend Naïve Bayes structure by implementing the Tree Augmented Naïve Bayes. Unfortunately, our experiments show that TAN performs even worse than Naïve Bayes in classification (figure 10). Responding to this problem, we have modified the traditional TAN learning algorithm by implementing a novel learning algorithm, called Forest Augmented Naïve Bayes. We experimentally test our algorithm in data image of faces and compared it to NB and TAN. The experimental results show that FAN improves significantly NB classifiers' performance in classification. In addition, the results show that the mean of classification accuracy is better when the number of cluster is optimal that's mean the number of cluster that can reflected better the data. Then, we marked that the structure of FAN per class performs better than Global FAN. This results is explained by the use of structure of FAN per class reflect better the dependence of the attribute intra-class (in the same class), and the use of a global structure reflects better the dependence inter-class (between the classes).

## REFERENCES

- [1] Mojsilovic, "A computational model for color naming and describing color composition of images". *IEEE Transactions on Image Progressing*, vol. 14, no.5, pp.690-699, 2005.
- [2] Zhang Q., Ebroul I. "A bayesian network approach to multi-feature based image retrieval". *First International Conference on Semantic and Digital Media Technologies*. GRECE, 2006.
- [3] Rodrigues P.S, Araujo A.A."A bayesian network model combining color, shape and texture information to improve content based image retrieval systems", 2004.
- [4] Jayech K , Mahjoub M.A "New approach using Bayesian Network to improve content based image classification systems", *IJCSI International Journal of Computer Science Issues*, Vol. 7, Issue 6, November 2010.

- [5] S.Aksoy et R.Haralik. "Textural features for image database retrieval". IEEE Workshop on Content-Based Access of Image and Video Libraries, in conjunction with CVPR. June 1998.
- [6] C.Biernacki, R.Mohr. « Indexation et appariement d'images par modèle de mélange gaussien des couleurs ». *Institut National de Recherche en Informatique et en Automatique*, Rapport de recherche n\_3600\_Janvier 2000.
- [7] Abbid Sharif, Ahmet bakan Tree augmented naïve bayesian classifier with feature selection for FRMI data, 2007.
- [8] Ben Amor, S.Benferhat, Z.Elouedi. « Réseaux bayésiens naïfs et arbres de décision dans les systèmes de détection d'intrusions » *RSTI-TSI*. Volume 25 - n°2/2006, pages 167 à 196.
- [9] Cerquides J., R.L.Mantaras. Tractable bayesian learning of tree augmented naïve bayes classifiers, January 2003.
- [10] Friedman N., N.Goldszmidt. Building classifiers using bayesian networks. *Proceedings of the American association for artificial intelligence conference*, 1996.
- [11] Francois O. De l'identification de structure de réseaux bayésiens à la reconnaissance de formes à partir d'informations complètes ou incomplètes. Thèse de doctorat, Institut National des Sciences Appliquées de Rouen, 2006.
- [12] Francois O., P.Leray. Learning the tree augmented naïve bayes classifier from incomplete datasets, LITIS Lab., INSA de Rouen, 2008.
- [13] Leray P. Réseaux Bayésiens : apprentissage et modélisation de systèmes complexes, novembre 2006.
- [14] Li X.. Augmented naïve bayesian classifiers for mixed-mode data, December, 2003.
- [15] Naim P., P.H.Wuillemin, P.Leray, O.Pourret, A.Becker. *Réseaux bayésiens*, Eyrolles, Paris, 2007.
- [16] Smail L.. Algorithmique pour les réseaux Bayésiens et leurs extensions. Thèse de doctorat, 2004.
- [17] M. Turk and A. Pentland. "Eigenfaces for recognition". *Journal of Cognitive Neuroscience*, 3, 71-86, 1991.
- [18] M. Turk and A. Pentland. Face Recognition Using Eigenfaces. In IEEE Intl. Conf on Computer Vision and Pattern Recognition (CVPR), pages 586–591, 1991.
- [19] A. Nefian and M. Hayes. Hidden Markov Models for Face Recognition. In IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), volume 5, pages 2721–2724, 1998.
- [20] F. Cardinaux, C. Sanderson, and S. Bengio. User Authentication via Adapted Statistical Models of Face Images. *IEEE Trans. on Signal Processing*, 54(1):361–373, 2005.
- [21] Huang,R, Pavlovic.V, N. Metaxas.D , "A Hybrid Face Recognition Method using Markov Random Fields" , *Proceedings of the Pattern Recognition, 17<sup>th</sup> International Conference on (ICPR'04) Volume 3 –* Volume 03 2004 ISBN ~ ISSN:1051-4651 , 0-7695-2128-2.
- [22] Sharavanan.S, Azath.M. LDA based face recognition by using hidden Markov Model in current trends. *International Journal of Engineering and Technology* Vol.1, 2009, 77-85.
- [23] Alexandre Lemieux, « Système d'identification de personnes par vision numérique », Thèse, Université LAVAL, QUEBEC, Décembre 2003.
- [24] Gang Song, Haizhou Ai, Guangyou Xu, Li Zhuang, "Automatic Video Based Face Verification and Recognition by Support Vector Machines" , in *Proceedings of SPIE Vol. 5286 Third International Symposium on Multispectral Image Processing and Pattern Recognition*, edited by Hanqing Lu, Tianxu Zhang, 139-144 (2003).
- [25] Kuang-Chih Lee, Jeffrey Ho, Ming-Hsuan Yang, David Kriegmanz," Video-Based Face Recognition Using Probabilistic Appearance Manifolds ",IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), 16-22 June 2003.
- [26] L. Torres, L. Lorente, Josep Vila , "Automatic face recognition of video sequences using selfeigenfaces " , in *ICASSP, Orlando, FL, May 2002*.
- [27] M.J. Aitkenheada, A.J.S. McDonaldb, "A neural network face recognition system", *Engineering Applications of Artificial Intelligence* 16 (2003) 167–176.
- [28] Z. M. Hafeh and M.D.Levine, "Face Recognition Using the Discrete Cosine Transform", *International Journal of Computer Vision* 43(3), 167–188, 2001.
- [29] D. Heckerman, 1995. A tutorial on learning with Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Redmond, Washington. Revised June 1996.
- [30] P. Simard, Y. Le Cun, J. Denker, and B. Victorri. Transformation Invariance in Pattern Recognition — Tangent Distance and Tangent Propagation. G. Orr and K.-R. Müller, editors, *Neural networks: tricks of the trade*, volume 1524 of *Lecture Notes in Computer Science*, Springer, Heidelberg, pages 239–274, 1998.

#### AUTHORS PROFILE

**Khelifa Jayech** is a Ph.D. student at the department of Computer Science of National Engineering School of Sfax. She obtained her master degree from Higher Institute of Applied Science and Technology of Sousse (ISSATS), in 2010. Her areas of research include Image analysis, processing, indexing and retrieval, Dynamic Bayesian Network, Hidden Markov Model and recurrent neural network.

**Dr Mohamed Ali Mahjoub** is an assistant professor in the department of computer science at the Preparatory Institute of Engineering of Monastir. His research interests concern the areas of Bayesian Network, Computer Vision, Pattern Recognition, Medical imaging, HMM, and Data Retrieval. His main results have been published in international journals and conferences

# Skew correction for Chinese character using Hough transform

Tian Jipeng<sup>1</sup>  
DoS in Computer Science,  
University of Mysore,  
Mysore-570 006, India

G.Hemantha Kumar<sup>2</sup>  
DoS in Computer Science,  
University of Mysore,  
Mysore-570 006, India

H.K. Chethan<sup>3</sup>  
DoS in Computer Science,  
University of Mysore,  
Mysore-570 006, India

**Abstract**— Chinese Handwritten character recognition is an emerging field in Computer Vision and Pattern Recognition. Documents acquired through Scanner, Mobile or Camera devices are often prone to Skew and Correction of skew for such document is a major task and important factor in optical character recognition. The goal of the work is to correct skew for the documents. In this paper we have proposed a novel method for skew correction using Hough transform. The proposed approach with high precision can detect skew with large angle (-90 to +90) the experimental result reveal that the proposed method is efficient compared to well-known existing methods. The experimental results show the efficacy compared to the result of well-known existing methods.

**Keywords**- Handwritten Chinese character; Computer Vision; Pattern Recognition; skew detection; Hough transforms.

## I. INTRODUCTION

Off-line Chinese character recognition is a very important branch of pattern recognition. Chinese is the most population used language. Along the fast development of China, most family has had computer and internet connection already. Online library and data searching become more common during the life. Documents and data imports become a huge problem that we have to face. By manual data input takes too much time, therefore improving the Chinese OCR rate and speed provides efficient works. Recognition of Chinese character is considered a hard problem because of the large categories, the complex structure, and the widely variable and many similar shapes of Chinese character [1]. Generally, the Chinese OCR has preprocessing, feature extraction, recognition, and post processing (see in Fig 1). Effective preprocessing method is an important factor to ensure the recognition rate of whole OCR system. Preprocessing has text layout segmentation, line segmentation, and character segmentation. Because of the skew angle while scanning or writing, it makes the characters in input image skew. The skew characters will seriously affect the recognition results. Therefore, character skew detection and transformation should be done before recognition (Fig.2). The frequently used skew detection methods are Hough, Robust, MST cluster, KNN, and etc. [1-10]. In this paper some experiments are performed using Hough Transform method to detect and correct the skew character image. The Hough technique is particularly useful for computing a global description of a feature, given local measurements. The motivating idea behind the Hough technique for line detection is that each input measurement indicates its contribution to a

globally consistent solution. In case of the parameter space is not than two-dimensional, this kind of transformation has a desired effect. Because of excellent characteristics of Hough transform, such as sensitive to local defects, random noise robust and parallel processing. Hough transform method is widely used in image processing, pattern recognition, and computer vision regions.

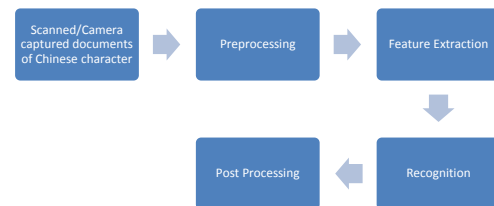


Figure 1: Chinese OCR process

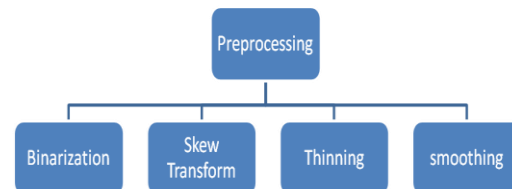


Figure 2: Preprocessing of Chinese OCR

## II. SKEW ANGLE DETECTION

Hough transform (HT) is a technique which can be used to isolate features of a particular shape within an image. The classical Hough transform is most commonly used for the detection of regular curves such as lines, circles, ellipses, etc. Presently, HT is widely used in image analysis, computing vision, and pattern recognition [2]. It becomes a standard tool on pattern recognition. HT parameter space's peak detection is a problem of cluster detection. The thresh value selection is the key to success. There are two methods of Hough Transform. One method is weighted image space to change the peak distribution of parameter space. The other method is directly searching the maximum value of parameter space.

### A. Preprocessing

Hough transform algorithm is mainly applied on binary image (i.e. edge image). Therefore gray image preprocessing should be done before Hough transformation, such as image filtering and edge detection. The preprocessing is the most important preliminary work of Hough transformation. The result of preprocessing directly influences the skew result. Gaussian noise and Impulse noise are two wide known common image noises. In this paper, a method based on multiple median extractions bilateral filtering was used. This method covers both spatial neighborhood correlation and pixel intensity similarity. Through the reference pixel value, which selected by pseudo-median filter, it beautifully protects the edge of the testing image that has Gaussian noise and Impulse noise. Compare to past noise filter methods which only aim to one kind of noise, multiple median extractions bilateral filtering can process images with mixed noise, and it has a good filtering result. Using the method of iterative bilateral filtering instead of Canny operator's Gaussian filtering process and adaptive filtering process, it avoids the fuzzy edge of the images which caused by filtering process, and gets a better edge detection result.

### B. Rough transform algorithm

Standard HT uses the voting way by ballot to realize the straight line test, take an image  $N \times N$  as the example, the value of  $\rho$  is range from  $[-N, \sqrt{2}N]$  the value of  $\theta$  is from  $[0, \pi]$ . Firstly, the  $\theta$  space discrete into  $M$  parameter sector, and thought a non-zero  $(x, y)$  possibly belongs to each  $\theta$  parameter sector of the image space, in other words, it may belongs to any straight-line. In each  $\theta$  parameter sector, according to the equation (1) to calculate the value of  $\rho$ , then falling into the corresponding  $\rho$  parameter sector, finally check each  $\rho$  parameter sector voting number (the length of the straight line), if it more than the setting threshold, it will be considered to be a straight line.

$$\rho = x \cos \theta + y \sin \theta \quad (1)$$

### C. Algorithm implementation

The algorithm concrete realization step is as follows:

a) Ascending order searches in the image, until a non-zero point  $p$  of the image was found, take this point as the seed point.

b) Take the  $P$  point as the original point to select a  $m \times m$  zone  $A_p$ , searching the non-zero points in the  $A_p$  zone, then according to the equation (1), and calculate the straight parameter pair  $(\rho_i, \theta_i)$  of every non-zero point  $p_i$  and  $P$ .

c) Make the deviation range  $\rho$  as  $\Delta\rho$ , the deviation range of  $\theta$  as  $\Delta\theta$ , voting in  $A_p$ , counting the number  $n_i$  of the parameter pair which falling into every parameter interval  $(\rho + \Delta\rho, \theta + \Delta\theta)$ ;

Find out the parameter zone of maximum vote  $n_{\max}$  from  $A_p$ , and get the average value of  $(\rho_i, \theta_i)$ , to reduce the influence

of quantitative error, and get  $(\bar{\rho}, \bar{\theta})$ , set it as the straight line parameter which cross the point  $P$  of  $A_p$ .

d) Set  $T_1$  as a length threshold of  $A_p$ , if  $n_{\max} > T_1$ , then turn to step (6) and search others points that belonging to the line. Otherwise, the line which cross point  $P$  is not exist, and set gray value of  $P$  as zero, then turn to step (1), and restart searching.

e) Set the search zone into entire image, according to equation (1) to calculate the value  $p$  of non-zero point  $P$ , which have been found, and set the value  $\theta$  as  $\bar{\theta}$ . If  $|\rho - \bar{\rho}| < \Delta\rho$ , then  $p$  belongs this line, then cumulate the voting number, and set the gray value as zero.

f) After entire image searching, if the voting number of the line is larger than preset threshold value  $T_2$ , then the line is exist, get the parameter and put into set line( $n$ ),  $n$  is the line number detected. Otherwise the line does not exist.

g) Set the gray value of  $P$  as zero, return step a), and restart searching, until there is no non-zero point in image.

## III. CHARACTER IMAGE ROUGH TRANSFORMATION

Skew image correction is the transformation of coordinate. Set  $(x, y)$  as the dot coordinate of  $P$ , then transform  $\theta$  degree to get image  $P'$ . The corresponded coordinate  $(x, y)$  is  $(x', y')$ . Set the values  $\theta$  positive to clockwise, and negative to counterclockwise. Then the coordinate can be represented as follows:

$$\begin{cases} x' = x \cos \theta - y \sin \theta \\ y' = y \cos \theta + x \sin \theta \end{cases} \quad (2)$$

The image was form by pixels, which is discrete variable. Images can be transformed through the following methods. To horizontal scanned or captured images, while the direction of pixel steps is up-towards, the angle of  $P$  is

$\tan^{-1}\left(\frac{p-1}{\text{rectwidth}}\right)$ ,  $\text{rectwidth}$  is width which has selected;

while the direction of pixel steps is down-towards, the angle of

$P$  is  $-\tan^{-1}\left(\frac{p-1}{\text{rectwidth}}\right)$ . To vertical scanned or captured images, while the direction of pixel steps is right-towards, the

angle of  $P$  is

$\tan^{-1}\left(\frac{p-1}{\text{rectheight}}\right)$ ,  $\text{rectheight}$  is height which has selected; while the direction of pixel steps is left-towards,

the angle of  $P$  is

$-\tan^{-1}\left(\frac{p-1}{\text{rectheight}}\right)$ . While skew correcting, clockwise skew images are token. During determining the skew angle, the lines are represented by sub-pixel step; therefore the skew correction can be implementing through the corresponding offset of each pixel decrease or

increase. The following below is the horizontal and vertical skew correction functions.

$$Y\_shift_j = INT\left(\frac{(p-1)j}{rectwidth}\right) \quad (3)$$

In the function, rectwidth stands for width of rect, INT () is a function to get the closest integer value. Skew angle is positive while step is up-toward, or negative while down-towards.

$$X\_shift_i = INT\left(\frac{(q-1)i}{rectheight}\right) \quad (4)$$

In the function, rectheight stands for height of rect, INT () is a function to get the closest integer value. Skew angle is positive while step is right-toward, or negative while left-towards. Generally, the new image after transform is larger than the original image. The new height and width is:

$$\begin{cases} \text{New\_width} = \text{old\_width} + X\_shift_{\text{old\_height}} \\ \text{New\_height} = \text{old\_height} + Y\_shift_{\text{old\_width}} \end{cases}$$

When the skew angle is positive, the original image pixel of column i and row j corresponds column (i+Y\_shiftj) and row (j - X\_shifti + X\_shiftdold\_height) of new image. When the skew angle is negative, the original image pixel of column i and row j corresponds column (i - Y\_shiftj + Y\_shiftdold\_width) and row (j + X\_shifti) of new image.

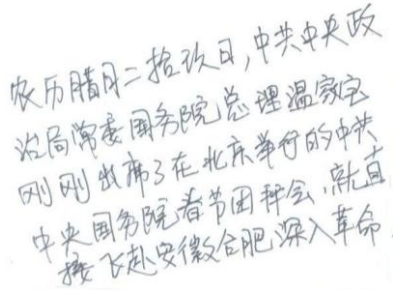


Figure 3: Scanned Handwritten Chinese Character

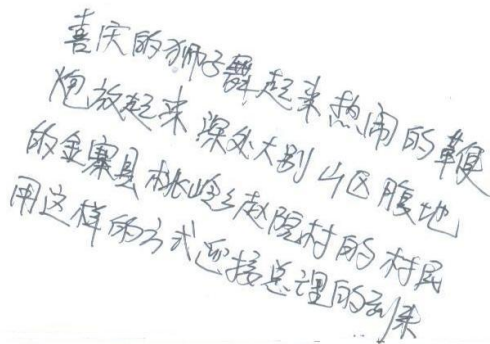


Figure 4: Camera Captured Handwritten Chinese Character

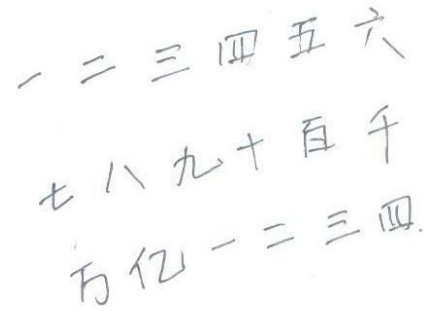


Figure 5: Scanned Handwritten Chinese Numeral

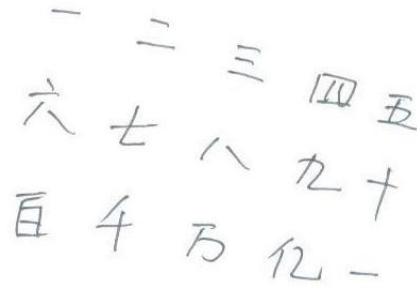


Figure 6: Camera Captured Handwritten Chinese Numeral

#### IV. EXPERIMENTAL RESULT AND ANALYSIS

To evaluate our proposed method, we compare both scanned and camera captured handwritten Chinese character and numerals. Both dataset are documents without tables, which are listed below as Fig 3 to Fig 6. The proposed approach with high precision can detect skew with large angle (-90 to +90) the experimental result reveal that the proposed method is efficient compared to well known existing methods. Table 1 gives the experimental result by using our proposed method. The experimental results show the efficacy compared to the result of well known existing methods. The results show Hough transform has a good performance on noise image skew correction, which does not include tables. Fig. 3 and 5 are scanned images, which skewed while scanning, and according to the experiment result, using Hough transform to correct the skew document are very efficient and exact. Fig. 4 and 6 are camera captured images with a skew angle, which caused by camera capturing angle. The experiment gets a good result too. Therefore, after image document preprocessing, and using our proposed method, it improves the efficient of image skew correction. Hough Transform has high Noise immunity and adaptive.

Table 1: Four skew angles experiments

True Angle in degree	Proposed Method using Hough Transform		
	M	SD	CT (s)
10	10.0	0.327	<b>1.71</b>
15	14.99	0.292	<b>1.78</b>
20	20.00	0.346	<b>1.81</b>
30	30.15	0.429	<b>1.43</b>

## V. CONCLUSION

In summary the paper reveals that only few works have been reported Skew correction for Scanner and camera captured documents .In our work we have corrected skew using Hough transform. Experiments have been performed for documents that without graphics and also for noisy images. Experimental results reveal that it works better for both types with graphics and without graphics and therefore proposed method is efficient, novel and accurate for skew documents. Where as in the case of noisy images the proposed method performance degrades as the noise density increases.

## REFERENCES

- [1] J.Z. Liu, Two-layer assignment method for online Chinese character recognition, 1999,Image Signal Process, Vol 147, 122-125
- [2] Qu Yang, Hough Transform OCR Image Slant Correction Method, 2001, Journal of Image and Graphics, Vol 69A).334-337
- [3] Shutao Li, Skew detection using wavelet decomposition and projection profile analysis, 2006, Pattern Recognition Letters, 28, 555-562
- [4] Xu Jianhua, Image Processing and Analysis, Science Publish, 1992, 194-195
- [5] Fan Linan, Image Processing and Pattern Recognition, Science Publish, 2007.163-169
- [6] G. Louloudis, Text line detection in handwritten documents, 2008,

Pattern Recognition, 41, 3758-3772

- [7] Yoshihiko Mochizuki, N-point Hough transform for line detection, 2009, J.Vis.Commun.Image R. 20,242-253
- [8] Huei-Fen Jiang, A fast approach to detection and correction of skew documents, 1997, Pattern Recognition Letters, 18, 675-686
- [9] Sofien Touj, Generalized Hough Transform for Arabic Printed Optical Character Recognition, The international Arab Journal of Information Technology, Vol. 2, No. 4, October 2005
- [10] F. H. Cheng, Recognition of Handwritten Chinese character by modified Hough Transform techniques, Journal IEEE Transaction on Pattern Analysis and Machine intelligence, Volume 11 Issue 4, April 1989

## AUTHORS PROFILE



**Tian Jipeng** graduated from Department of Computer Science, Preston University America in 2006. He is currently pursuing his PhD from University of Mysore India, under the supervision of Prof. G. Hemantha Kumar on “Character Recognition”. Since 2007, he has been teaching at International College, Huanghuai University, China, as an associate professor on computer science. Currently, three papers have been presented in International Conference, four projects were finished and one won the second prize of Province Scientific and Technological Award in 2009, one textbook on Computer Application is publishing in 2011.

His research interests include numeral recognition, character recognition, and image processing.

# Recombinant Skeleton Using Junction Points in Skeleton Based Images

Mrs. Komala Lakshmi  
Assistant Professor,  
Department of Computer Science,  
Coimbatore-6, Tamil Nadu, India.

Dr.M.Punithavalli  
Dean, School of Computer sciences,  
Dr.SNS Rajalakshmi College of Arts and SNRSONS  
COLLEGE,  
Coimbatore, Tamil Nadu, India

**Abstract**— We perform the task of combining two skeleton images and to produce the recombinant skeleton. We propose the recombinant skeleton algorithm to produce the recombinant skeletons. The existing skeleton representation has been taken and the merge vertex detection algorithm was used before applying the recombinant skeleton algorithm. We can design and apply this recombinant skeleton in motion detection, image matching, tracking, panorama stitching, 3D modeling and object recognition. We can generate or manufacture the true real time object from the recombinant skeleton produced. The proposed method utilize local search algorithm for junction validation. Our frame work suggests the range of possibility in getting the recombinant skeleton. The boundary is essential for any transformation hence the bamboo skeleton algorithm is deployed for computing the boundary and for storing the skeleton together with the boundary. Thus our representation is skeleton with border or outline. From this new skeleton representation the proposed recombinant is achieved.

**Keywords**- *Recombinant skeleton; bamboo skeleton; valance skeleton point (VSP); core skeleton point(CSP); junction skeleton points (JSP).*

## I. INTRODUCTION

In the field of computer vision, recognition of objects in the images is a main and important task which should be performed for understanding the scene. This task may be done using different features of the objects with respect to related application. Some of these features are shapes of objects, color, brightness, motion vector, etc [1]. Based on the application, a feature or a combination of them may be useful and may be applied. Shape of object is a rich and effective feature of the object which has a key role for recognition of objects. In this paper, we aim to reconstruct the objects using their skeleton shape representation [2]. Interest point detection is a recent terminology in computer vision that refers to the detection of interest points for subsequent processing. An interest point is a point in the image which in general can be characterized as follows: it has a clear, preferably mathematically well-founded, definition, it has a well-defined position in image space, the local image structure around the interest point is rich in terms of local information contents, such that the use of interest points simplify further processing in the vision system, it is stable under local and global perturbations in the image domain, including deformations as those arising from perspective transformations (sometimes reduced to affine

transformations, scale changes, rotations and/or translations) as well as illumination/brightness variations, such that the interest points can be reliably computed with high degree of reproducibility. Optionally, the notion of interest point should include an attribute of scale, to make it possible to compute interest points from real-life images as well as under scale changes. By applying the selected interest point as junction point [3]. we can produce the recombinant skeleton that is a unique stable that satisfies the skeleton topology.

## II. BACKGROUND STUDY

The skeleton [4] is important for object representation and recognition in different areas, such as image retrieval and computer graphics, character recognition, image processing, and the analysis of biomedical images. Skeleton-based representations are the abstraction of objects, which contain both shape features and topological structures of original objects. Many researchers have made great efforts to recognize the generic shape by matching skeleton structures represented by graphs or trees. Because of the skeleton's importance, many skeletonization algorithms[5] have been developed to represent and measure different shapes. We can distinguish skeleton points into various types as follows.

### A. Simple Points.

A simple point [6][7][8] is an object point which can be removed without changing the topology of the object.

### B. Regular points.

Regular points on a 1D that have exactly two neighbors.

### C. End-points.

End-points of a curve that have exactly one neighbor

### D. Joints.

The thinness property can be easily checked if the junction points are known in advance. Some skeletonization methods directly identify junction points [9]. If junction points are not known in advance, they have to be identified with another method.

### E. Generating Points

Every Skelton Point is linked to the boundary points that are tangential to its maximal circle. These are called generating points.

#### F. Junction points.

Junction points (where curves meet) which can have three or more neighbors. A junction point [10] should satisfy the following skeleton properties.

- 1) it should preserve the topological information of the original object
- 2) the position of the skeleton should be accurate
- 3) it should be stable under small deformations
- 4) it should contain the centers of maximal disks, which can be used for reconstruction of original object
- 5) it should be invariant under Euclidean transformations, such as rotations and translations, and
- 6) it should represent significant visual parts of objects.

Also if we remove that junction point, then the resulting skeleton will not be a best skeletal extraction of the original image. [11]

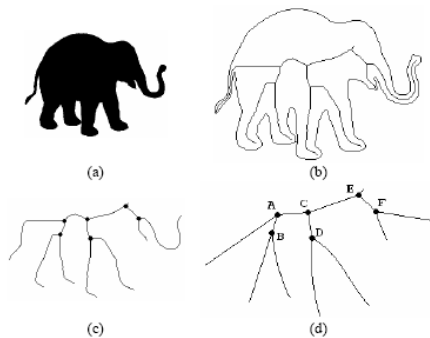


Fig2.1

(a): shape of an elephant, (b): boundary and skeleton, (c): skeleton with junction points, (d): connectivity graph of the skeleton

#### III. SIGNIFICANCE OF JUNCTION POINTS

Great progress has been made that define a significance measure for skeleton points and remove points whose significance is low. In [12] give a complete analysis and compare such pruning methods. Propagation velocity, maximal thickness, radius function, axis arc length, and the length of the boundary unfolded belong to the common significance measures of skeleton points. In [13] present a few significance measures for pruning complex Voronoi skeletons without disconnecting the skeletons. In [14] they combine a flux measurement with the thinning process to extract a robust and accurate connected skeleton.

#### IV. USAGE OF JUNCTION POINTS

One of the biggest uses of skeletons is in analysis of scientific data where complex topologies can be easily explained using line-like drawings. Furthermore, skeletons can be used for reduced modeling and to explain simple physical phenomena. Examples include plume visualization [15], vortex core extraction [16], feature tracking [17] and many others. In all the above usage junction points play a vital role. Using these junction points it is possible to segment a digital image

topologically into meaningful parts. This junction points play the vital role in the applications of skeletons namely computer vision, image analysis, and digital image processing, including optical character recognition, fingerprint recognition, visual inspection, pattern recognition, binary image compression, and protein folding [18].

#### V. METHODOLOGY

##### A. Step 1.

Using bamboo skeleton algorithm to identify the boundary points. [19].

##### B. Step 2.

Using merge vertex detection algorithm to extract the merging point. [20].

##### C. Step 3.

Deploying recombinant skeleton algorithm for merging the selected interest point (merge vertex derived from step 2) to obtain the junction point for the recombinant skeleton.

##### D. Step 4.

Deploy the validation check for the skeleton properties and the junction point to confirm the recombinant skeleton produced

#### VI. RECOMBINANT SKELETON ALGORITHM

1. For the first input image, compute the boundary points using bamboo skeleton algorithm

a)  $B_i \in IP1$  for all  $i=1$  to  $n$ ,  $n$  is the set of all Pixels on the boundary.

b)  $B_j \in IP2$  for all  $j=1$  to  $m$ ,  $m$  is the set of all pixels on the boundary.

2. For each  $B_i \in IP1$  using merge vertex detection algorithm compute,

a)  $VSP_i$  for all  $i=1-n$

b) For each  $B_j \in IP2$  using merge vertex detection algorithm compute,

c)  $VSP_j$  for all  $j=1-m$

3. Using visual Data exploration

Let  $rs$  be the number of boundary points in the recombinant skeleton.

$$rs=m+n$$

Compute the  $VSP_i$  and  $VSP_j$  combination to merge together so as to produce the junction points  $JSP_x$  for  $x=1-rs$ .

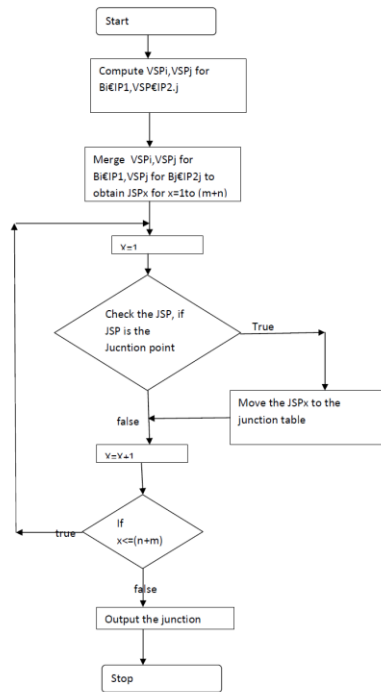
Check for the validity of JSP using local search algorithm [21].

4. Confirm the JSP and connect the two images  $IP1$  and  $IP2$  to produce recombinant skeleton. The output recombinant skeleton produced is unique and stable and preserves the skeleton topology.



## VII. FLOW DIAGRAM

Recombinant skeleton Algorithm flowchart



## VIII. RESULTS AND DISCUSSION

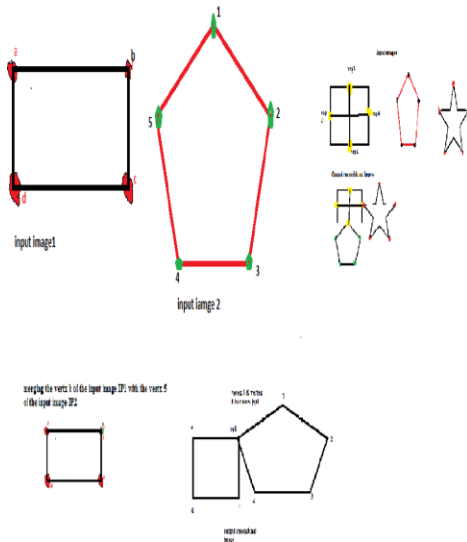


Fig 8.1

The main idea is to extract the boundary points [22] of the input image using bamboo skeleton algorithm. Say  $vsp_1, vsp_2, vsp_3, vsp_4$  as in the fig 8.1. then compute the merging point [23] by testing the validity of the junction point and the stability of the skeleton and topological preservation.

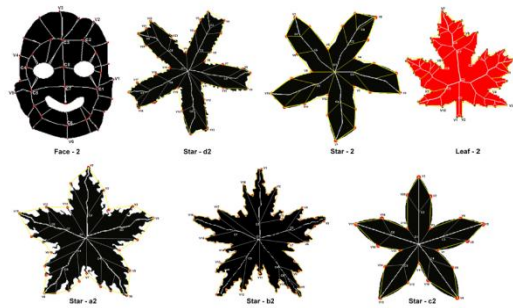


Fig 8.2

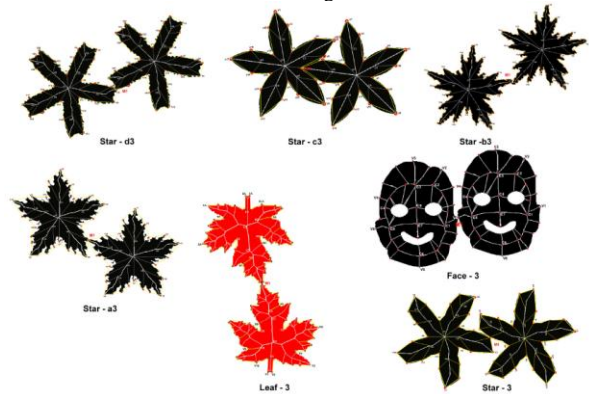


Fig 8.3

Combining the vertex points after the validity check and performance evaluation to obtain the recombinant skeleton structure. The results of the proposed method is depicted in fig 8.2, fig 8.3.

### A. Contribution and Discussion

We have presented an unconventional approach to shape representation and recognition by using skeletons.

1. The boundary point selected from the contour is independent of the threshold value. Whatever be the threshold value, the boundary points should not be removed and it is much more mandatory to remove the boundary deformation. In the existing discrete skeleton evolution method, the author utilizes the threshold value using global measure and hence there are Variety of different skeleton representation based on the threshold value[22], Where as in the proposed method, we do not remove the interested boundary point.
2. The boundary point selected, is checked for its positive maximum curvature. The boundary is formed with the pixels having positive maximum curvature value.[25]
3. Protrusion strength is used to check the interested skeleton point as the junction point.[26], to decompose the skeleton representation into meaningful parts. In this

Proposed method, when merging the two images, the protrusion strength of the merging point becomes the junction point of the recombinant shape representation.

This point is having negative minima of curvature and the end point of that recombinant skeletal arc becomes the boundary point of the newly formed recombinant skeletal representation and the end point on that boundary should have the positive minima of curvature.

That is the two end points of the segment is of opposite extremum[27].

#### IX. PERFORMANCE EVALUATION

As the bamboo skeleton algorithm is free from boundary deformation [24] and the mergevertx detection algorithm is unique and stable, the proposed recombinant skeleton algorithm [29] is also stable and preserves skeleton topology [28]. As the boundary is preserved, any shape matching algorithm can be used to retrieve the shape and recognize the objects.

#### X. CONCLUSION AND FUTURE SCOPE

The obtained skeleton is stable and unique and from that skeleton, we can produce or reconstruct object from skeleton that will definitely provides a platform for the research professionals to manufacture machinery parts from the skeletons of the various images in the databases. Soon we can produce galaxy man may be aliens to some other planets.

As the skeleton representation is derived from the input image by using digital image processing techniques, instead of moving to the recognition process, we are directed to reconstruct the given input image or new 2D objects from the recombinant skeleton representation.

Any modifications developed in the generated real time objects can also be studied and accordingly the work can be extended from objects point of view and its representation. The proposed tracking algorithms can be used for transforming the input image into the modified output image that could be used for further image processing. Also, it is possible to reconstruct the real time objects from the digital representation of the proposed method. The experimental results prove that it could be possible and achievable.

#### ACKNOWLEDGMENT

We thank X. Bai and L. Latecki for the skeleton and curvature code, and we also thank the authors of [22]. The author would like to express her gratitude to Mr. Dr .V. Sengoden , Principal and secretary, SNR SONS COLLEGE, COIMBATORE for providing necessary infrastructure and for his constant encouragement that led to improvise the presentation of the quality of the paper.

#### REFERENCES

- [1] H. Blum. A Transformation for Extraction New Descriptors of Shape, Models for the Perception of Speech and Visual Form, MIT Press, 1967.
- [2] T.Y. Kong, A. Rosenfeld. Digital topology: Introduction and survey. Comp. Vision, Graphics, and Image Proc., 48(3):357-393, 1989.
- [3] Mrs. J.KomalaLakshmi,and Dr.M.Punithavalli,"A Survey on junction points in digital Image Processing" Paper accepted for the international conference IACC'09 IEEE Delhi Section Thapar University, Patiala-147004 Punjab India

- [4] J.Komala Lakshmi and M.Punitha Valli:A Survey on skeletons in digital image processing ,in the international conference proceedings of IEEE Computer Society ,2009:260-269.
- [5] J.Komala Lakshmi and M.Punitha Valli:A Survey on skeletonization in digital image processing ,in the international conference proceedings of Managing Next Generations Software Applications08,Sponsored by CSIR NewDelhi,2008:825-839
- [6] G. Bertrand and Z. Aktouf. A three-dimensional thinning algorithm using subfields, Vision Geometry III, 2356:113-124. SPIE, 1994.
- [7] T.Y. Kong, A. Rosenfeld. Digital topology: Introduction and survey. Comp. Vision, Graphics, and Image Proc., 48(3):357-393, 1989.
- [8] T.Y. Kong, A.W. Roscoe, A. Rosenfeld. Concepts of digital topology, Topology and its App., 46(3):219-262, Elsevier, 1992.
- [9] J. Chuang, C. Tsai, Min-Chi Ko. Skeletonization of Three-Dimensional Object Using Generalized Potential Field, IEEE PAMI, 22(11):1241-1251, 2000
- [10] Detection and characterization of junctions in a 2D image Computer Vision and Image Understanding, Volume 93, Issue 3, March 2004, Pages288-309
- [11] F.F. Leymarie. [3D Shape Representation via Shock Flows](#), Ph.D. thesis, Brown University, May 2003.
- [12]. D. Shaken and A. M. Bruckstein. Pruning Medial Axes. Computer Vision and Image Understanding,69(2): 156-169, 1998.
- [13]. R. L. Ogniewicz, O. Kübler, Hierarchic Voronoi skeletons, Pattern Recognition, 28 (3): 343 –359,1995.
- [14]. K. Siddiqi, S. Bouix, A. R. Tannenbaum, S. W. Zucker. Hamilton-Jacobi Skeletons. International Journal of Computer Vision, 48(3): 215-231, 2002.
- [15] K. Santilli, K. Bemis, D. Silver, J. Dastur, P. Rona. Generating realistic images from hydrothermal plume data, IEEE Vis., 2004.
- [16] D.C. Banks, B.A. Singer. Vortex tubes in turbulent flows: identification, representation, reconstruction, IEEE Vis., 1994.
- [17] B. Vrolijk, F. Reinders, F.H. Post. Feature tracking with skeleton graphs, in Data Visualization: The State of the Art, pp. 37–52, Kluwer Academic Publishers, 2003.
- [18] A Transformation for Extracting New Descriptors of Shape" by H. Blum, in Models for the Perception of Speech and Visual Form, W. Whaten-Dunn (Ed.). MIT Press: Cambridge, MA, pp. 362–380R. Bergevin and A. Bubel
- [19] J.Komala Lakshmi and M.Punitha Valli,"discrete skeleton reconstruction using bamboo skeleton"communicated to IEEE transaction on image processing.
- [20] J.Komala Lakshmi and M.PunithaValli "Impact of Boundary points in skeleton based images " in the international journal of advanced engineering and applications IJAEA: An International Journal of Advanced Engineering & Applications Year 2010, Volume-1, June Issue,Print ISSN : 0975 – 7783,Online ISSN : 0975 – 7791, \_steps-india.com/ijaea/cfp.html
- [21]Shape Recognition by Clustering and Matching of Skeletons Hamidreza Zaboli Amirakbar University of Technology, Tehran, Iran
- [22]. Xiang Bai and Longin Jan Latecki, "Discrete Skeleton Evolution", EMMCVPR,Ezhou, China, 2007. (Applications: visual skeletons, corner detection, shape decomposition)
- [23] J.Komala Lakshmi and M.Punitha Valli," Published an article titled ,"Computation of merging points in skeleton based digital images" in the Global Journal Of computer science and technology GJCST Volume 9- Number 4 September 2009 Issue, \_www.globaljournals.org Print ISSN : 0975 – 4350,Online ISSN : 0975 – 4172
- [24] Xiang Bai, Longin Jan Latecki, and Wen-Yu Liu: "Skeleton Pruning by Contour partitioning with Discrete Curve Evolution," IEEE Trans. PAMI, 29(3): 449-462, 2007
- [25] Aslan, C.; Erdem, A.; Erdem, E.; Tari, S.; Microsoft Corp., Redmond, WA ,Disconnected Skeleton: Shape at Its Absolute Scale, pattern analysis and machine intelligence,ieee transaction on,IEEE computer society, **Volume:** 30 Issue :12, 02 February 2008

- [26] JingTing Zeng, Rolf Lakaemper, XingWei Yang, Xin Li, Temple University, Philadelphia, PA. 2D Shape Decomposition Based on Combined Skeleton-Boundary Features, Lecture Notes in Computer Science
- [27] Michael Leyton, Symmetry-Curvature Duality, computer vision ,graphics and image processing, **38**, 327-341 (1987).
- [28] J. komala lakshmi and M.punithavalli, Published an article titled " 2D shape reconstruction based on combined skeleton boundary features" in the international journal of image processing ,  
ISSN (Online) 1985-2304, Volume 4 – Issue 4 September 2010, Malaysia, [www.cscjournals.org](http://www.cscjournals.org)
- [29] J. komala lakshmi and M.punithavalli, "a survey on Performance evaluation in object detection techniques of Digital Image Processing " Internation journal of Computer science Issues vol 7, Issue 6, November 2010. ISSN Online 1694-0814, IJCSI, Maritius.

AUTHORS PROFILE



The author Mrs J.Komala Lakshmi , has finished her B.Sc Mathematics from SeethaLakshmi Achi College for women ,Pallathur, during 1995. She has completed her M.C.A from J.J.college Of arts and Sciences, Pudhukottai during 1998. She has been Awarded her M.Phil Computer Science, Bharathiar University during 2008. She Has been Awarded with the Silver meadal for her academic Excellence in M.C.A from the Institution from. She has Nine years of teaching experience in Collegiate service. She is currently working as Assistant Professor, Department of Computer Science, SNR SONS COLLEGE , Coimbatore. She has presented her papers in three international conferences .She has published three papers in four international journals .She also contributed a book publications to the computer science students. She was a Graduate Student Member of IEEE. She has been awarded with the recognition from Who's Who in the World 2011 , 28th Edition for her notable contribution to the society.

# ID Numbers Recognition by Local Similarity Voting

Shen Lu<sup>1</sup>, Yanyun Qu<sup>2,\*</sup>, Yanyun Cheng<sup>3</sup>, Yi Xie<sup>4</sup>

Xiamen University  
Computer Science Department  
Xiamen, China

**Abstract**—This paper aims to recognize ID numbers from three types of valid identification documents in China: the first-generation ID card, the second-generation ID card and the driver license of motor vehicle. We have proposed an approach using local similarity voting to automatically recognize ID numbers. Firstly, we extract the candidate region which contains ID numbers and then locate the numbers and characters. Secondly, we recognize the numbers by an improved template matching method based on the local similarity voting. Finally, we verify the ID numbers and characters. We have applied the proposed approach to a set of about 100 images which are shot by conventional digital cameras. The experimental results have demonstrated that this approach is efficient and is robust to the change of illumination and rotation. The recognition accuracy is up to 98%.

**Keywords**—template matching algorithm; ID number recognition; OCR

## I. INTRODUCTION

The admission control of private or safety regions has been widely used nowadays. A person is required to show his valid identification document when he enters airports, hotels, and railway stations, etc. In China, there are three main types of valid identification documents: the first-generation ID card, the second-generation ID card and the driver license of motor vehicle. If all identification documents are checked manually, it will be laborious and monotonous for officers/administrators. Furthermore, along with the day-by-day distensible human contacts, the needs of valid identification documents inspection will grow; however, just depending on the manual control may lead to lower efficiency. So it is necessary to implement the automatic checking of valid identification documents. In this paper, we aim to recognize the ID numbers in the three types of documents mentioned above.

These identification documents have different layout, as shown in Table I. There are 15 digits, or 18 digits, or 17 digits along with a verification letter in the first-generation ID card, and the face image is in the left of the card. There are 18 digits, or 17 digits along with a verification letter in the second-generation ID card, and the face image is in the right of the card. And there are 15 digits, 18 digits, or 17 digits along with a verification letter in the driver license, and the face image is

the right bottom of the driver license. When referring to the positions of the ID numbers, they are both at the bottom of the two ID cards, and are in the middle of the driver license.

In order to deal with these three layouts, we need to solve three difficulties. The first is to locate ID numbers. The second is to recognize the ID numbers considering the impacts of illumination, rotation and blur. The third is to implement the real-time processing.

TABLE I. THE MAIN LAYOUT IN ID DOCUMENTS

	The Positions of Face Image	The Positions of ID Numbers	The Compositions of ID Numbers
<b>First-generation ID Card</b>	In the left of the card	At the bottom of the card	15 digits; or 18 digits; or 17 digits + 1 letter
<b>Second-generation ID Card</b>	In the right of the card	At the bottom of the card	18 digits; or 17 digits + 1 letter
<b>Driver License</b>	In the right bottom of the driver license	In the middle of the driver license	15 digits; or 18 digits; or 17 digits + 1 letter

Optical Character Recognition (OCR) is an important branch in image recognition. There are many works to introduce handwriting recognition. Suen [1] surveyed the handwriting problem before. And the neural network was widely used to solve handwriting recognition [2, 3]. But classical methods based on neural network are time-consuming in training stage. Tappert et al. [4] proposed an on-line approach for handwriting recognition. After that, digital number recognition became an active research problem [5-10] because it was nearer to the industrial application.

However, the existing works are not suitable to our scenario, because the digital numbers are mainly standard printed in the ID card, which may simplify our problem. We design a target-oriented systematic approach which can avoid the complicated middle computation. We focus on making our approach embedded in hardware devices which require our approach to be fast and efficient in computation. The proposed approach of ID number recognition is based on the similarity voting. Unlike the prior works, the proposed approach need not train a lot of data beforehand. Our experiments demonstrate that the recognition accuracy achieves 98% and the speed is less 1 second per image.

The research work was supported by the National Basic Research Program of China under Grant No. 2007CB311005, the Fundamental Research Funds for the Central Universities, No.2010121067, and National Defense Basic Scientific Research program of China under Grant B1420110155.

The rest of this paper is organized as follows: In section 2, we briefly discuss related works. Then we describe the extraction and recognition of ID numbers in section 3 and systematically evaluate our method in section 4. Finally, we draw conclusions.

## II. RELATED WORKS

OCR aims at making computer recognize text characters. Recently, the automatic identification technology for printed and handwritten numbers and characters has been applied in many fields. Especially, the technology for the ID number recognition gets more attention because of its wide applications.

A few years ago, most researchers paid attentions to locating the characters in the first generation ID card [15, 17] and segmenting the ID numbers [14, 16]. After that, the researchers rushed to develop new recognition system for the second generation ID card [12, 13]. However, the existing ID number identification systems have several disadvantages as follows:

- All images are required to be gotten by scanners or CCD cameras under the interior light. However, these two devices are expensive and not popular which limits the application of these systems.
- Most existing systems almost just pay attention to the second-generation ID card, while in China there are three types of valid identification documents.
- The major technologies used in digital recognition are the BP neural network and the geometric point features based on the digital Hilditch thinning. However, both technologies are rarely used in real applications because of the high complexity and the low efficiency.

In this paper, we design a systematic approach and develop a system to deal with the images shot by conventional digital cameras, and our approach is robust to blur, rotation, illumination and over-exposure. And besides the second-generation ID card, the system can identify the ID numbers from the first-generation ID cards and the driving licenses automatically. The proposed approach is a simple but efficient local-similarity-voting method, which combines an improved template matching algorithm with the structure features of digital numbers. Many experiments have verified that it can achieve a high rate of accuracy.

## III. THE FRAMEWORK OF OUR APPROACH

The framework of our approach is shown in Fig.1. There are mainly four parts in our system: firstly, we detect the face image; secondly, we estimate the ID numbers region; thirdly, we recognize the ID numbers; finally, we verify and correct the ID numbers and output the result.

Because the layout of the three types of documents is different, it is important to ascertain the type of the query image. We use a simple but efficient method to acquire the type of a query image. We observe that the face image is a salient feature for the three documents. The face image locates in the left of the first-generation ID card, and in the right of the second-generation ID card, and in the right bottom of the driver

license. So we can ascertain the type of the query image according to the location of face image. We detail our approach in the following.

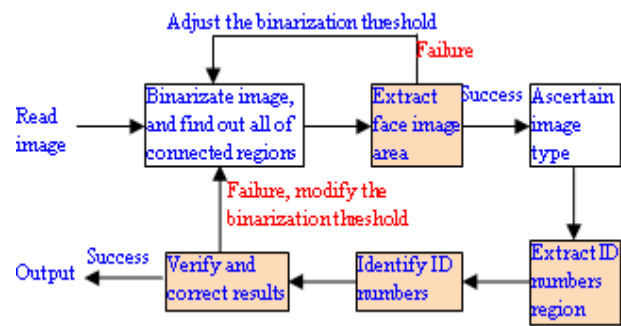


Figure 1. The framework of our approach.

### A. Face Localization

Considering that the human face occupy the most remarkable position in the ID cards and the driver license, we use an adaptive threshold to extract the human face area, and then ascertain the image type according to the location of the human face.

We define the pixels between 180 and 255 gray level as "bright pixels". If the bright pixels in the document image reach to 5%, the image belongs to "bright" images; otherwise, it is a "dark" image. The initial binarization threshold is set as 100 for the bright images and 70 for the dark images (both threshold values are acquired by experience).

We find that when the binarization threshold value is suitable, the face image usually forms the largest connected regions in the binary images: (1) in the first-generation ID card, the hair, the collar, and the five-pointed star regions are universally most conspicuous in the whole binary picture; (2) in the second-generation ID card, the hair, the collar or the entire head regions usually are the largest connected regions; (3) in the driver license, the hair, the collars, the seals or the whole head regions usually occupy most conspicuously; as shown in Fig.2. Therefore, the binarization threshold value can be adaptively decided by the ratio between the connected region area and the document region area. We search the connected regions in the query image, and analyze the ratio between the largest connected region area in the left side to the one in the right side. Furthermore, we extract the face region according to the location of the largest connected region. Finally, we ascertain which type the query image belongs to. The details of the judging rule are as follows:

- If the ratio between the area of left-side largest connected region to the right-side one is under 0.5, the image may be a second-generation ID card or a driver license. In this case, in order to decide the ID card type, we need to associate with the location of the largest connected region in the document region: if the largest connected region is in the right of the query image, it is a second-generation ID card; if the largest connected region is in the right bottom of the query image, it is a driver license.

- If the ratio between the left-side largest connected region area to the right-side one is above 1.0, and the image has been judged as neither a driver license nor a second-generation ID card, that is, the ratio is not conformed to the two conditions mentioned above, and then it is a first-generation ID card.
- If the ratio between the left-side largest connected region area to the right-side one is above 0.75 under 2.5, the image is a driver license.

We do not use the Otsu's method in [11] to modify the binarization threshold, because the images shot by conventional cameras are often noisy and influenced by illumination changes. Moreover, in the three types of documents, the color information is different. The Otsu's method does not obtain the expected results.

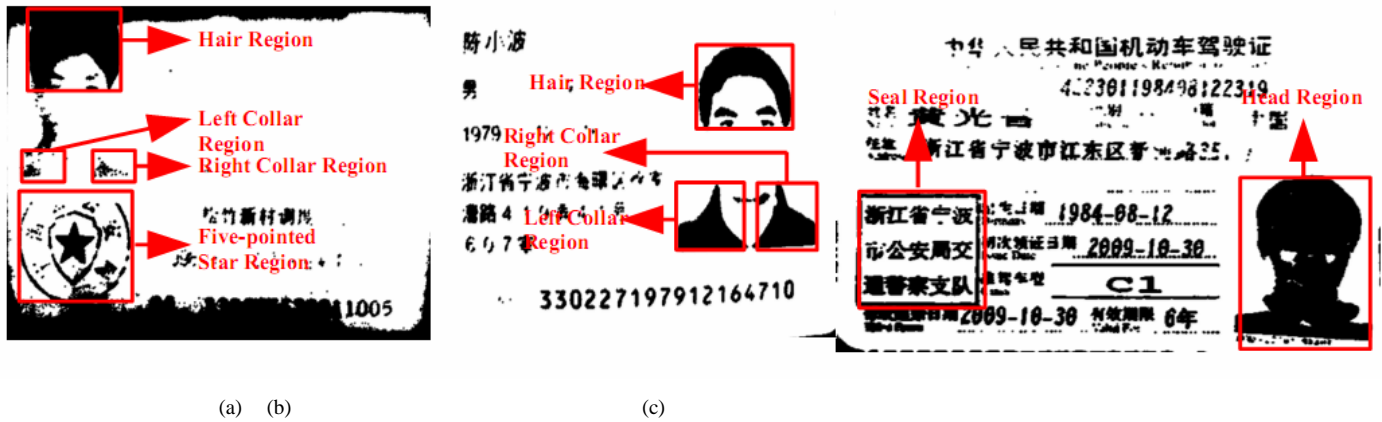


Figure 2. The examples of binary images: (a) the first generation ID card; (b) the second generation ID card; (c) the driver license.

### B. ID Number Localization

According to the face location, we can ascertain the image type, and then locate the ID numbers region. In the first-generation ID card and the second-generation ID card, the ID numbers are all in the bottom of the image, while they are in the middle of the driver license. However, there are two inevitable problems in ID numbers region detection, one is the illumination and the other is the rotation. In the following, we detail our approach on dealing with the illumination and the rotation.

1) *Special details about illumination:* Variations in the ambient illumination pose a significant challenge to ID numbers recognition algorithms. We propose a simple method to compensate such variations and boost recognition performance. Illumination variations cause intensity changes among each point and its neighbors. We modify the binarization threshold iteratively which is increased gradually for the bright images, or is lowered gradually for the dark images, until the ratio of the black pixels in the literalness region reached above 1/6 but under 1/3. Fig 3(a) shows the initial image which is the middle result after face localization stage. Its black pixels distribute very little in the literalness region, so we increase its binarization threshold, and Fig 3(b) shows the result after the first threshold is modified. But the ratio of the black pixels is still under 1/6, we have to modify the threshold iteratively, until the ratio of the black pixels reach 1/6 or above. Fig 3(c) shows the middle result after the threshold value is modified 4 times, and Fig 3(d) shows the

final result image. We can observe that the literalness region is restored sequentially.

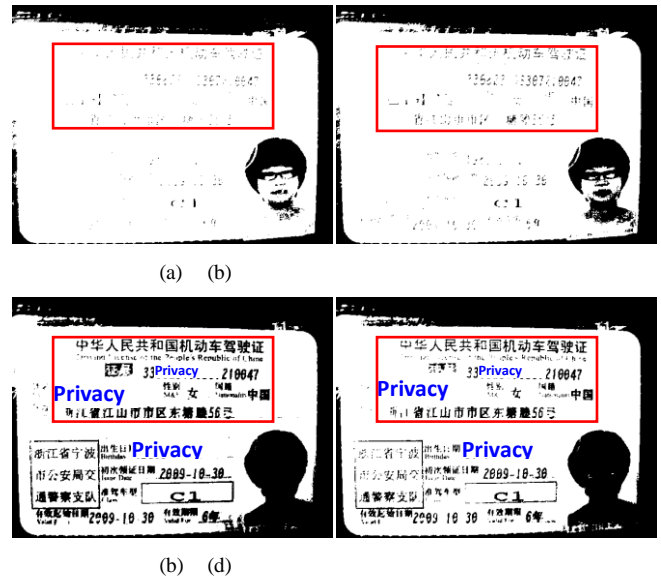


Figure 3. Schematic diagrams for illumination image. The red box area is the threshold adjustment region. (a) Initial binary image; (b) After first threshold modified; (c) After 4th-time threshold modified; (d) Final binary image

2) *Special details about rotation:* Image distortion caused by rotation is another challenge to ID numbers recognition. Considering that the face image is a salient feature in the ID card, we find the baseline based on the key points and compute

the rotation angle. As shown in Fig 4, there are two pairs of the key points in the face, one is the pair of collar-end points, and the other is the pair of temples. In an upright image, each pair of key points ought to lie on a horizontal line. So we firstly detect the two collar-end points by extracting the collar connected region, and then compute the angle between the line passing through the two points and the horizontal line. We use the two temples to verify the rotation angle.

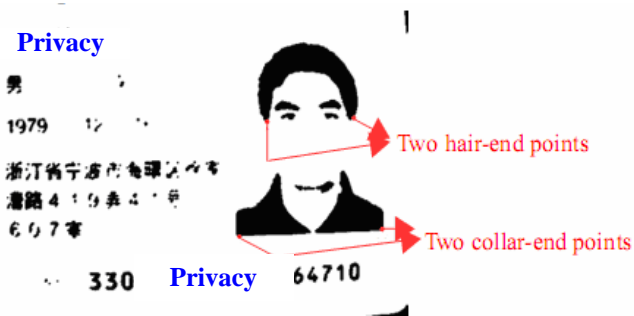


Figure 4. Baseline detection based on pairs of key points.

### C. ID Number Extraction

Since the ID numbers region contains more than one character, in order to recognize the character, we must segment each character. We use the projection profile to solve the problem. As shown in Fig.5, we firstly transform the ID numbers region to a binary image, and then compute its horizontal and vertical histogram, finally segment each character.

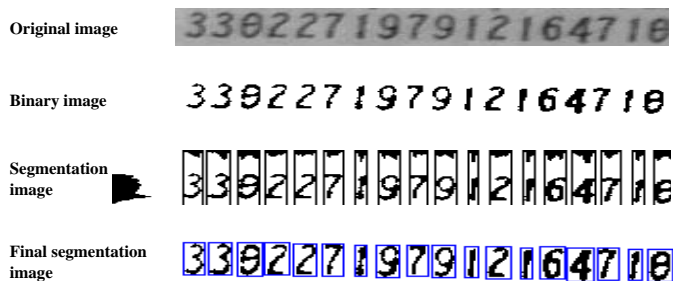


Figure 5. An example for histogram segmentation.

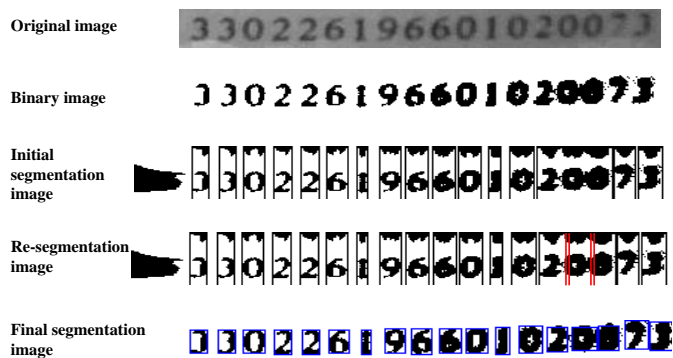


Figure 6. An example for adhesion segmentation.

Number adhesion is another challenge for us, yet we find that the aspect ratio of a single character region is about 1/2. So if the aspect ratio exceeded 1/2, there is adhesion between the characters. We gradually modify the binarization threshold until all the characters have been segmented distinctly. The processing is shown in Fig. 6.

### D. ID Numbers Recognition

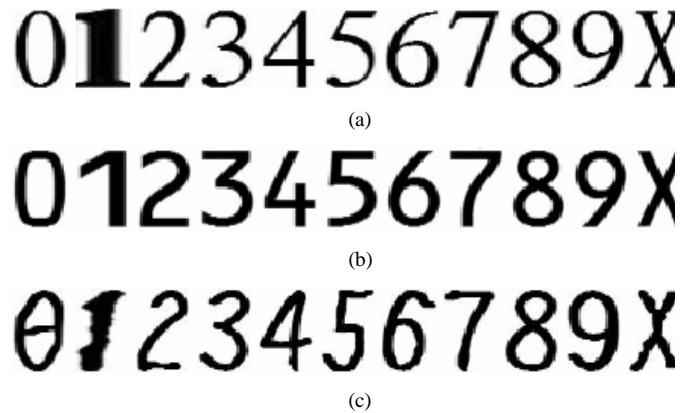
Among the recognition algorithms, the template matching algorithm is undoubtedly popular. It measures the similarity between the templates and the query image, and the query image belongs to the character whose template has the largest similarity value. Obviously, the template matching is based on the characters' geometrical features. Take two-dimensional image for an example, we briefly introduce the matching algorithm.

Firstly, we search the matching pair points between the query image and the template. Let  $p$  denote the point in the template and  $p'$  denote the point in the query image; if the gray-scale value of  $p$  is equal to that of  $p'$ , the two points are matched. We count the number of the matching points as the similarity score. Let  $f(x, y)$  be the query image, and  $t(x, y)$  be the template, and  $r$  be the matching score. We define the foreground as 1 and the background as 0, therefore the similarity score is computed as

$$r = \frac{\sum f(x, y) \cdot t(x, y)}{\sum t(x, y)}$$

and the related characters have the greater similarity value. Due to the difference of the characters' fonts in the three types of identification documents, we adopted three sets of different font templates for matching. Fig.7 displays all of the templates we used, and the size of the template is  $30 \times 45$ .

Figure 7. The templates in three types of fonts: (a) the font templates of first-



generation ID cards;(b) the font templates of second-generation ID cards;(c) the font templates of driving licenses.

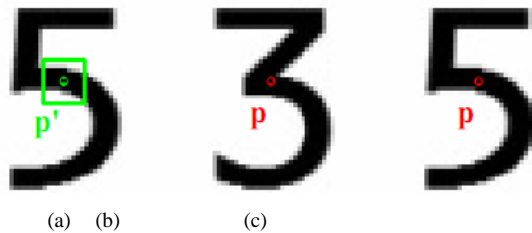
However, the classical template matching algorithm is susceptible to noise, and the conventional digital camera may make noise. In order to improve the recognition accuracy, we propose a novel template matching method which is based on similarity voting. And we compute the similarity score not

comparing the point pair between the template and the query image, but based on the local patch.

Take the point  $p$  in the binary standard template for example, as shown in Fig. 8, we search its corresponding point  $p'$  in the query image. We judge whether there is a point whose value is equal to that of the point  $p'$  in the neighborhood of the point  $p$ . And we set a counter for each standard template. If the point exists, the counter adds one. Followed by the operation, we search for each point of the template in the query image, and the amount of the successful matching points is used to compute the similarity score as

$$r = \frac{\sum_{i=2}^{n-1} \sum_{j=2}^{m-1} \sigma \left( \sum_{p=i-k}^{i+k} \sum_{q=j-k}^{j+k} f(i, j) \cdot t(p, q) \right)}{(n-2) \cdot (m-2)},$$

where  $n$  and  $m$  denote the width and length of the template;  $k$  denotes the size of searching window, in our experiments the size of searching windows is  $3 \times 3$ , and the function  $\sigma(\cdot)$  is defined as



$$\sigma(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

Figure 8. An example for the improved template matching algorithm. (a) The input image; (b) Template of the digit 3; (c) Template of the digit 5.

#### E. Improved Template Matching Algorithm with the Structure Features of a Character

In order to improve the recognition accuracy, we propose to combine the structure features of a character with the improved template matching algorithm, because the structure features of a character are different from each other. Take the digit "5" in the second-generation ID card for an example, as shown in Fig.9, if we draw a vertical line at the half width of the template, we will get six intersections. If we draw a horizontal line at a quarter height of the template, we will get two intersections, which are both on the left side of the template. If we draw a horizontal line at  $2/3$  height of the image, we will obtain two intersections which are both on the right side of the template. We select the distinctive rules for each digital in order to further verify the query image. To the matched candidate templates which pass the verification, we increase their voting weights; and to the matched candidate templates which do not pass the verification, we lower their voting

weights. Then we choose the digital template which has the largest voting weight as the new matched template.

The method further improves the recognition accuracy, especially for the low-resolution images, rotated images, as well as the images which has larger projective transformation.

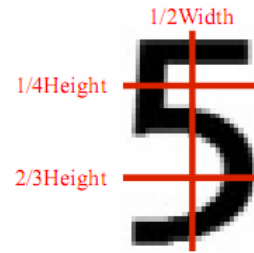


Figure 9. The structure features of number "5"

#### F. ID Number Verification

As we know there are some rules designed for the ID number. For the 18-bit ID numbers, the first 6 digits stand for an area code, the digits from the 7th to 14th bit stand for the birthday of a person, the digits from 15th to 17th bit stand for the born order on the same day, and the 18th bit stand for a check code. The last character of the ID number can be computed according to the check code rules. Moreover, for the 15-bit ID numbers, the first 6 digits stand for an area code, the digits from 7th to 12th bit stand for the birthday of a person, and the 13th to 15th bit stand for the born order on the same day. We can correct the recognition result with the help of those rules mentioned above. Therefore, we save two other candidate numbers for each ID bit. The three candidate templates are of the largest weights. We verify whether the recognition result is satisfied the arrangement rules of ID number, if the verification is successful, we output the recognition result; otherwise, we alternately replace each bit with its candidate template, and until the verification is successful. The final recognition result is the candidate ID number with the largest similarity sum of the 18 or 15 bits. If there are not any successful candidate template for a bit, we modify the binarization threshold of the whole image and start our operation again.

For an example, in the recognition stage we obtain an ID code: 330226196601020013, but the ID code is not satisfied with the arrangement rules of 18-bit ID numbers, and we find in the verification stage that it is a mistake. We use the two other candidate digits one by one to replace the current bit. We find that the ID number is successful to verification when the digit 1 in the 17th bit is replaced with the digit 7, and it has the largest similarity sum in all of the alternative codes that could successfully pass the information verification test, so we choose it as the final recognition result.

The verification stage can improve the recognition accuracy further.

## IV. EXPERIMENTAL RESULTS

We implement our approach on a dataset which has about one hundred collected ID card images. The system is supported by VC6.0++, and is running on PIII.8G with 512MB memory. The running time is about 1000 milliseconds average for each



query ID image. Table II shows some recognition results, and they demonstrate that our approach is not only successful to recognize the ID number in the good images, but also robust for the variation of image rotation, blurring and illumination and the low-resolution image.

Considering that the ID number is a personal privacy, we do not show the whole result of each ID number, and the part of ID number in the original image is occluded. So the star is corresponding to the occluded digits.

In order to evaluate the effect of the structure features of a character, we compare template matching algorithm with the structure features and the one without the structure features. Fig.10 demonstrates the recognition accuracy of the both two configurations. The recognition accuracy for the approach with the structure features is 98.51%, and the one for the approach without the structure features is 94.55%, the structure features can improve the accuracy of the recognition by about 4%.

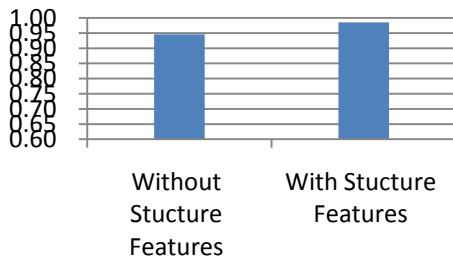


Figure 10. Improvement of recognition with structure features

In order to evaluate the performance of our approach, we also compare our method with OCR engine. Currently, there is rarely OCR engine aiming at recognizing digits only. We use a popular OCR engine named Tesseract for comparison. Tesseract was one of the top 3 engines in the 1995 UNLV Accuracy test. Even to the present, it is probably one of the most accurate open source OCR engines. The source code is available on Google Code [18]. As the OCR engine required pure text image as input, we firstly extract the ID areas from the ID card images. In the ID regions, Otsu's method is used for binarization and the resulting binary images are used as the input of the OCR engine. We compare our method with OCR according to the recognition precision and the average time of processing the recognition ID number except the ID number detection.

As shown in Fig. 11, the precision of our approach achieve 98.51%, which is much higher than the Tesseract OCR engine. Average processing time of proposed method to recognize ID numbers is 45ms, which is almost 4 times faster than the Tesseract OCR engine.

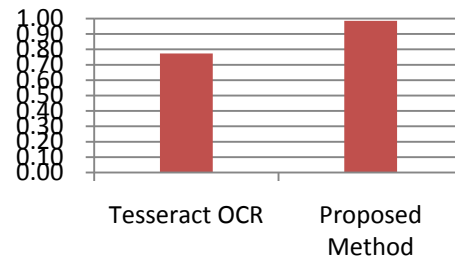







Figure 11. Comparison of Tesseract OCR with the proposed method

TABLE II. SOME RECOGNITION RESULTS

Image Type	Original Image	Face Region	Results
Standard Image			Binarization threshold value: 60 First-generation ID card ID number: 332625*****100555
			Binarization threshold value: 70 Second-generation ID card ID number: 330227*****164710
			Binarization threshold value: 100 Driver license ID number: 330227*****164710
Rotation			Binarization threshold value: 70 Adjust rotation Second-generation ID card ID number: 330227*****164710
			Binarization threshold value: 100 Adjust rotation Second-generation ID card ID number: 330621*****037757
Blur			Binarization threshold value: 110 First-generation ID card ID number: 330226*****020013 Error, check correct: 330226*****020073

Illumination			Binarization threshold value: 120 Driver license ID number: 330823*****210047
Low-resolution Image			Binarization threshold value: 100 Second-generation ID card ID number: 350103*****033148
			Binarization threshold value: 120 Second-generation ID card ID number: 410782*****140416

## V. CONCLUSIONS

We have proposed a simple but efficient approach to recognize the ID numbers on three types of identification documents: the first generation ID card, the second-generation ID card and the driver license. We use the face features to determine the type of the ID card image, and we also use the face features to compute the rotation angle. In the stage of recognition, we use a novel method to vote the similarity about the unknown digital to each character template. Our approach is robust to the variations of illumination and rotation. The experimental results have demonstrated that the proposed approach is efficient, we tested a set of about 100 images which were shot by conventional digital cameras; for each image, the processing time is about one second on average, and the total recognition accuracy was up to 98%; Our method reached higher recognition accuracy and processing speed than the current open source OCR engines. In the future, we will further study how to analyze the other information such as the text information automatically.

## REFERENCES

- [1] Suen, C.Y., et al. "Future Challenges in Handwriting and Computer Applications," 3rd International Symposium on Handwriting and Computer Applications, Montreal, May 29, 1987.
- [2] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, 1(4), pp. 541-551, Winter 1989.
- [3] Y. LeCun, L. D. Jackel, B. Boser, J. S. Denker, H. P. Graf, I. Guyon, D. Henderson, R. E. Howard, and W. Hubbard. "Handwritten digit recognition: Applications of neural net chips and automatic learning," *IEEE Communication*, pp. 41-46, November 1989. invited paper.
- [4] Tappert, Charles C., et al. "The State of the Art in On-line Handwriting Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12 No 8, August 1990, pp. 787-808.
- [5] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Handwritten digit recognition with a backpropagation network. In David Touretzky, editor, *Advances in Neural Information Processing Systems 2 (NIPS\*89)*, Denver, CO, 1990. Morgan Kaufman.

- [6] B. Boser, E. Sackinger, J. Bromley, Y. LeCun, and L. Jackel. "An analog neural network processor with programmable topology," *IEEE Journal of Solid-State Circuits*, 26(12), pp. 2017-2025, December 1991.
- [7] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, U. A. Muller, E. Sackinger, P. Simard, and V. Vapnik. "Comparison of learning algorithms for handwritten digit recognition," In F. Fogelman and P. Gallinari, editors, *International Conference on Artificial Neural Networks*, pp. 53-60, Paris, 1995. EC2 & Cie.
- [8] L. Bottou, Y. LeCun, and Y. Bengio. "Global training of document processing systems using graph transformer networks," In *Proc. Of Computer Vision and Pattern Recognition, Puerto-Rico, 1997*. IEEE.
- [9] Y. LeCun, L. Bottou, and Y. Bengio. Reading checks with graph transformer networks. In *International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 151-154, Munich, 1997. IEEE.
- [10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, november 1998.
- [11] Nobuyuki Otsu. "A threshold selection method from gray-level histograms". *IEEE Trans. Sys., Man., Cyber.* 9, pp. 62-66.
- [12] JIA Zhen-bin, TIAN Li-yan. "Recognition algorithm using BP neural network for ID," *Journal of Suzhou Vocational University*, 17(3):2006.
- [13] Ruixiang Yin, Guohua Li. "An Automatic Recognition System for identifying ID Card Numbers," *Journal of South China University of Technology (Natural Science Edition)*, 30(2):2002.
- [14] Chi Ma, Hongyun Zhang, Duoqian Miao, Xuedong Zhang. "Improved Multi-threshold and Dynamic Binarization Algorithm," *Computer Engineering*, 32(06), pp. 203~205,2006.
- [15] Ying Shen, Junbo Fan. "Algorithm Research of Character Location in ID Card Recognition," *Computer Applications and Software*, 21(03), pp. 80~82,2004.
- [16] Zhen Zhang, Shan Huang, Daizhang Li, Guoli Yan. "A Method Applied for Precise Segmentation of the Characters in the ID Card," *Computer Engineering and Applications*, 39(13):2003.
- [17] Manhua Li, Ruixiang Yin, Chanhu Chen, "Preprocessing for ID Information Recognition," *Journal of Shantou University (Natural Science Edition)*, 18(02):2003.
- [18] <http://code.google.com/p/tesseract-ocr>.

## AUTHORS PROFILE

**Shen Lu**, born in 1986, is currently a graduate at Xiamen University. He received his B.Sc. degree from Fuzhou University in 2009. His research interests largely lie in the areas of pattern recognition, machine learning, computer vision and related areas.





**Yanyun Qu**, received her B.Sc. and M.Sc. degrees in Computational Mathematics from Xiamen University and Fudan University, China, in 1995 and 1998, respectively, and received her Ph.D. degrees in Automatic Control from Xi'an Jiaotong University, China, in 2006. She joined the faculty of Department of Computer Science in Xiamen University since 1998. She was appointed as a lecturer from 2000 to 2007 and was appointed as an associate professor since 2007. She is a member of the IEEE. Her current research interests include pattern recognition, computer vision, image/video processing, machine learning, etc.

**Yanyun Cheng**, born in 1985, is currently a graduate at Xiamen University. She was awarded a B.Sc. in Computer Science and Technology Department, Xiamen University in 2008. Her research interests largely lie in the areas of pattern recognition, machine learning, computer vision and related areas.

**Yi Xie**, received her B.Sc. and M.Sc. degrees from Xi'an Jiaotong University and received her Ph.D. degree from The Hong Kong Polytechnic University in 2008. Currently, she is an assistant professor with Department of Computer Science in Xiamen University. Her current research interests include image/video processing, system modeling, system simulation and network protocol analysis.

APPEXDIX

Image Type	Original Image	Face Region	Results	Image Type	Original Image	Face Region	Results
Uneven illumination / serious reflection			Binarization threshold value: 60 First-generation ID card ID number: 530226****11005 Error, check correct: 330226****11005	Over-exposed			Binarization threshold value: 110 Second-generation ID card ID number: 330621****037757
Seriously blurred image			Binarization threshold value: 110 First-generation ID card ID number: 330226****020013 Error, check correct: 330226****020073	Reflective			Binarization threshold value: 110 Driver license ID number: 422301****122319
Fuzzy small-angle rotation			Binarization threshold value: 70 Adjust rotation Second-generation ID card ID number: 330227****164710	Jitter			Binarization threshold value: 120 Driver license ID number: 422301****122319
Low-resolution image			Binarization threshold value: 70 Second-generation ID card ID number: 350103****033148	Special Shading			Binarization threshold value: 70 Driver license ID number: 330823****210047
Large-angle rotation			Binarization threshold value: 100 Adjust rotation Second-generation ID card ID number: 330621****037757	Darkness shadow			Binarization threshold value: 60 Second-generation ID card ID number: 330106****162712
Blurred image			Binarization threshold value: 100 Second-generation ID card ID number: 330621****037757	Darkness shadow			Binarization threshold value: 40 Driver license ID number: 330823****210047
Low-resolution / blur / jitter			Binarization threshold value: 70 Second-generation ID card ID number: 441229****123139	Special shooting angle			Binarization threshold value: 100 Second-generation ID card ID number: 330621****037757

<p>A complex background in larger proportion</p>			<p>Binarization threshold value: 110                  Second-generation ID card                  ID number: 232103*****295819</p>	<p>Low-resolution / small-angle rotation</p>			<p>Binarization threshold value: 100                  Adjust rotation                  Second-generation ID card                  ID number: 350103*****033148</p>
--	---	---	---	--	--	---	--

# Component Localization in Face Alignment

Yanyun Qu<sup>1</sup>, Tianzhu Fang<sup>2</sup>, Yanyun Cheng<sup>3</sup>, Han Liu<sup>4</sup>

Xiamen University  
Computer Science Department  
Xiamen, China

**Abstract**—Face alignment is a significant problem in the processing of face image, and Active Shape Model (ASM) is a popular technology for this problem. However, the initiation of the alignment strongly affects the performance of ASM. If the initiation of alignment is bad, the iteration of ASM optimization will be stuck in a local minima, and the alignment will fail. In this paper, we propose a novel approach to improve ASM by building the classifiers of the face components. We design the SVM classifiers for eyes, mouth and nose, and we use Speeded Up Robust Features(SURF) and Local Binary Pattern(LBP) feature to describe the components which are discriminative for the components than Haar-like features. The face components are firstly located by the classifiers and they indicate the initiation of the alignment. Our approach can make the iterations of ASM optimization converge fast and with the less errors. We evaluate our approach on the frontal views of upright faces of IMM dataset. The experimental results have shown that our approach outperforms the original ASM in terms of efficiency and accuracy.

**Keywords**—face alignment; ASM; component localization; LBP; SURF

## I. INTRODUCTION

Face alignment has been widely used in computer vision, such as object detection, tracking, alignment, and etc. Face alignment aims to deform a face model to match it with the features of the image of a face by optimizing an appropriate cost function. It is essentially an image registration problem. Face alignment is a challenging problem due to the face variation on pose, illumination and expression, as well as the occlusion. There are many works related to image alignment. Kass et al [1] proposed Active Contour Models (ACM) in 1987. Cootes and Taylor [2] proposed Active Shape Model (ASM) in 1994, which is one of the early and popular approaches that attempt to fit the model on data, and that is a generative model based on statistical computation.

In order to fit the shape to a face image robustly, many methods of face alignment combined the discriminative model with ASM [3,4,5,6,7]. These methods include three key factors: the description of the appearance shape, the design of the optimize function, and the search mechanism. In this paper, we pay attention to the search mechanism, especially to the initiation of the landmarks, because of that ASM is sensitive to the initial alignment, if the initial alignment is bad, the iteration

of ASM optimization will be stuck in a local minima, and ASM will fail. In order to obtain the good initial alignment, we combine the detections of face components with ASM, and locate the initial landmarks according to the component locations.

How to describe the face components is a critical factor for the localization of face components. Haar-like features [8] are often used to represent the face appearance, and Viola et al. [9] have demonstrated that Adaboost classifier based on the Haar-like features are successful in face detection. However, we find that Haar-like features are not powerful in representing the face components, so we use Speeded Up Robust Features (SURF) and Local Binary Pattern (LBP) feature to describe the face components, which are more distinctive than Haar-like features. One advantage of our method is that it locates the face components appropriately; the other advantage is that it makes the initiation of alignment good for ASM.

Later we use extensive experiments to show that this framework improves the robustness, accuracy and efficiency, compared with the original ASM, especially for the new data.

The rest of this paper is organized as follows. Section 2 briefly introduces the ASM. In Section 3, the description of the component detector and its experimental evaluation are provided. Section 4 describes our approach. The experiment results in Section 5 shows the advantages of our approach. In section 6, we draw the conclusions.

## II. ACTIVE SHAPE MODEL

Given a face image  $E = \{(x, y) \in R^2\}$ , the aim of face alignment is to find  $N$  landmark points to characterize it, which can be expressed as  $p_i = (x_i, y_i), i = 1, 2, \dots, N$ . In the face dataset we used, each face image is manually labeled 58 landmarks, which are distributed in the face contour, the eyebrows, the eyes, the nose and the lip. We define a feature vector for the  $i$ -th face image as  $f_i = (x_{i1}, y_{i1}, x_{i2}, y_{i2}, \dots, x_{iN}, y_{iN})^T$ . Hence, the  $M$  training images form a matrix  $F = [f_1, f_2, \dots, f_M]$ . Next, making use of principal component analysis (PCA), we can get a parametric model about the face shape, which is expressed as

$$S(b) = T(\bar{f} + \Phi b) \quad (1)$$

The research work was supported by the National Basic Research Program of China under Grant No. 2007CB311005, the Fundamental Research Funds for the Central Universities, No.2010121067, and National Defense Basic Scientific Research program of China under Grant B1420110155.

where  $b = [b_1, b_2, \dots, b_t]^T$  is the shape parameter controlling the shape change;  $\bar{f}$  is the average shape of all the training face images,

$$\bar{f} = \frac{1}{m} \sum_{i=1}^m f_i \quad (2)$$

$\Phi = (\phi_1 | \phi_2 | \dots | \phi_t)$  is the projection matrix obtained by PCA which consists of the first  $t$  eigenvectors that are corresponding to the first  $t$  eigenvalues of the covariance matrix of  $F$ . ASM synthesizes new shape by adjusting the parameter  $b$ , for the human face shape,  $b$  satisfies the constraints  $-3\sqrt{\lambda_k} \leq b_k \leq 3\sqrt{\lambda_k}$ , where  $\lambda_k$  is the  $k$ -th largest eigenvalue of the covariance matrix of  $F$ . From (1) we have,

$$b = \Phi^{-1}(T^{-1}(S(b)) - \bar{f}) \quad (3)$$

In order to find the best matching point for each landmark point, a Point Distribution Model is used to capture the shape variants in ASM, and each landmark has a point distribution model. We choose  $l$  points with an equal interval in the direction of profile at each landmark point and compute their first-order derivatives as  $g_1, g_2, \dots, g_l$ . The quality of fitting a gradient profile  $g_i$  at the location  $i$  of a query image to the  $j$ -th model is computed as the Mahalanobis distance,

$$f(g_i) = (g_i - \bar{g}_j)^T g_{cov}^{-1} (g_i - \bar{g}_j) \quad (4)$$

where  $\bar{g}_j$  is the mean of the profile in the model of the  $j$ -th landmark and  $g_{cov}$  is the covariance matrix along the profile in the model of the  $j$ -th landmark.

Given a query face image, the algorithm of ASM is shown as follows, more details see [2]:

- Step 1. Initialize with the mean shape  $\bar{S}$ .
- Step 2. Start the coarsest resolution level.
- Step 3. For each landmark, compute the Mahalanobis distance for each point at the profile, and then move landmark to the position where the Mahalanobis distance is the minimum.
- Step 4. Fit the shape model to the displaced landmarks via (3) and (1).
- Step 5. Iterate steps 3 and 4 until the process converges.
- Step 6. If the current resolution is similar to the previous, the iteration is stopped, otherwise, go to Step 3.

### III. COMPONENT DETECTOR

Major prior works of face alignment is using Haar-like features to describe the face appearance in their discriminative models. However, the Haar-like feature is not suitable for the component description. In our experiment, the detection rate of Haar-like features is as high as the error rate. Besides, the Haar-like features with the Adaboost classifier do not localize the component accurately. As shown in Fig. 1, Adaboost classifier based on Haar-like features does not localize the nose in a tight bounding box, but we need to precisely calibrate the target location just like in Fig. 1(c). Therefore, the Haar-like features do not meet our requirements. So we have to consider more discriminative descriptors just like SURF features and LBP features to describe the face components.

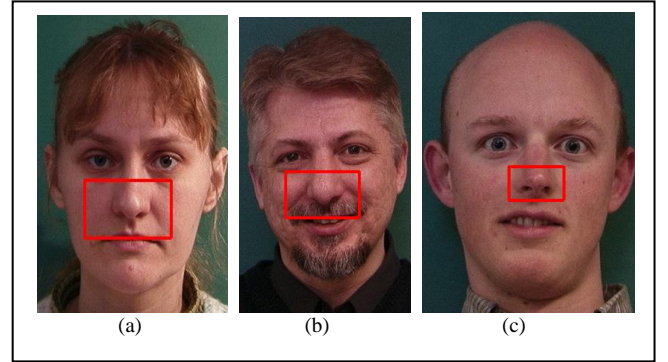


Figure 1. Some results detected by Haar-like features. (a)(b) the result of Haar-like features detection. (c) the ground truth label.

#### A. SURF

SURF is a local feature descriptor. There are two parts in SURF description: one is to detect the interesting points, and the other is to describe the interesting points by SURF. We detail it as follows.

Firstly, we randomly sample points in the component patch, and then construct a circular region around the sampled points. We compute the dominant orientation for the sampled points and describe the sampled patches by the invariant local feature. The orientation is computed using Haar wavelet, and responses in both  $x$  and  $y$  directions. The dominant orientation is estimated and included in the interest point information.

Secondly, SURF descriptors are constructed by extracting square regions around the interest points, which are oriented in the directions assigned in the previous step. The windows are split up in  $4 \times 4$  sub-regions in order to retain some spatial information. In each sub-region, Haar wavelets are extracted at regularly spaced sample points. The wavelet responses in horizontal and vertical directions ( $d_x$  and  $d_y$ ) are summed up over each sub-region. Furthermore, the absolute values  $|d_x|$  and  $|d_y|$  are summed in order to obtain information about the polarity of the image intensity changes. Hence, the underlying intensity pattern of each sub-region is described by a vector  $\mathbf{V} = [\Sigma d_x, \Sigma d_y, \Sigma |d_x|, \Sigma |d_y|]$ . Therefore, we obtain the SURF feature which is a vector of 64 dimensions. Furthermore,

we can obtain the more discriminative features if we compute the sum of  $d_x$  under the condition  $d_y \geq 0$  and  $d_y < 0$  and do the same operation for  $|d_x|$  and do the similar operation for  $d_y$  and  $|d_y|$ . This results in a descriptor vector for all  $4 \times 4$  sub-regions of 128 dimensions. Fig. 2 shows a simple process for SURF. See more details in [10].

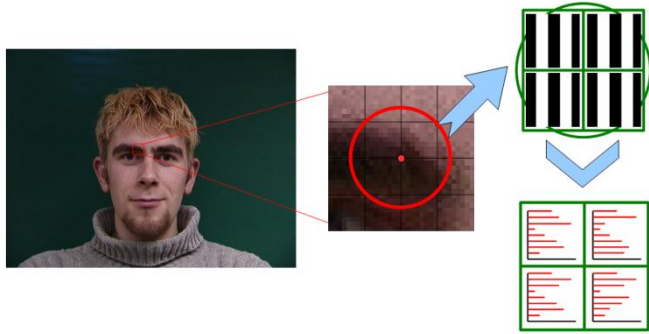


Figure 2. A simple process for SURF.

Finally, we cluster the feature vectors through k-means algorithm, thus the clustering center could be regarded as a word. We represent the vectors of these component patches as histograms by the histograms of words, and use SVM in [11] for training. In this way, we get a SVM classifier based on the SURF features.

### B. Local Binary Pattern

The Local Binary Pattern (LBP) operator was introduced by Ojala et al [12]. The operator labels the pixels of an image by thresholding a  $3 \times 3$  neighborhood of each pixel with the center value and considering the results as a binary number. The 256-bin histogram of the labels computed over a region can be used as a texture descriptor. Each bin can be regarded as a micro-texture. Later the operator was extended to use neighborhood of different sizes using circular neighborhoods [13].

The  $LBP_{P,R}$  operator produces  $2^P$  different output values, corresponding to the different binary patterns that can be formed by the P pixels in the neighbor set. It has been shown that certain bins contain more information than others [13]. Ojala et al. called these fundamental patterns uniform patterns. Fig. 3 shows an example for LBP.

After representing the face components by LBP features, we also carry on SVM [11] to train them, thus obtain a classifier.

### C. Evaluation of Component Recognition

We compared SURF and LBP with Haar-like features on the performance of the component recognition.

We use the Adaboost classifier to detect face components, which are described by Haar-like feature. As comparison, we apply the SVM classifier with Gaussian kernel to detect eyes and noses which are described by LBP feature, besides, we also use the SVM classifier to detect eyes and mouths which are described by SURF.

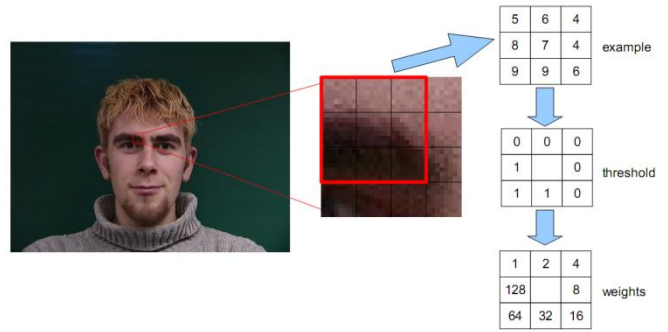


Figure 3. An example for LBP.

As shown in Table I, SURF is not sensitive to the noses features (possibly due to noses were at little difference with the surrounding texture and the edge features were relatively few), and LBP is not sensitive to the lip features (mainly due to the beard shading and the teeth impact), they still have the satisfying results: for nose, the error rate by LBP classifier is much smaller than the one by Haar-like classifier, and for mouth, the error rate by SURF classifier is much smaller than the one by Haar-like classifier, and for eyes the error rates by both the two mentioned classifier are much smaller than Haar-like classifier. Generally speaking, SURF and LBP are more distinctive than Haar-like features in describing components. And we do not show the detection rate of mouths by LBP classifier and that of noses by SURF classifier, due to their low detection rate.

TABLE I. THE DETECTION RATES OF SOME CLASSIFIERS

Component		Eyes	Nose	Mouth
Haar-like	Error Rate	24.17%	100%	100%
	Miss Rate	2.08%	15.83%	0%
LBP	Error Rate	1.6%	2.96%	/
	Miss Rate	43.75%	6.67%	
SURF	Error Rate	8.33%	/	12.5%
	Miss Rate	4.58%		13.75%

## IV. ALGORITHM OF FACE ALIGNMENT

According to the description in Section II, the original ASM can be described as

$$S_0 = \bar{S};$$

$$S_{t+1} = \{g' \mid f(g') = \min f(g_i), g_i \in \{\text{profiles in points } g\}, g \in S_t\}$$

where  $f(g_i) = (g_i - \bar{g})^T g_{Cov}^{-1} (g_i - \bar{g})$  and  $S_t$  denotes the shape state of the t-th time.

Face alignment is sensitive to the initial shape  $S_0$ . And all points in  $S_t$  are searched by adjusting  $S_0$ . If the initial landmarks are put in the suitable localization, the shape would aim at the initial points localization through component recognition. The flow chart of our approach is shown in Fig. 4.

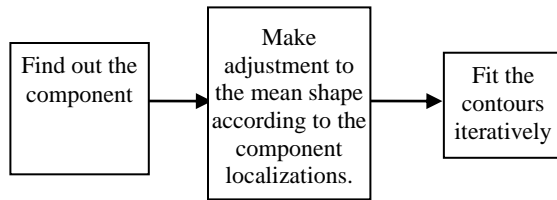


Figure 4. The flow chart of our approach.

### A. Constructing the training dataset

As we observe, the important components of a face are the eyes, the mouth and the nose. So we train the three component detectors.

We crop the component patches from the IMM dataset as the positive samples and cropped other patches as negative samples. In the eyes training set, there are 37 positive samples and 178 negative samples; in the nose training set, there are 37 positive samples and 124 negative samples; and in the mouth training set, there are 37 positive samples and 88 negative samples. Some examples are given in Fig. 5 and Fig. 6.

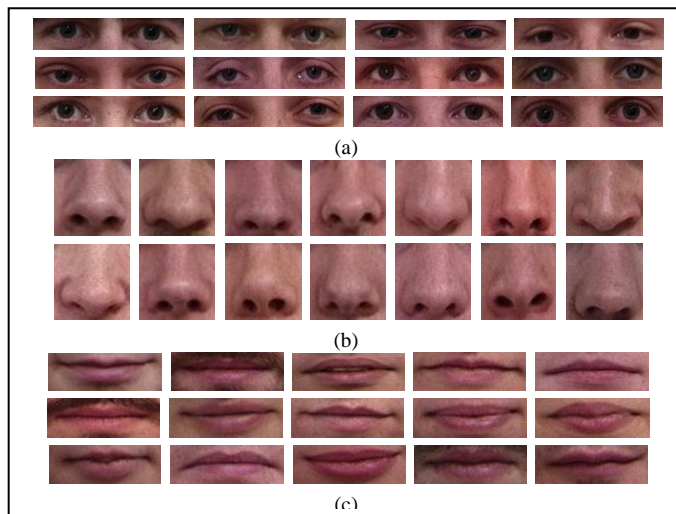


Figure 5. Some positive examples used in component detection. (a) the samples for eyes, (b) the samples for noses, (c) the samples for mouths.



Figure 6. Some negative examples used in component detection. (a) the negative samples for the eyes training, (b) the negative samples for the nose training, (c) the negative samples for the mouth training.

### B. ASM Algorithm

#### 1) Training:

- Step 1: Select  $m$  face images which have been annotated manually using 58 landmarks around the eyebrows, the eyes, the nose, the mouth and the jaw, and then build the shape model:

$$S(b) = T(\bar{f} + \Phi b)$$

- Step 2: For each landmark, generate sample  $l$  points along the profile of the landmark, and build the gray-level appearance model.
- Step 3: Train the three classifiers for the eyes, the nose and the mouth.

#### 2) Fitting:

- Step 1: Detect the three components mentioned above and put the initial landmarks in suitable points, and adjust the mean shape.
- Step 2: Make the adjusted mean shape as iterative starting shape.
- Step 3: For each landmark, computer the Mahalanobis distance and find the position which has minimum Mahalanobis distance.
- Step 4: Fit the shape model to displaced landmarks via (3) and (1).
- Step 5: Iterate steps 3 and 4 until the solution is convergence.
- Step 6: If the current resolution is similar to the previous, the iteration terminate; otherwise, goto Step 3.



## V. EXPERIMENTAL RESULTS

We implement our method on IMM dataset with PIII1.8GHZ processor and 256MB memory. We used 40 upright faces in the dataset for training and the others 80 upright faces for testing.

In addition, we use two criteria to evaluate our approach, the average frequency of convergence (AFC) given by the number of trials where the alignment converges divided by the total number of trials and the mean square error (RMSE). And the AFC is defined as

$$c = \sum_{i=1}^n c_i / n$$

where  $c_i$  denotes the convergence frequencies of the  $i$ -th image.

The RMSE is defined as

$$Err = \sqrt{\frac{\sum_{i=1}^n ((x'_i - x_i)^2 + (y'_i - y_i)^2)}{n - 1}}$$

where  $\{(x'_i, y'_i)\}$  denotes the shape computed by our approach and  $\{(x_i, y_i)\}$  denotes the ground truth shape.

In this paper, we have 58 landmark points for each face image, that is,  $n = 58$ . We considered two factors which influence our approach performance: the number of the face components and the descriptor. We compared the five methods: (1) the original ASM, (2) the eye detector with LBP feature combined with ASM, (3) the detectors for the eyes and the nose based on LBP feature combined with ASM, (4) the eye detector with SURF combined with ASM, (5) the detectors for the eyes and the nose with SURF combined with ASM. The result is shown in Table II. It demonstrates that the component detection can improve the AFC and RMSE of ASM, and the more components are combined with ASM, the better accuracy the ASM is. Furthermore, the components detector with SURF is superior to those with LBP in the fitting accuracy.

Fig. 7 and Fig. 8 show some results in or outside the training set. The results demonstrate that our approach is stable and can improve the accuracy of shape fitting in ASM.

However, we also have some failed cases, just as shown in Fig. 9. We analyze the failure cases and find that the faces have large rotation in the failure cases, while our approach is only suitable to the upright faces.

TABLE II. THE GENERAL SITUATION OF THE EXPERIMENT

METHOD	AFC	RMSE
ASM	1150.158	44.83577
Eyes + LBP	842.5444	9.600739
Eyes + Nose + LBP	935.5402	9.30989
Eyes + SURF	1044.188	11.4465
Eyes + Nose + SURF	1068.383	9.115111

## VI. CONCLUSIONS

In this paper, we have proposed an improved approach, which combines the component learning with ASM for face alignment. The experiment results on IMM dataset have shown that component localization speeds up the convergence of ASM iterations, and the results in matching the face shape is more accurate than the original ASM. In the future, we will apply the symmetrical information in the face alignment, which may reduce the time complexity of ASM and improve the results fitting the template on the face image.

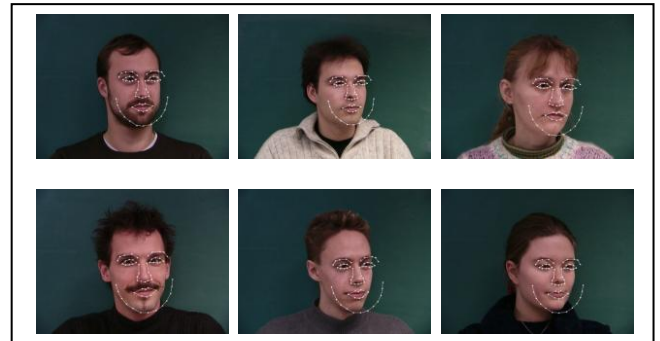
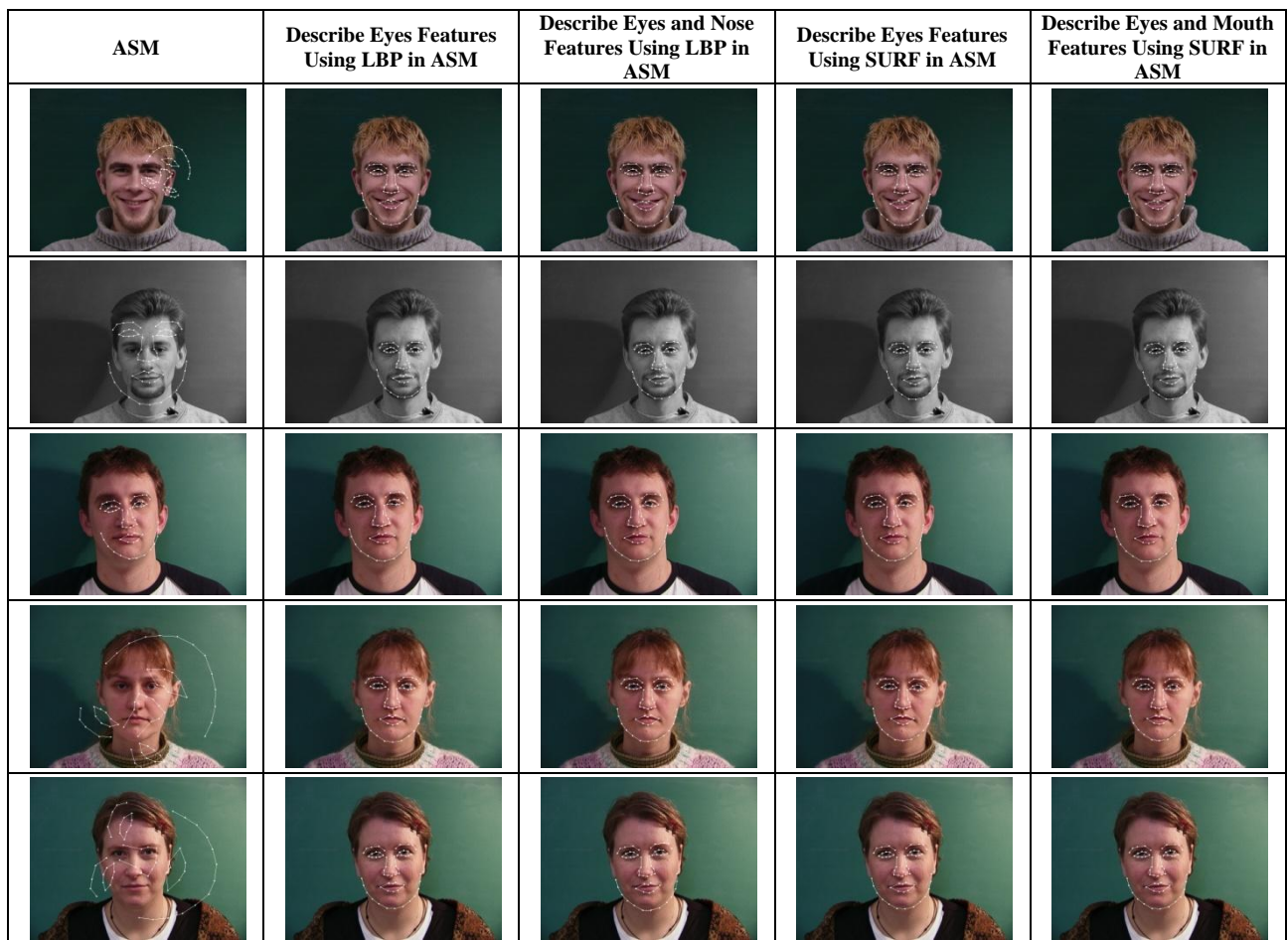


Figure 9. Some failure examples.



Figure 7. Some examples on the training set based on component localization combined with ASM vs original ASM.



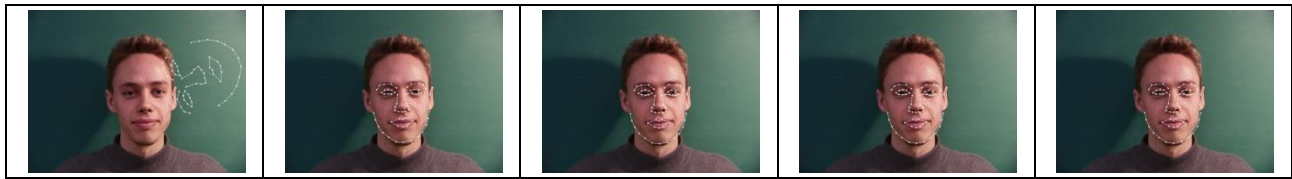


Figure 8. Some examples in the testing set based on component localization combined with ASM vs original ASM.

## REFERENCES

- [1] M. Kass, A. Witkin and D. Terzopoulos. "Active Contour Models," The 1st ICCV, London, UK, 1987.
- [2] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. "Active Shape Models – Their Training and Application," Computer Vision and Image Understanding, 1995, 61(1): 38-59.
- [3] S.C. Yan, C.Liu, S.Z.Li, H.J.Zhang, H. Shum, Q.S.Cheng. "Face Alignment Using Texture-Constrained Active Shape Models," Image and Vision Computing, 2003, 21(1): 69-75.
- [4] B.V. Ginneken, A.F. Frangi et al. "A Non-linear Gray-level Appearance Model Improves Active Shape Model Segmentation," IEEE Workshop on Math Models in Biomedicine Hawaii, USA, 2001.
- [5] C Du, Q Wu, J Yang, Z Wu. "SVM based ASM for facial landmarks location," IEEE 8th International Conference on Computer and Information Technology, 2008.
- [6] Y. Li and W. Ito. "Shape parameter optimization for adaboosted active shape model," In Proc. 10th ICCV, 2005.
- [7] L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component based discriminative search. In ECCV, 2008.
- [8] Papageorgiou, Oren and Poggio, "A general framework for object detection", International Conference on Computer Vision, 1998.
- [9] Viola and Jones, "Rapid object detection using boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001.
- [10] H. Bay, T. Tuytelaars, and L. Van Gool. "Surf: Speeded up robust features," The 9th European Conference on Computer Vision, 2006.
- [11] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM – A Library for Support Vector Machines," <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>.
- [12] T. Ojala, M. Pietikinen, and D. Harwood. "A comparative study of texture measures with classification based on featured distribution," Pattern Recognition, 1996, 29(1):51–59.
- [13] T. Ojala, M. Pietikinen, and T. Menp. "Multiresolution grayscale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7):971–987.

## AUTHORS PROFILE

**Yanyun Qu**, received her B.Sc. and M.Sc. degrees in Computational Mathematics from Xiamen University and Fudan University, China, in 1995 and 1998, respectively, and received her Ph.D. degrees in Automatic Control from Xi'an Jiaotong University, China, in 2006. She joined the faculty of Department of Computer Science in Xiamen University since 1998. She was appointed as a lecturer from 2000 to 2007 and was appointed as an associate professor since 2007. She is a member of the IEEE. Her current research interests include pattern recognition, computer vision, image/video processing, machine learning, etc.

**Tianzhu Fang**, born in 1987, is currently a graduate at Xiamen University. He received his B.Sc. degree in Software Engineering from Fuzhou University in 2009. His research interests largely lie in the areas of pattern recognition, machine learning, computer vision and related areas.

**Yanyun Cheng**, born in 1985, is currently a graduate at Xiamen University. She was awarded a B.Sc. in Computer Science and Technology Department, Xiamen University in 2008. Her research interests largely lie in the areas of pattern recognition, machine learning, computer vision and related areas.

**Han Liu**, born in 1985, is currently a graduate at Xiamen University. He was awarded a B.Sc. in Mathematical Science, Xiamen University in 2008. His research interests largely lie in the areas of pattern recognition, machine learning, computer vision and related areas.

# Fine Facet Digital Watermark (FFDW) Mining From The Color Image Using Neural Networks

N.Chenthalir Indra

Assistant Professor, Computer science Department,  
S.T.Hindu College, Nagercoi, Tamilnadu, India. 629001.

Dr. E . Ramaraj

Technology Advisor, M.K. University  
Madurai, Tamilnadu,India.

**Abstract—** On hand watermark methods employ selective Neural Network techniques for watermark embedding efficiently. Similarity Based Superior Self Organizing Maps (SBS\_SOM) a neural network algorithm for watermark generation. Host image is learned by the SBS\_SOM neurons and the very fine RGB feature values are mined as digital watermark. Discrete Wavelet Transform (DWT) is used for watermark entrench. Similarity Ratio and PSNR values prove the temperament of the Fine Facet Digital Watermark (FFDW). The Proposed system affords inclusive digital watermarking system.

**Keywords-** Similarity based Superior SOM; Discrete Wavelet Transform; Digital watermark; embedding; PSNR.

## I. INTRODUCTION

Digital image watermarking is a technique which embeds additional information called digital signature or watermark into the digital content in order to secure it [10]. A watermarking system is usually divided into four distinct steps, collecting digital watermark, embedding, attack and detection. Digital watermark may be any text, image, signals or any derived values. The proposed system generates watermark from host image by using Similarity Based Superior Self Organizing Maps (SBS\_SOM) a neural network algorithm. In embedding, an algorithm accepts the host and the data to be embedded and produces a watermarked signal. The watermarked signal is then transmitted or stored.

If anybody makes a modification, this is called an attack. While the modification may not be malicious, the term attack arises from copyright protection application, where pirates attempt to remove the digital watermark through modification. There are many possible modifications, for example, lossy compression of the data, cropping an image or intentionally adding noise. This analysis verifies the robustness of watermark by smooth JPEG compression and adding standard noise. Detection (often called extraction) is an algorithm which is applied to the attacked signal to attempt to extract the watermark from it. In robust watermarking applications, the extraction algorithm should be able to produce the watermark again, even if the modifications were strong. In fragile watermarking, the extraction algorithm should fail if any change is made to the signal. Reference [8] uses BPN (Back Propagation Network) model to learn the relationship between the watermark and the watermarked image. Reference [5] used a full counter-propagation neural network (FCNN) for copyright protection where the ownership information was embedded and detected by a specific FCNN. Reference [1]

proposed a new blind watermarking scheme in which a watermark was embedded into the DWT (Discrete Wavelet Transform) domain. It also utilized RBF Neural network to learn the characteristic of the image, using which the watermark would be embedded and extracted. Reference [4] presented a specific designed full counter-propagation neural network for digital image watermarking. Most of the systems used CPN, BPN and RBF algorithms. Various neural network algorithms were used to strengthen and retrieve the watermark values, *but not for watermark value generation*. The proposed system SBS\_SOM a neural network algorithm was trained to generate digital watermark values from the image. The proposed system submits robust watermarking scheme. The section two gives detail about basic techniques Similarity based superior SOM which is used for watermark generation and detection and Discrete Wavelet Technique (DWT) for embedding. Section three describes the proposed system process. The experimental results are discussed in the section four. Section five summarizes the result.

## II. PROPOSED SYSTEM ENVIRONMENT

Instead of using standard media as digital watermarks or mathematically derived watermarks the projected system imply a neural network algorithm called Self Organizing Maps (SOM). The SOM is a particular type of neural network used in clustering, visualization and abstraction [6]. It is an unsupervised competitive learner. Its learning pattern and order are unpredictable.

Existing applications on SOM express the benefits of using the SOM in applications with massive sets of data in finance, macroeconomics, medicine, biology, and other fields. The Self-Organizing Maps have been used at the Research Center in such applications as: Automatic speech recognition, Clinical voice analysis, Monitoring of the condition of industrial plants, processes Cloud classification from satellite images, analysis of electrical signals from the brain organization and retrieval of data from large document collections. The outputs of SOM applications are highly visual, which assists the analyst in understanding the data's internal relationships.

However in the field of watermarking SOM was used for the process of embedding. The paper recommends the improvised SOM known as Similarity Based Superior SOM (SBS\_SOM) for watermark generation. Conventional SOM and most of the extended SOM are using Euclidean-based distance metrics. The proposed SBS\_SOM is more efficient

than the conventional SOM. It uses Jaccard or Dice measures for winner node selection. So the training starts from maximum distance nodes, whereas SOM starts its training from minimum distance nodes. Its factors such as weight initialization, learning rate, training epochs and neighborhood function are influencing the training nature of the network. These settings are standardized as per the rules. The same environment is created with conventional SOM for the analysis intention.

#### A. Similarity Based Superior SOM

References [2] and [3] reliably recommend Similarity based Self Organizing Maps neural network for image training. The Similarity based Superior Self Organizing Maps is the focal process to generate watermark. Significant initial factors are set by authorized person. Without knowing those values detection of watermark is impossible.

##### 1) Algorithm:

**Step0:** Initialize weights with random method or by having previous knowledge of Pattern distribution. Set Topological neighborhood parameters. Set learning rate parameter

**Step1:** While stopping condition is false, do steps 2 – 8

**Step2:** For each input vector  $\mathbf{x}$ , do steps 3– 5

**Step3:** For each  $\mathbf{j}$ , compute  $d(\mathbf{j})$  by using any one of the following distance measures as in (1) or (2).

a) *Jaccard Coefficient:*

$$\text{sim}(x_i, w_j) = \frac{\sum_{h=1}^k x_{ih} w_{jh}}{\sum_{h=1}^k x_{ih}^2 + \sum_{h=1}^k w_{jh}^2 - \sum_{h=1}^k x_{ih} w_{jh}} \quad (1)$$

b) *Dice Coefficient:*

$$\text{sim}(x_i, w_j) = \frac{2 \sum_{h=1}^k x_{ih} w_{jh}}{\sum_{h=1}^k x_{ih}^2 + \sum_{h=1}^k w_{jh}^2} \quad (2)$$

**Step4:** Find index  $\mathbf{J}$  such that  $\mathbf{d}(\mathbf{j})$  is a minimum.

**Step5:** For units  $\mathbf{j}$  within a specified neighborhood of  $\mathbf{J}$  and for all  $\mathbf{I}$   $W_{ij}(\text{new}) = W_{ij}(\text{old}) + \alpha [X_i - W_{ij}(\text{old})]$

**Step6:** Update learning rate

**Step7:** Reduce radius of topological at specified times

**Step8:** Test stopping condition.

The Learning Rate is a slowly decreasing function of time. The radius of the neighborhood around a cluster unit also decreases as the clustering process progresses. The updated weight network is well equipped with the host image neural structure. SBS\_SOM exactly imitates human neural learning logic. Hence trivial imbalanced values can be identified through the analysis. These insignificant map elements in SBS\_SOM network is determined and used as watermark values.

#### B. Discrete Wavelet Transformation(DWT)

The first Discrete Wavelet Transform (DWT) was invented by the Hungarian mathematician Alfred Haar. For an input

represented by a list of  $2^n$  numbers, the **Haar wavelet** transform may be considered to simply pair up input values, storing the difference and passing the sum. This process is repeated recursively, pairing up the sums to provide the next scale: finally resulting in  $2^n - 1$  differences and one final sum.

DWT decomposes input image into four components namely LL, HL, LH and HH. The lowest resolution level LL consists of the approximation part of the original image. The remaining three resolution levels consist of the detail parts and give the vertical high (LH), horizontal high (HL) and high (HH) frequencies. In the proposed technique, embedding and extraction of watermark takes place in the high frequency component. For a one level decomposition, the discrete two-dimensional wavelet transform of the image function  $f(x, y)$  is found in [7] and [9]. DWT transform based watermarking scheme is robust against many common image attacks. The analysis results proved robustness very well.

### III. FINE FACET DIGITAL WATERMARK SYSTEM (FFDW)

In the proposed system host image is learned by the SBS\_SOM neurons and the very fine RGB feature values are mined as digital watermark. Discrete Wavelet Transform (DWT) is used for watermark entrench.

#### A. FFDW Generator

##### 1) Preprocessing:

- 1) Collect the input images. Select the image for watermarking.
- 2) Extract its RGB colors attributes in separate 2-D spaces.

##### 2) Apply SBS-SOM:

- 1) Set three SBS\_SOM network for representing Red, Green and Blue with three layers. (Input, hidden and output with two dimensional space matrix).
- 2) Initialize its weight vectors, neighborhood function, fix the epochs, set initial learning rate and fix reduction of learning rate for each epoch.
- 3) Train the RGB networks. Find the trained feature maps from output layer.
- 4) Find the difference between input values and trained values for each color. The resultant values are accepted as watermark values. Thus the first pass obtains three sets of watermarks.

##### 3) Embedding:

- 1) Activate one-level DWT to original image's red vectors.
- 2) The red attribute watermark is embedded in to the high frequency component HH of DWT.
- 3) Execute inverse wavelet transform to obtain the watermarked red features.
- 4) Repeat the above three steps for other two green and blue colors too.

5) By combining RGB watermarked plane the watermarked image will be obtained.

**B. FFDW Detector**

Projected watermarking proposal is capable of mine watermark information in the absence of the original image or secrete key. Hence it is unsighted watermarking.

- 1) Trigger one level DWT to the destination image and take away the embedded watermark from the HH sub band.
- 2) Regenerate watermark from the transferred image by using SBS\_SOM neural logic as mentioned in the generation algorithm.

**IV. PROFICIENCY ANALYSIS ON FFDW**

In order to prove the efficiency of SBS\_SOM mined FFDW, the quality assessment experiments were done. It is proved that the proposed watermark technique has robustness, imperceptibility and authenticity.

For the experimentation, the digital watermark values with 64x64 size was generated from the host image which is be watermarked. Since the FFDW is mined through SBS\_SOM neural network for each individual host image, it is unique. No one can predict the watermark values by means of common calculations. Minute details having inaptness were identified and collected as digital watermark. The RGB colors are trained individually. Hence the result organized three sets of digital watermarks.

Each 2-D plans are embedded by means of corresponding color FFDW values by using DWT. Subsequent to the embedding process the three 2-D planes are combined to form 3-D watermarked image. The Peak Signal to Noise Ratio (PSNR) as in (3), and Similarity Ratio (SR) as in (1), are estimated between host image and watermarked image. Besides, the same process was done with conventional SOM to compare the efficiency of the proposed SBS\_SOM for watermarking. The experiment was carried out with the sample image gallery. The ten image samples and its corresponding watermarked image PSNR and SR values are tabularized in the Table 1.

$$PSNR = 10 \log_{10} \left( \frac{255^2}{MSE} \right) \quad (3)$$

TABLE 1. PSNR & SR OF 10 SAMPLE HOST IMAGES

Host images	SBS_SOM		Conventional SOM	
	PSNR(dB)	SR	PSNR(dB)	SR
Cap girl	53.7	1	32.4	0.88
Blue boy	52.5	1	32.8	0.84
Chubby babby	52.3	1	32.5	0.84
Cute baby	50.3	1	30.5	0.83
Mary	52.1	1	33.1	0.84
Green boy	52.5	1	32.2	0.83
softy	53.9	1	31.2	0.82

Been	52.6	1	32.1	0.85
Infant	51.7	1	33.2	0.85
Smiley	55.6	1	30.5	0.81

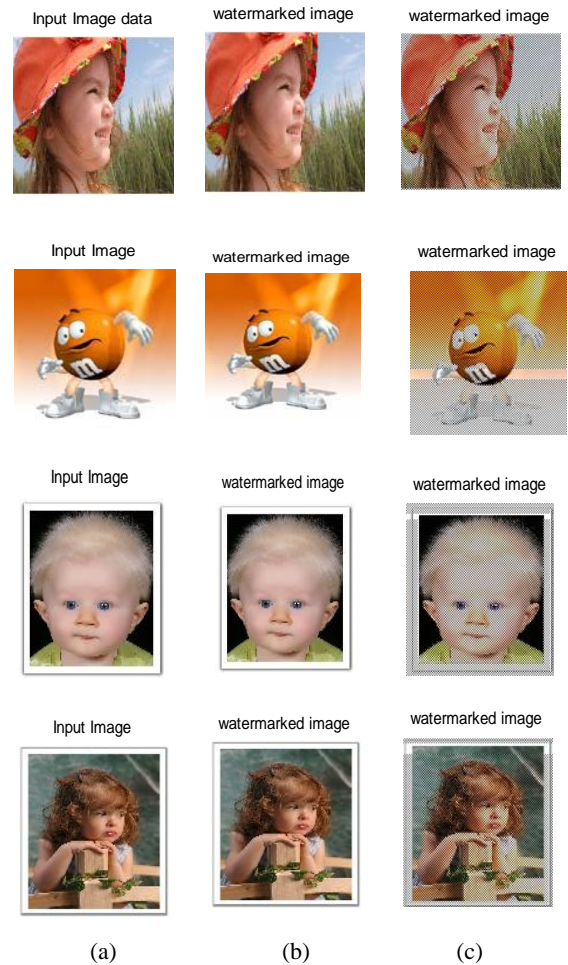


Figure 1. (a) Input images (b) SBS\_SOM watermarked images (c) SOM watermarked images

Among those watermarking process four different color host images and its related watermarked images are given in the Fig. 1 (b) with SBS\_SOM high level imperceptibility is proved. Visibly there is no degradation of image was found in the watermarked image. From the Table 1, it was proved that host image and watermarked image has fine similarity ratio '1'. Well decent PSNR also found in proposed watermarking technique. Compressed and transferred watermarked image without any attack at the destination system receives imperceptible watermarked images. On the other hand SOM watermarking shows visible changes in the watermarked images. Fig. 1(c) column is the evident for SOM watermarking.

The robustness of the watermarking is checked by applying special kinds of noise like Gaussian, Poisson, Salt & Pepper and Speckle. The attacked image of cap girl is shown in the Fig. 2. The watermarks are detected and extracted from the transferred watermarked image. The extracted watermark is

compared with the original watermark by deducing PSNR value. The cap girl image watermark PSNR values with various attacks are mentioned in the Table 2.

TABLE 2. ROBUSTNESS OF WATERMARK

Types of Attacks	SBS-SOM	SOM
	PSNR(dB)	PSNR(dB)
No Attack	51.9	33
Compressed	51.9	32
Gaussian	51.2	32.1
Poisson	51.6	33.2
Salt & Pepper	51.4	32.4
Speckle	51.3	31.7
Damaged	51.8	30.4

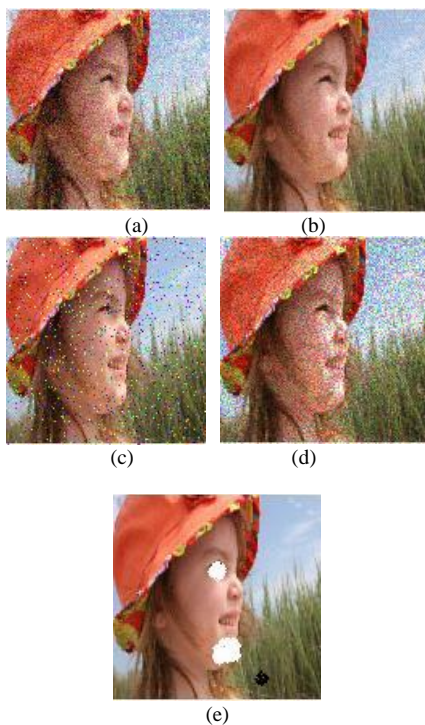


Figure 2. Noisy watermarked images (a) Gaussian noise (b) Poissian noise (c) Salt & Pepper noise (d) Speckle noise (e) Damaged

One and the same results were evaluated in the case of other images too. The reasonable PSNR value even after noise attacks in the case of SBS\_SOM proves the robustness of the watermarking. At the receiver end the attacked image can be strongly identified. For this the watermark is detected by authorized method and removed from the received image to get the original image back. Watermark removed image and host image PSNR and SR are calculated and tabulated in the Table 3. The Table 3 clearly states that received watermarked image without any attack will produce exact host image values. The PSNR is at its maximum infinitive. This means that there is no difference between the values of host and watermark removed image at the receiver side.

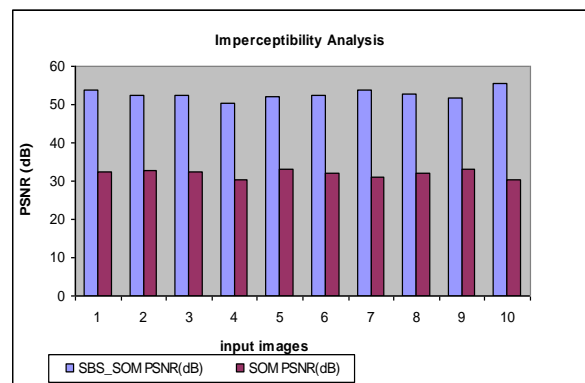
At the receiver end the attacked image can be strongly identified. For this the watermark is detected by authorized method and removed from the received image to get the original image back. Watermark removed image and host image PSNR and SR are calculated and tabulated in the Table 3. It clearly states that received watermarked image without any attack will produce exact host image values. The PSNR is at its maximum infinitive. This means that there is no difference between the values of host and watermark removed image at the receiver side.

TABLE 3. QUALITY OF AN IMAGE AFTER THE REMOVAL OF WATERMARK

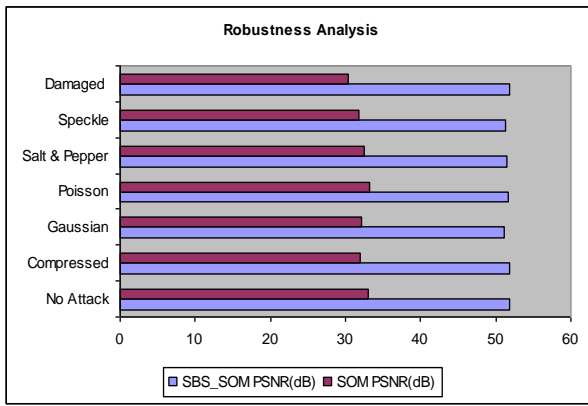
Attacks	SBS_SOM		SOM	
	PSNR (dB)	SR	PSNR (dB)	SR
No attack	<b>INF</b>	<b>1</b>	35.2	0.87
Compressed	46.7	0.999	34.6	0.81
Gaussian	38.9	0.979	34.2	0.80
Poisson	41.6	0.983	34.4	0.78
Salt & Pepper	39.1	0.970	33.8	0.82
Speckle	38	0.972	34.3	0.79
Damaged	42	0.981	33.5	0.82

FFDW system does not affect the originality of the host image. And also the watermark is not detectable because the FFDW system uses unsupervised SBS\_SOM for watermark generation. This neural network is unpredictable and the almost similar results can be evaluated only through the known parameter settings. Each and every parameter settings have their own authority on neural node organization. Hence using common statistical analysis or correlative attacks will never be able to find the watermark. Without proper watermark detection the original image cannot be restructured. Since FFDW system assures for secured watermarking.

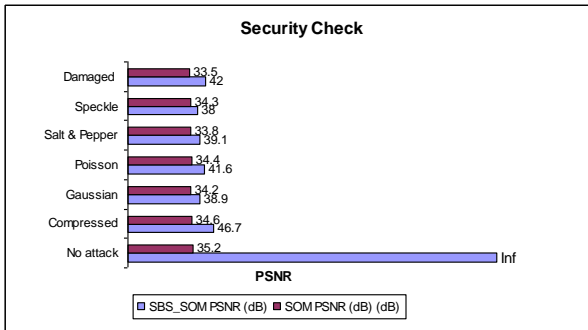
The attacked images had lost their originality. Even though the watermark is robust the original image was not reconstructed. The Table 3 flaunts the trim down of PSNR and SR. For the copy right claim and authentication applications, the *infinitive* PSNR values are needed. FFDW system can be strongly proposed for authentic applications. The Fig. 3 visibly presents the secured image transformation with its self-generated digital watermarking quality assurance.



(a)



(b)



(c)

Figure 3. Fig. 4.3 (a) Imperceptibility Analysis of watermarked image (b) Robustness verification of watermark (c) Security Check of watermark removed image at the destination system

The robustness, imperceptibility and securities were analyzed and plotted as a graph to explicitly show the good quality of FFDW system with SBS\_SOM. The Fig. 3 exhibits the significance of proposed watermarking. SBS\_SOM based FFDW system in watermark generation is one more land mark in watermarking techniques.

## V. CONCLUSION

SBS\_SOM mined Fine Facet Digital Watermark system is an innovative technique for watermarking research. The other image, text or audio digital values are not used as digital watermark to embed. The purpose of embedding values in this

process is not protecting embedding values but identifying the misuse of the host image. The watermark of each individual image is mined from the corresponding host image. Hence collecting watermark values from other image is impossible. Unauthorized users cannot detect the watermark values from watermarked image by using any statistical or correlation calculation because SBS\_SOM is a complete unsupervised network trainer. FFDW is a good digital watermarking technique because it satisfies the basic requirements such as Robustness, Imperceptibility and Security. The image communication to claim authentication may use the proposed system confidently. The proposed system used DWT for the watermark embedding. In future the SBS\_SOM itself will be used for embedding to prove its comprehensiveness.

## REFERENCES

- [1] Cheng-Ri Piao, Suenghwa Beack, Dong-Min Woo and Seung-Soo Han, "A Blind Watermarking Algorithm based on HVS and RBF Neural Network for Digital Image", ICNC 2006, Part 1, LNCS 4221, pp. 493-496, 2006.
- [2] Chenthalir Indra N, Ramaraj E, "Similar - Dissimilar Vector Measure Analysis to Improve Image Knowledge Discovery Capacity of SOM". In: Communications in Computer and Information Science, 2010, Volume 101, Part 2, 389-393, DOI: 10.1007/978-3-642-15766-0\_612.
- [3] Chenthalir Indra N, Ramaraj E, "Magnitude of Self Systematizing Resemblance Measures in Knowledge Mining". In: Software Technology and Engineering, 2009, pp 239-243, DOI : 10.1142 / 9789814289986\_0044.
- [4] Chuan-Yu Chang and Sheng-Jyun Su, "The application of a Full Counterpropagation Neural Network to Image Watermarking", 2005.
- [5] Chuan-Yu Cahng, Hung-Jen Wang, Sheng-Jyun Su, "Copyright authentication for images with a full counter propagation neural network", Expert Systems with Applications 37, 2010.
- [6] Dorina Marghescu, Mikko J. Rajanen - "Assessing the Use of the SOM technique in Data Mining" - Proceedings of the 23rd International Multi-Conference on Databases and Applications, February 14-16, 2005, Innsburck, Austria.
- [7] Kumar, S., Raman, B., Thakur, M.: "Real Coded Genetic Algorithm based Stereo image Watermarking". In: IJSDIA 1(1), pp.23-33 (2009).
- [8] Jun Zhang, Nencho Wang, Feng Xiong, "Hiding a Logo Watermark into the Multiwavelet Domain using Neural Networks", In the Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence, 2002.
- [9] Sujatha S S and Mohamed Sathik M "Feature Based Watermarking Algorithm by Adopting Arnold Transform". In: ICT 2010, CCIS 101, pp. 78-82, 2010. © Springer-Verlag Berlin Heidelberg 2010.
- [10] Sujatha S S and Mohamed M Sathik, "A Novel Feature Based Blind Watermarking Scheme". In: CiiT International Journal of Digital Image Processing, Vol 2, No 7, pp.185-188, July 2010.



# Automatic Image Registration Using Mexican Hat Wavelet, Invariant Moment, and Radon Transform

Jignesh N Sarvaiya  
Electronics Engg. Dept  
S V National Institute of Technology  
Surat, India

Dr. Suprava Patnaik  
Electronics Engg. Dept  
S V National Institute of Technology  
Surat, India

**Abstract**— Image registration is an important and fundamental task in image processing used to match two different images. Given two or more different images to be registered, image registration estimates the parameters of the geometrical transformation model that maps the sensed images back to its reference image. A feature-based approach to automated image-to-image registration is presented. The characteristic of this approach is that it combines Mexican-Hat Wavelet, Invariant Moments and Radon Transform. Feature Points from both images are extracted using Mexican-Hat Wavelet and control-point correspondence is achieved with invariant moments. After detecting corresponding control points from reference and sensed images, to recover scaling and rotation a line and triangle is form in both images respectively and applied radon transform to register images.

**Keywords**-Image Registration; Mexican-hat wavelet; Invariant Moments; Radon Transform.

## I. INTRODUCTION

In this paper, we propose and implement a novel image registration method to register image with large rotation, scaling and translation. In all types of image registration, robustness of the algorithm is the main and required goal. However, due to diversification of images acquired their contents and purpose of their alignment, it is almost impossible to design universal method for image registration that fulfill all requirements and suits all types of applications [2]. Many of the image registration techniques have been proposed and reviewed [1], [2] [3]. Image registration techniques can be generally classified in two categories. Intensity based and feature based. The first category utilizes image intensity to estimate the parameters of a transformation between two images using an approach involving all pixels of the image. In second category a set of feature points extracted from an image and utilizes only these extracted feature points instead of all whole image pixels to obtain the transformation parameters. In this paper, a new algorithm for feature based image registration is proposed. The proposed algorithm is based on three main steps, feature point extraction, correspondence between extracted feature points and transformation parameter estimation.

Feature point and corner detectors have been area of interest to researcher in image registration. A point in an

image is considered as a feature point if it has properties of distinctiveness and invariance. Several approach have been developed for feature point extraction and the points extracted by these methods differ in locations and structure, for example, edges, corner, blob-like shape, etc. In general, the main objective is to develop a feature point extractor that is robust to the most common geometric transformations and any possible degradation. Bas et al [4] compared major feature extraction techniques: Harris Corner detector, the Susan corner detector, and Achard-Rouquet detector. Feature extraction using Gabor wavelets have presented in [5], [6] give some good results in robustness to face recognition application. A popular method in computer vision applications is the scale invariant feature transform (SIFT) which is based on the feature point detection in scale space [7]. Recently developed technique which is based on scale interaction of Mexican hat wavelets [8]. This Mexican hat wavelet based feature extraction was inspired by the method presented in [8] [14].

We have used Mexican Hat Wavelet (MHW), to extract significant image features from reference and sensed image and make two sets of extracted feature points for both images. Next step of correspondence between extracted feature points is performed using Hu's Invariant moment-based similarity measure. This correspondence is evaluated using a circular neighborhood centered on each feature point. Among various types of moments available, Hu's moments is superior in terms of their orthogonality, rotation invariance, low sensitivity to image noise, fast computation and ability to provide faithful image representation [25]. After detecting matched feature points (control points), radon transform is used to estimate scaling and rotation. Radon transform gives  $R(\theta, x')$  matrix from  $f(x, y)$  matrix. The scaling factor has been obtained by getting ratio of maximum values of R matrices of the images. Likewise the rotational difference is obtained by having  $\theta$  difference between the matrices. After extracting these parameters derotation and descaling is performed to the sensed image. The translation difference is simultaneously found out by having coordinates of feature points. At last sensed image is shifted accordingly to fit and match with the reference image. Thus, the registered image is obtained and registration task is performed. Experimental results show that the proposed image registration algorithm leads to acceptable registration accuracy and robustness against several image deformations and image processing operations.

A. 2-D Continuous Wavelet Transform

We can represent the 1-D continuous wavelet transform (CWT) in frequency domain given by following equation

$$CWT(s, \omega) = \sqrt{s} F(\omega) \Phi(s\omega) \quad (1)$$

The 2-D CWT is the inner product of a signal or image with a scaled, rotated and translated version of a wavelet function. Now we represent 2-D continuous wavelet transform (CWT) in time domain and frequency domain by following equations

$$CWT(s, b, c) = \frac{1}{\sqrt{s}} \int f(x, y) \psi\left(\frac{x-b}{s}, \frac{y-c}{s}\right) dx dy, \quad (2)$$

(time domain)

$$CWT(s, \omega_1, \omega_2) = \sqrt{s} F(\omega_1, \omega_2) \Phi(s\omega_1, s\omega_2) \quad (3)$$

(frequency domain)

The wavelet  $\Psi$  is highly localized in space; it is either compactly supported or has fast decay. Its integral is zero: for a given scale  $s > 0$  the CWT behaves like a band-pass filter, providing information on where in the image we can find oscillations or details at that scale. At small scales the CWT captures short-lived variations in color such as thin edges; comparing the CWT at different scales reveals what kind of discontinuity is present; at large scales it blurs the image. If the wavelet is stretched in one direction, the CWT gives information on local orientation in the image [16]. For our wavelet, we choose the Mexican hat wavelet, which is stretched in the direction of one of the axes in accordance with parameter [16].

$$MHW(x_1, x_2, s) = \frac{1}{\sigma} \left( 2 - \frac{x_1^2 + x_2^2}{\sigma^2} \right) e^{-\frac{x_1^2 + x_2^2}{2\sigma^2}} \quad (4)$$

Where  $\sigma = 2^{-S}$ ,  $S$  is a scale of the function,  $x_1$  and  $x_2$  are the vertical and horizontal coordinates respectively. When  $\sigma = 1$ ,  $MHW(s)$  is the laplacian of  $g(x_1, x_2) = e^{-0.5(x_1^2 + x_2^2)}$ , a bi dimensional Gaussian; it is isotropic, and in that case the CWT gives no information on object orientation. When scaled, its essential support is a disk with radius proportional to the scale. If  $\sigma \neq 1$ , we have the anisotropic Mexican hat, stretched out or shortened, and its support is an ellipse [17]. The frequency domain response of Mexican hat wavelet is given by

$$\Phi(\omega_1, \omega_2) = -2\pi(\omega_1^2 + \omega_2^2) e^{\frac{1}{2}(\omega_1^2 + \omega_2^2)} \quad (5)$$

We observe the Mexican Hat shaped waveform in the time domain as shown in Fig. 1(a) and frequency domain in Fig. 1(b) respectively.

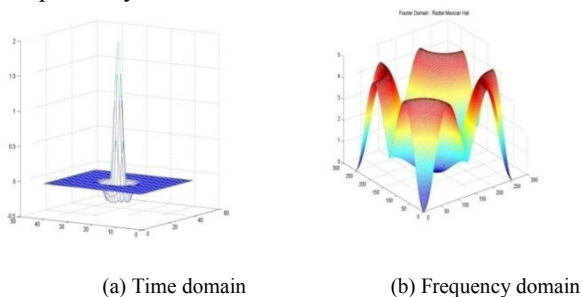


Figure 1 Time and frequency domain representation of the 2D CWT [17].

B. MHW Response at Different Scale

At lower scale, MHW give high frequency detail like corner, thin edges in the image. At higher scale, it gives low frequency smooth or blurs effect in the image. In order to extract feature point from the image, we have to decide appropriate scale. The Mexican Hat Wavelet has perfect circular symmetry in frequency and spatial domains. Fig.2 to Fig.5 shows MHW response in time and frequency domain with coefficient of cameraman image at different scale 2, 3, 4, and 6 respectively.

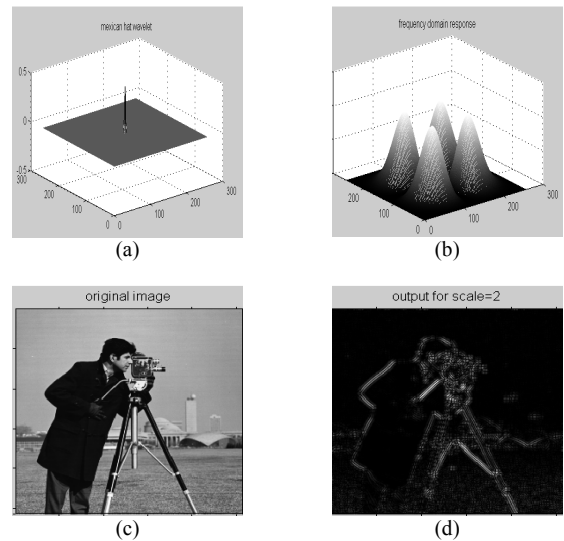


Figure 2 MHW (a) response in time domain (b) response in frequency domain (c) Cameraman image (d) MHW response of image at scale = 2

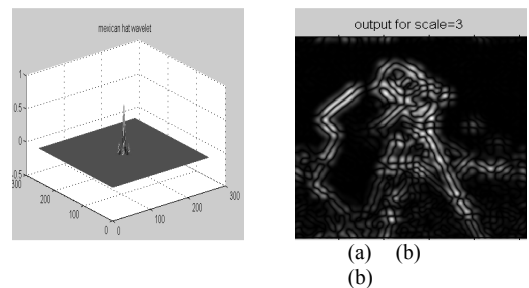


Figure 3 MHW (a) response Time domain (b) MHW response of image at scale=3

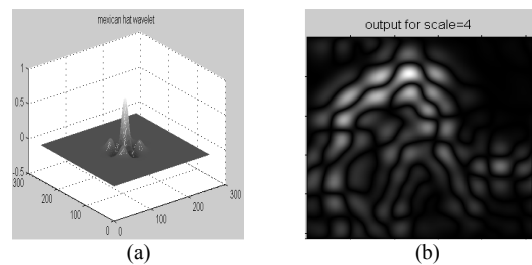


Figure 4 MHW responses (a) Time domain (b) MHW response of image at scale = 4

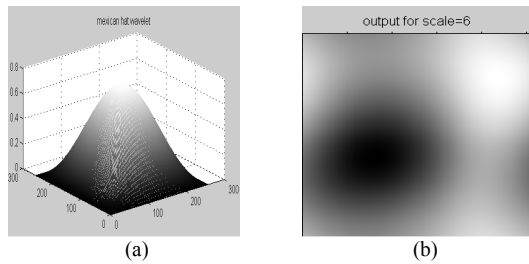


Figure 5 (a) MHW response in time domain  
(b) MHW response of image at scale = 6

From the above results, we can visualize that the Mexican-hat wavelet is equivalent to a band-pass filter. Tuning the wavelet scale controls the spread of the wavelet in the spatial domain and the bandwidth in the frequency domain. At scale 2, 3 and 4 MHW gives high frequency detail whereas at scale 6 and 9 it gives low frequency detail in the cameraman image. In our algorithm, we have decided to extract the feature points at scale 2, 3 and 4.

## II. FEATURE POINT EXTRACTION USING SCALE INTERACTION OF MHW

Our proposed algorithm is based on feature based image registration. So first and important task we have to do is feature extraction using scale interaction based on Mexican hat wavelets. For the image registration point of view, the extracted feature points must have to fulfill some basic objectives like they should be robust enough to some level of variations in scale and the feature extractor should have the ability to be modified to adapt image structures at different scales and the extracted features should have a well-localized support in the image. Here we have presented feature point extraction method that is based on scale interaction of Mexican hat wavelets. This method is based on finding the local maxima of the response of a feature detection operation, which involves convolving the image with the Mexican-hat wavelets. Feature point extraction using scale interaction of Gabor wavelets was proposed in [12]. The Mexican hat is a Laplacian of a Gaussian and its isotropic property makes it insensitive to orientation and a good choice for feature extraction [18]. A Mexican-hat wavelet has the shape of a signal with a positive peak in a negative dish. Convolving an image with Mexican-hat wavelets results in a response, which more likely detects blob-like shapes, bright areas surrounded by dark pixels or vice versa. Varying the width of the central peak of the Mexican-hat wavelet controls the size and the shape of the response [20] [23]. Mexican hat based feature extraction methods such as the ones in [19] and its further development in [21] [24].

An illustration of this feature extraction process is shown in Fig. 6. First of all reference image is taken as input, we call it  $I_R(x, y)$ . After taking its Fourier transform, its convolution is done with the Mexican hat wavelet. Here we use two different scales for the Mexican hat wavelet  $S_1$  and  $S_2$ . This is known as

MHW ( $S_1$ ) and MHW ( $S_2$ ). After convolving the Mexican hat wavelet with the image and taking inverse Fourier transform, we obtain the response  $R(S_1)$  and  $R(S_2)$ . This represented in the equation form as

$$R(S) = I_R(x, y)_f \otimes \otimes \text{MHW}(S) \quad (6)$$

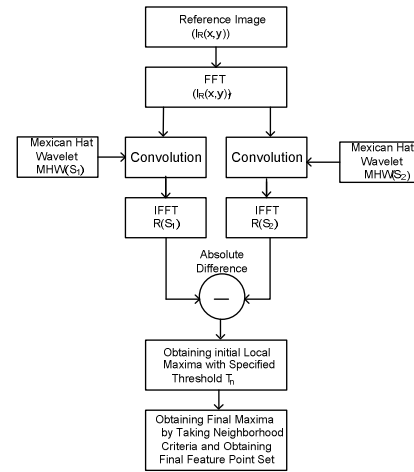


Figure 6 Block diagram of the feature extraction process.

Our next step is to get the scale interaction output. This is nothing but the absolute difference between the obtained responses. It is given as

$$R(s_1, s_2) = |R(s_1) - R(s_2)| \quad (7)$$

$$= |\text{IFFT} \{I_R(x, y)_f \otimes \otimes \text{MHW}(S_1)\} - \{I_R(x, y)_f \otimes \otimes \text{MHW}(S_2)\}|$$

Where  $I_R(x, y)_f$  denotes the Fourier transform of  $I_R(x, y)$ . The second stage of the feature extraction process localizes the feature points of the image by finding the local maxima of the response  $R(S_1, S_2)$ . This local maxima obtaining stage will be applied to the response  $R(S_1, S_2)$ , in Equation (8) using the following algorithm:

- (i) Find the maximal values that are greater than the specified threshold  $T_n$  in equally non-overlapped blocks of size  $N \times N$ ; such initial maximal values may include points on the boundaries of the blocks, which do not represent a local maximum of  $R(S_1, S_2)$ .
- (ii) Take each maximal point as the centre of a disk-shaped neighborhood of radius  $r_n$  and find one local maximum in each neighborhood; this will eliminate maximal points that are not local maxima of  $R(S_1, S_2)$  or local maxima that are too close to each other.
- (iii) Repeat step (ii) until the obtained local maxima do not change locations.
- (iv) In order to avoid the effects of the image borders on the feature extraction process, only the maxima found in the image area that is away from the image border by distance  $r_c$  are kept.

### A. Analysis of Feature point Extraction Process.

Here we have taken an 8-bit gray level ‘cameraman’ image of size  $256 \times 256$  as a reference image as shown in Fig. 7(a). First of all reference image and its convolution with the Mexican hat wavelet at scale  $S_1=2$  and  $S_2=4$  is taken shown in Fig. 7(b) and Fig. 7(c) respectively. Here inverted image of the result is shown only for the clear visualization purpose. The absolute difference between obtained responses is carried out as mentioned in Equation (7). The result of scale interaction process is shown in Fig. 7(d).

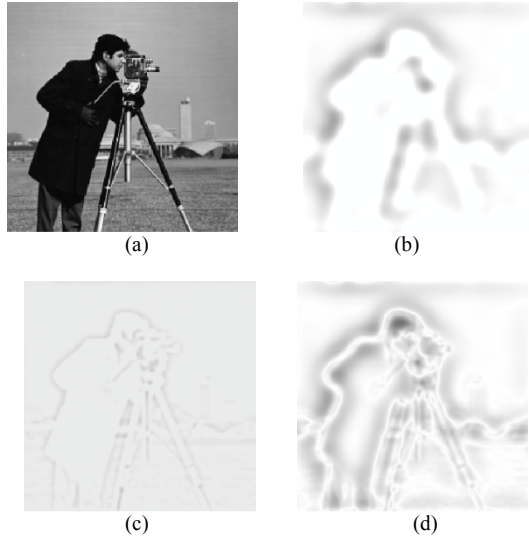


Figure 7 Feature point extraction stages (a) Reference image (b) Response of applying Mexican hat Wavelet with scale 2, (c) Response of applying Mexican-hat wavelet with scale 4, (d) Absolute difference of the two responses.

Next step is find the initial local maxima that are greater than a specified threshold  $T_n = 58$  in equally non overlapped blocks of size  $32 \times 32$ . Fig.8 (a) shows the image with superimposed local maxima. After getting local maxima we can visualize that the obtained initial local maxima might not be actual peaks and some of them might be very close to each other and looks like bunch of feature points. Take each local maximum point as the center of a disk-shaped neighborhood of radius  $r_n$  is 16 and find one local maximum in each neighborhood, this will eliminate local maxima that are very close to each other. Repeat above step until the obtained local maxima do not change locations. The final extracted feature points shown in Fig. 8 (b).

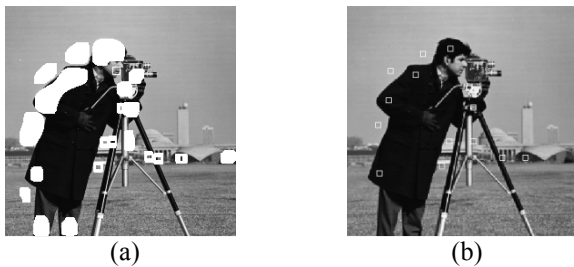


Figure 8 Reference image with the extracted points superimposed on the image (a) after Thresholding (b) after applying disk shaped neighborhood criteria.

We will get the similar results for the sensed image, represented in Fig. 9 and Fig. 10.

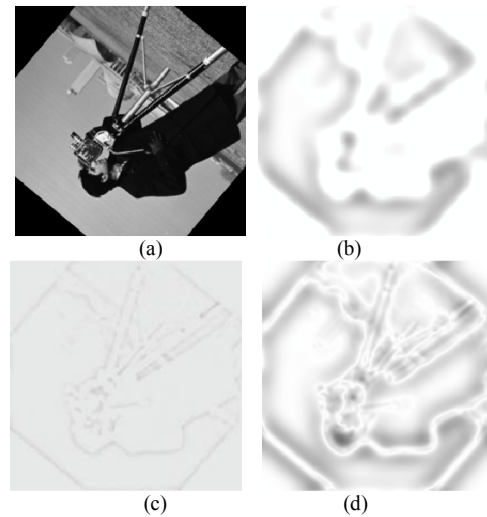


Figure 9 Feature point extraction stages (a) Sensed image (b) Response of MHW with  $S = 2$  (c) Response of applying MHW with scale = 4 (d) Absolute difference of the two responses.

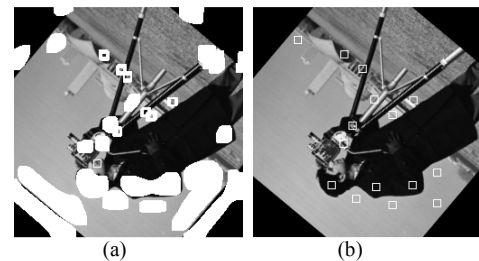
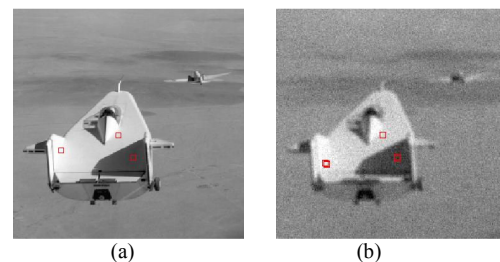


Figure 10 Sensed image with the extracted points superimposed on the image (a) after Thresholding (b) after applying disk shaped neighborhood criteria.

### B. Performance Analysis

The performance of this Mexican hat wavelet based feature point extractor is evaluated by taking one image as reference and applying different types of degradations to it and using it as sensed image. After visualization of the extracted feature points from both of these images as shown in Fig.11, we can say that it is a robust feature point extractor which extracts those feature points which are as many as possible in the same, or within the neighborhood of locations, with respect to the contents of the image, regardless of any degradations or distortions applied on the image. The presented scale-interaction Mexican hat wavelets feature point extraction method will be used in image registration as discussed in the subsequent chapters.



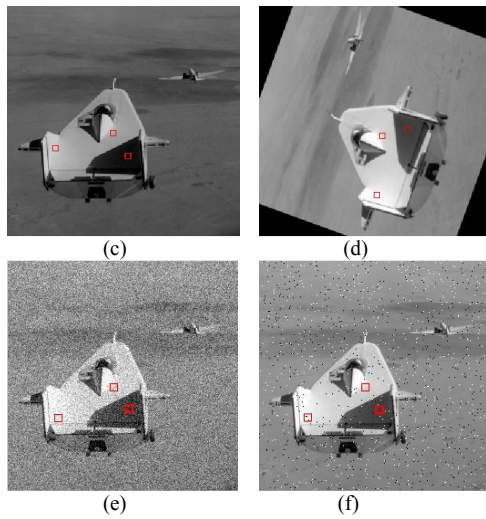


Figure 11 Feature point extraction of distorted/degraded images: (a) No distortion, (b) Blurring, (c) Brightness change, (d) Rotation, (e) Gaussian noise contamination (f) 'Salt and Pepper' noise

### III. FEATURE POINT MATCHING USING INVARIANT MOMENTS

Feature Points from both images are extracted using Mexican-Hat Wavelet and control-point correspondence is achieved with invariant moments. After detecting corresponding control points from both images radon transform approach is applied to register images. One problem in image registration is automatic registration of an image regardless of its size, position and orientation. To achieve this goal, the extracted image features should have the invariance properties against image transformations including scale change, translation and rotation [25]. These image transformations can be approximated by affine transformation which is an important class of linear 2-D geometric transformation. The affine transformation preserves collinearity (i.e., all points lying on a line initially still lie on a line after transformation) and ratios of distances (e.g., the midpoint of a line segment remains the midpoint after transformation). The image features with the invariance properties are called image invariants.

Fourier descriptors and Hu's seven moment invariants are the most popular shape based image invariants and have been used in image recognition, indexing, and retrieval systems [13],[15][26]. Fourier descriptors are boundary-based image features which only compute the pixels along the image contours. On the contrary, Hu's seven moment invariants are region-based image features which take all of the pixels of the image into account [15]. Hu's seven moment invariants have the invariance property against affine transformations including scale change, translation and rotation.

#### A. Moment Representation

Moment invariants (MIs) are a class of image descriptors first derived by Hu [25], [26] who employed the results of the theory of algebraic invariants and derived his seven famous invariants to rotation of 2-D objects. Moment-based invariants are the most common region-based image invariants which have been used as pattern features in many applications [27].

Hu first introduced a set of invariants using nonlinear combinations based on regular moments in 1961 [25]. For a 2-D continuous function  $f(x,y)$ , the moment of order  $(p+q)$  is defined as

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x,y) dx dy \text{ for } p,q = 0,1,2,\dots \quad (8)$$

A uniqueness theorem states that if  $f(x,y)$  is piecewise continuous and has nonzero values only in a finite part of the  $xy$ -plane, moment of all orders exist, and the moment sequence  $(m_{pq})$  is uniquely determined by  $f(x,y)$ . Conversely,  $(m_{pq})$  is uniquely determined by  $f(x,y)$ . The central moments are defined as

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y) dx dy \quad (9)$$

where  $\bar{x} = \frac{m_{10}}{m_{00}}$  and  $\bar{y} = \frac{m_{01}}{m_{00}}$

If  $f(x,y)$  is a digital image, then equation (5.2) becomes

$$\mu_{pq} = \sum (x-\bar{x})^p (y-\bar{y})^q f(x,y) \quad (10)$$

and the normalized central moments, denoted  $\eta_{pq}$ , are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, \text{ where } \gamma = \frac{p+q}{2} + 1 \text{ for } p+q=2,3,\dots \quad (11)$$

A set of seven invariant moments can be derived from the second and the third moments by Hu [20]. As shown below, Hu derived the expressions from algebraic invariants applied to the moment generating function under a rotation transformation. They consist of groups of nonlinear centralized moment expressions. The result is a set of absolute orthogonal moment invariants, which can be used for scale, position, and rotation invariant pattern identification.

$$\phi_1 = \eta_{20} + \eta_{02} \quad (12)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (13)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - 3\eta_{03})^2 \quad (14)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (15)$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \quad (16)$$

$$\phi_6 = (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (17)$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (18)$$

### IV. PROPOSED AUTOMATIC IMAGE REGISTRATION ALGORITHM

This Section introduces an algorithm for automatic feature based image registration. The main objective of the proposed algorithm is accurately registering images which are geometrically distorted. The proposed algorithm is shown in Fig. 12.

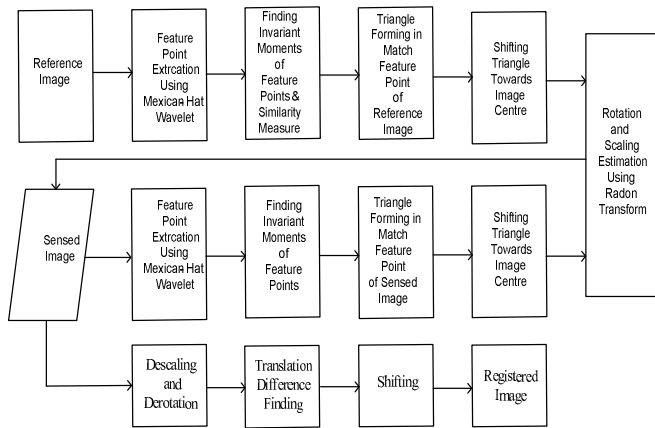


Figure 12 Proposed Automatic Image Registration Algorithm

As shown in Fig. 12, automatic feature points are extracted using Mexican-Hat Wavelet as discussed in Section III. After the strong feature points extracted from reference and sensed images, a correspondence mechanism is required between these two feature point sets. This correspondence mechanism fulfils the requirement of pairing the feature point of reference image with its correspondent one in the sensed image. In this proposed algorithm, Invariant moment based similarity measure approach is used to establish the correspondence between the two images.

This correspondence is evaluated using a circular neighbourhood centered on each and every feature point. For every extracted feature point, select a circular neighbourhood of radius  $R$  centered at this point and construct a set of absolute orthogonal moment invariants vectors  $(\phi_1, \phi_2, \phi_3, \dots, \phi_7)$  as described in Equations (12) to (18). The similarity between the regions is judged by computing the Euclidean Distance between invariable regions of reference image with sensed image.

In our proposed method, we had calculated distance between feature point moment invariant vectors of reference image and moment invariant vectors of all feature points of sensed image. For example, suppose there are five features point in reference image and seven feature points in sensed image. We had considered any one feature point invariant moment vector and find the distance between all extracted feature points invariant moment vectors of sensed image. Then we had considered second feature point moment invariant vector and find the distance with all extracted feature points vectors of sensed image. After detecting distance, corresponding feature points are detected by minimum distance rule with the threshold in the Euclidean space of the invariants.

After detecting, similar feature points from both images, we had considered any three similar feature points as control points from both images and form a triangle by considering any one feature point and find the maximum distance across all feature points in the image. Then, draw a line between this maximum distance feature points and second line with maximum slope and connect third line to form a triangle. This

way, we had drawn triangle in both images and shift it towards image centre.

After this, we had applied radon transform algorithm for extracting scaling. The radon function computes projections of an image matrix along specified directions. A projection of a two-dimensional function  $f(x,y)$  is a set of line integrals. The radon function computes the line integrals from multiple sources along parallel paths, or beams, in a certain direction [10] [13]. The beams are spaced 1 pixel unit apart. Radon transform of a two dimensional function  $f(x,y)$  is defined as

$$R(r, \theta)[f(x, y)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (f(x, y)\delta(r - x\cos\theta - y\sin\theta))dx dy \quad (19)$$

Where,  $r$  is the perpendicular distance of a line from the origin and  $\theta$  is the angle formed by the distance vector.

#### A. Analysis of Proposed Image Registration Algorithm

In this section, the performance evaluation is done for the proposed automatic image registration algorithm, for different types of distortions. An original image or reference image is geometrically distorted and different types of geometrical deformations added in it. A set of experiments has been performed to evaluate the performance of the proposed algorithm with respect to registration accuracy and robustness.

- (i) Feature Point Extraction Using Mexican-Hat Wavelet: Here we select an 8-bit gray level ‘living room’ image as a reference image. The size of the image is  $256 \times 256$ . Fig. 13 (a) and (b) shows the reference and sensed image respectively. Fig. 13 (a) and (b) shows the final extracted feature points  $(R_1, R_2, \dots, R_5)$  in reference image and  $(S_1, S_2, \dots, S_5)$  in sensed image.
- (ii) After extracting feature points from both images, we determined invariant moments by considering circular template across all feature points. We considered all the points of circular neighborhood of radius 20 around feature points. Table-I and Table-II shows invariant moments of extracted features of reference image and sensed image. Fig. 14 & Fig. 15 shows circular neighborhood around all extracted feature points of images.

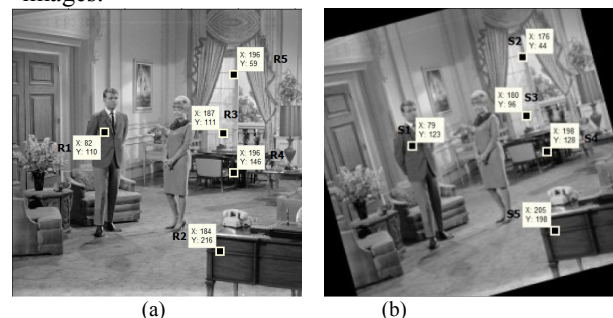


Figure 13 (a) Reference image (b) Sensed Image with feature points  
(Rotation= $15^\circ$ , Scaling=1)

TABLE I. INVARIANT MOMENT OF REFERENCE IMAGE FEATURE POINTS

Moment	$\phi_1$	$\phi_2$	$\phi_3$	$\phi_4$	$\phi_5$	$\phi_6$	$\phi_7$
MR1	2.7121	8.4255	11.0956	10.7246	21.6372	14.9400	22.6079
MR2	2.6609	7.6623	9.7978	9.9937	19.9234	13.8302	20.3094
MR3	3.0514	9.3793	11.442	11.879	23.8940	16.5696	23.5881
MR4	2.5252	7.6820	9.0593	9.4312	18.6784	13.3216	19.7068
MR5	3.0817	9.2234	12.446	12.419	25.0066	17.0800	24.9986

TABLE II. INVARIANT MOMENT OF SENSED IMAGE FEATURE POINTS

Moment	$\phi_1$	$\phi_2$	$\phi_3$	$\phi_4$	$\phi_5$	$\phi_6$	$\phi_7$
MS1	2.7095	8.3341	11.2460	10.7846	21.9022	14.9727	22.0125
MS2	3.08188	9.3097	12.4795	12.4058	24.9254	17.1700	25.1113
MS3	3.05326	9.1451	11.5194	11.9435	23.8650	16.5162	23.7922
MS4	2.52788	7.6959	9.08295	9.39771	18.6405	13.2671	19.6062
MS5	2.66365	7.6283	9.79171	9.97759	19.8722	13.8206	20.5355

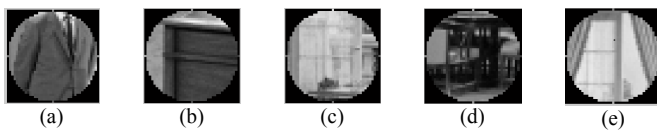


Figure 14 circular neighborhood around feature points  
(a) R1 (b) R2 (c) R3 (d) R4 (e) R5

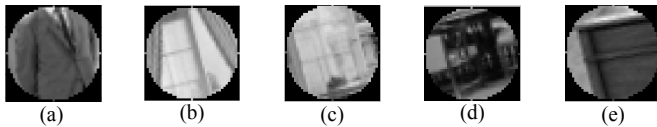


Figure 15 circular neighborhood around feature points  
(a) S1 (b) S2 (c) S3 (d) S4 (e) S5

(iii) After this, we find distance between invariant moments of reference and sense feature points. First, we consider moment invariants vector MR1 and determined the distance with all moment invariant vectors (MS1, MS2,...MS5) of sensed image. The minimum distance between this is mark as a corresponding feature point. Table-III shows the distance calculation of reference feature points and sensed feature points. As shown in above Table-III, the minimum distance indicate the similar feature points in reference and sensed image. The minimum distance  $D_{MR1MS1}$  indicate feature point R1 in reference image is match with feature point S1 of sensed image.

TABLE III. MINIMUM DISTANCE CALCULATION

Distance	$D_{MR1MS1}$	$D_{MR1MS2}$	$D_{MR1MS3}$	$D_{MR1MS4}$	$D_{MR1MS5}$
	<b>1.1975</b>	12.3404	7.6915	11.9250	7.8536
Distance	$D_{MR2MS1}$	$D_{MR2MS2}$	$D_{MR2MS3}$	$D_{MR2MS4}$	$D_{MR2MS5}$
	5.1995	20.3057	15.6568	4.0268	<b>0.3459</b>
Distance	$D_{MR3MS1}$	$D_{MR3MS2}$	$D_{MR3MS3}$	$D_{MR3MS4}$	$D_{MR3MS5}$
	10.4268	4.8187	<b>0.6637</b>	19.5860	15.5146
Distance	$D_{MR4MS1}$	$D_{MR4MS2}$	$D_{MR4MS3}$	$D_{MR4MS4}$	$D_{MR4MS5}$
	8.9728	24.0790	19.4301	<b>0.2666</b>	3.9924
Distance	$D_{MR5MS1}$	$D_{MR5MS2}$	$D_{MR5MS3}$	$D_{MR5MS4}$	$D_{MR5MS5}$
	14.8786	<b>0.4166</b>	4.4214	24.0379	19.9665

(iv) Then, we form a triangle between any three similar feature points in reference image (R1,R2,R5) and sense image (S1,S2,S5) and apply radon transform for calculating scaling between the images registering the images.

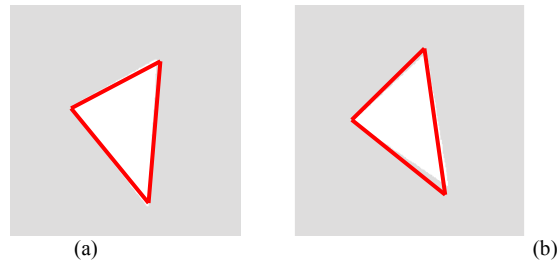


Figure 16 Triangle after region filling for (a) reference image  
(b) sensed image.

Radon transform is now applied on Fig. 16. Two matrices are obtained after applying radon transform on Fig. 16(a) and Fig. 16(b). Practical value of  $R_{1max}$  is 155.29 for reference image and  $R_{2max}$  is 154.49 for sensed image. The ratio  $R_{2max}/R_{1max}$  gives 0.995 which is nothing but the desired and accurate scaling between sensed and reference image. In order to recover rotation between images, we have formed line between two similar feature points. To make a line we need to get all the points which satisfy the line equation. We get floating point numbers so approximation is done and nearest integer is chosen to convert them to integers. After making line the images look like Fig. 17.

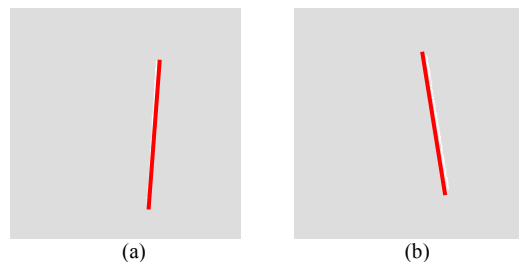


Figure 17 Line formed after joining two points are for  
(a) reference image and (b) sensed image.

Radon transform is now applied on Fig. 17. Here, column size is  $360^0$  because the radon transform is applied for  $1:360^0$ . Now cross-correlation is obtained between this and rotation is extracted. The maximum value of correlation occurs at the coordinates  $[r,c] = [405,345]$ . Here we want correlation of the column as shifting operation is related with  $\theta$ . Maximum column size of template i.e.  $360^0$  is subtracted from c, therefore  $360^0 - 345^0 = 15^0$ . This is nothing but extracted rotation of sensed image with respect to reference image.

(v) As we know the four steps of image registration, three steps are performed in previous chapters. The last step is image resampling and transformation. Here descaling, derotation and translation is applied to sensed image in order to obtain registered image. The final registered image is shown in Fig. 18.

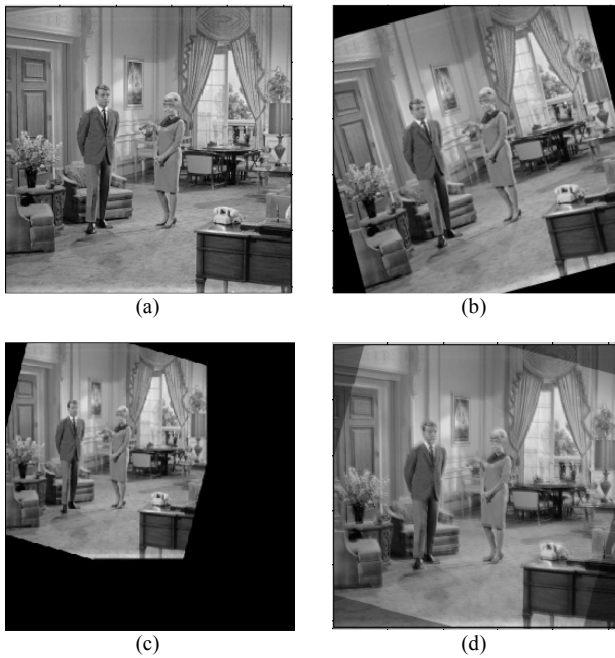


Figure 18 (a) Reference Image (b) Sensed image with rotation 15°  
(c) Registered image (d) Sensed image overlaid on reference image

**B. Simulation Results**

Here we have considered different 8-bit gray level images as a reference image and sensed image with geometrical deformation. In Fig. 19(b), the sensed image having 30° rotation, scale =1.3. After applying proposed algorithm, the registered image is shown in Fig. 19(d). It shows the accuracy of image registration. After successful registration of the “living room” image, we have applied different geometrical deformations. Fig. 20 to Fig. 22 shows the registered images with different geometrical deformations.

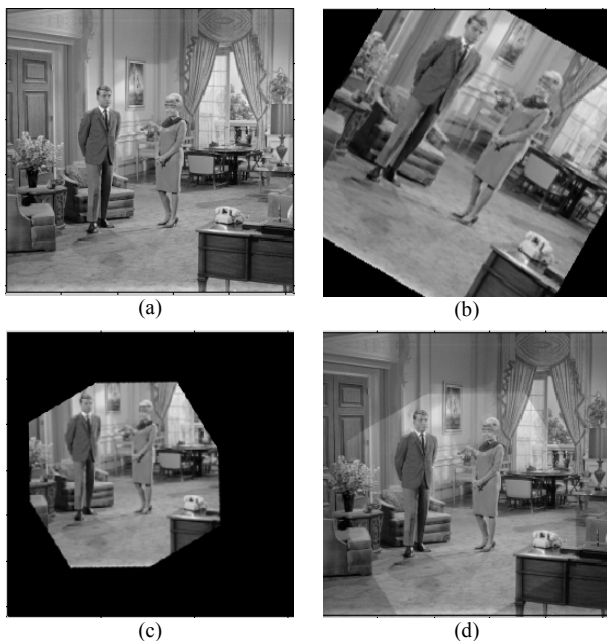


Figure 19 (a) Reference Image (b) Sensed image with rotation 30°, scale=1.3  
(c)Registered image (d) Sensed image overlaid on reference image

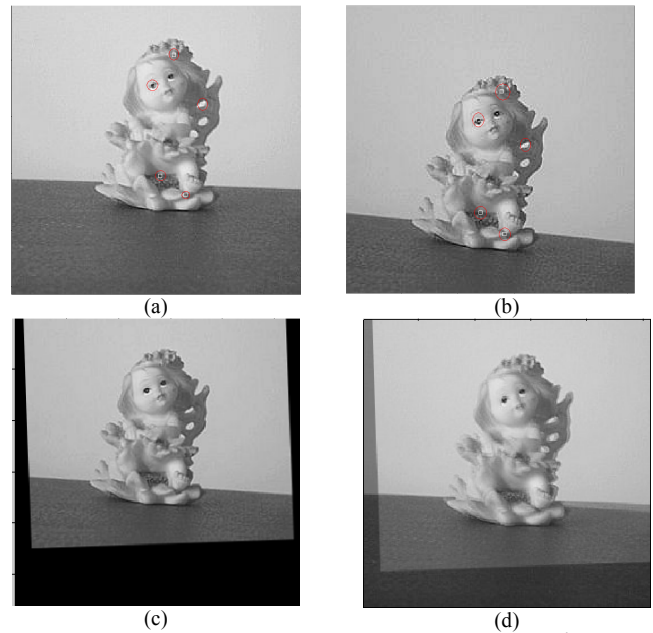


Figure 20 (a) Reference Image (b) Sensed image with rotation 2°, scale=1  
(c) Registered image (d) Sensed image overlaid on reference image

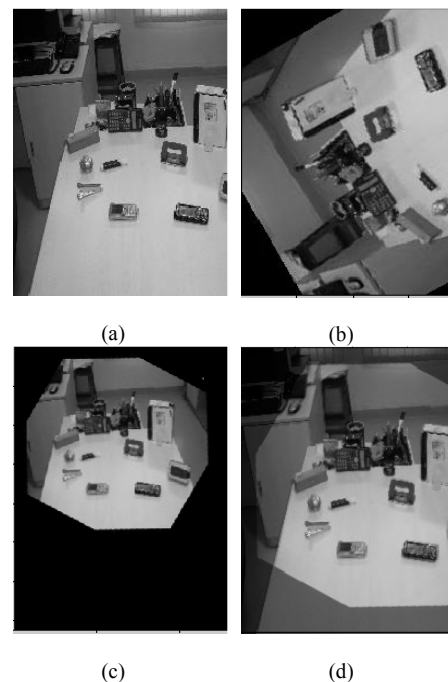
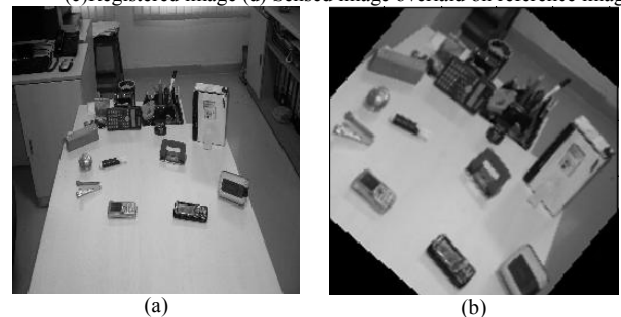


Figure 21(a) Reference Image (b) Sensed image with rotation 240°, scale=1.2  
(c)Registered image (d) Sensed image overlaid on reference image



(a) (b)



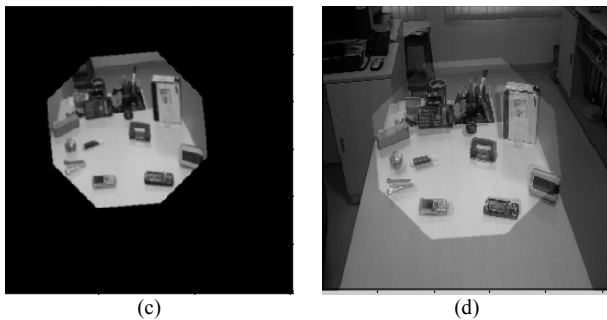


Figure 22 (a) Reference Image (b) Sensed image with rotation  $45^\circ$ , scale=1.8  
(c) Registered image (d) Sensed image overlaid on reference image

## V. CONCLUSIONS

In this paper, a new feature based image registration approach is proposed. This type of approach gives a new dimension to the existing feature based image registration methods. After successful feature extraction using Mexican Hat Wavelet, correspondence between extracted feature points is established using Hu's invariant moments. Once the set of correctly matched feature point pairs between two images are found, the scaling, and rotations are extracted using radon transform. With this proposed algorithm, we registered images with any degree of rotation and scale up to 1.8.

## REFERENCES

- [1]. L.G. Brown, "A survey of Image Registration Techniques," *ACM Computing Surveys*, vol. 24, no. 4, pp. 325-3760, 1992.
- [2]. Gang Hong and Yun Zhang, "Combination of feature-based and area-based image registration technique for high resolution remote sensing image," *Geoscience and Remote Sensing Symposium, 2007, IGARSS 2007*, pp.377-380, 2007.
- [3]. B. Zitova and J. Flusser, "Image Registration methods: A Survey," *Image Vision Computing*, vol. 21, no. 11, pp. 977-1000, 2003.
- [4]. P.Bas, J-M.Chassery, B.Macq, "Geometrically invariant watermarking using feature points, IEEE Trans. On Image Processing," Vol.11, pp. 1014-1028,2002
- [5]. K.vinay and B.Shreyas, "Face recognition using gabor wavelets," in Fortieth Asilomar Conference on Signal, Systems and Computers,2006, ACSSC'06, pp.593-597,2006
- [6]. G.Wang and Z.Ou, "Face recognition based on image enhancement and gabor feature," in Proc. The Sixth World Congress on Intelligent Control and Automation,2006, WCICA 2006, Vol. 2,pp.9761-9764,2006
- [7]. D.G.Lowe, "Distinctive image features from scale invariant key points," *International Journal of Computer Vision*, Vol. 60,pp.91-110,2004
- [8]. Steven Gillan and Pan Agathoklis, "Image registration using feature points, Zernike moments and an M-estimator," *The International Midwest Symposium on Circuits and Systems (MWSCAS) is the oldest Circuits and Systems symposia*, Washington, pp.434-437 August 2010.
- [9]. Collins, D.L. Holmes, C.J. Peters, T.M. and Evans, A.C., "Automatic 3D model-based neuro anatomical segmentation". *Human Brain Mapping*, pp.190-208, 1995.
- [10] M.Van Ginkel, C.L.Luengo Hendriks and L.J. van Vliet, "A short introduction to the radon and hough transforms and how they relate to each other," in the Quantitative Image Group Technical Report Series, N. QI-2004-01, pp.1-9,2004

- [11] K. P. Soman, K. I. Ramachandran, "Insight into Wavelets, from theory to practice, Prentice-Hall of India Private Limited, 2006.
- [12] B. Manjunath, C. Shekhar, and R. Chellappa, "A new approach to image feature detection with applications," *Pattern Recognition*, vol. 29, No. 4, pp. 627-640, 1996.
- [13] A K Jain, "Digital Image Processing," 2<sup>nd</sup> Edition, Prentice Hall Publication, 2006.
- [14] M. S. Yasein and P. Agathoklis, A robust, feature-based algorithm for aerial image registration," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE 2007)*, Vigo, Spain, pp. 1731-1736, 2007.
- [15] R C Gonzalez, "Digital Image Processing Using MATLAB," 2<sup>nd</sup> Edition, Mc-Graw Hill Publication, 2010.
- [16] Heejong Yoo, "2D Continuous Wavelet Transform", Available on line on <http://www.isye.gatech.edu>, Downloaded in September 2010.
- [17] Diego Romero, Ana Ruedin, and Leticia Seijas, *Wavelet-Based Feature Extraction for Handwritten Numerals*, in Springer Berlin / Heidelberg, *Image Analysis and Processing (ICIAAP 2009)*, vol. 5716, pp. 374-383, 2009.
- [18] S. Bhattacharjee and M. Kutter, "Compression tolerant image authentication," in *Proceedings of the IEEE International Conference on Image Processing (ICIP-1998)*, Chicago, USA, pp. 435-439, 1998.
- [19] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, Toward second generation watermarking schemes," in *Proceedings of the IEEE International Conference on Image Processing (ICIP-1999)*, Kobe, Japan, pp. 320-323, 1999.
- [20] M. S. Yasein and P. Agathoklis, Automatic and robust image registration using feature points extraction and zernike moments invariants," in *Proceedings of the Fifth IEEE International Symposium on Signal Processing and Information Technology*, Athens, Greece, pp. 566-571,2005.
- [21] M. S. Yasein and P. Agathoklis, A robust, feature-based algorithm for aerial image registration," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE 2007)*, Vigo, Spain, pp. 1731-1736, 2007.
- [22] Chih-Wei Tang and Hsueh-Ming Hang, A Feature-Based Robust Digital Image Watermarking Scheme, in *IEEE Trans. on Signal Processing*, vol. 51, No.4, pp.950-959, 2003.
- [23] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings Royal Society of London*, vol. 207, No. 1167, pp. 187-217, 1980.
- [24] A.L. Cunha, J. Zhou, and M.N. Do, "The Nonsubsampled Contourlet Transform: Theory, Design, and Applications," *IEEE Trans. Image Processing*, vol. 15, No. 10, pp. 3089-3101, 2006.
- [25] Zhihu Huang and Jinsong Leng, "Analysis of Hu's Moment Invariants on Image Scaling and Rotation," 2<sup>nd</sup> International Conference on Computer Engineering and Technology, Vol.7, pp.476-480, 2010.
- [26] Qing Chen, Petriul, and Xiaoli Yang, "A Comparative Study of Fourier Descriptors and Hu's Seven Moment Invariants for Image Recognition," *Canadian Conference on Electrical and Computer Engineering*, pp.103-106, 2004.
- [27] Youcef Bentoutou and N. Taleb, "Automatic Extraction of Control Points for Digital Subtraction Angiography Image Enhancement," *IEEE Trans. on nuclear science*, Vol. 52, No. 1, pp.238-246, 2005.

AUTHORS PROFILE



Prof. Jignesh N. Sarvaiya received his M.Tech. Degree from IIT Bombay, India. Presently, He is Assistant Professor at Electronics Engineering Department, SVNIT Surat, (Gujarat) India. His Research Interest includes Image Processing, Image Registration, Bio-Medical Instrumentation and Electronics System Design.



Dr. Suprava Patnaik received M.Tech. Degree in Electronic & Communication Engineering from NIT, Rourkela, in 1992, and Ph.D. from Electrical, Electronics & Communication Engineering Depttment, IIT, Kharagpur in 2004. Presently, She is Associate Professor in the Electronics Engineering Department, SVNIT, Surat (Gujarat) India. Her research interests include Signal and Image processing, Image Segmentation, Pattern Recognition, Digital Watermarking, Image Registration and Video Compression .

# Human Face Detection under Complex Lighting Conditions

Md. Golam Moazzam

Department of Computer Science  
and Engineering,  
Jahangirnagar University, Savar,  
Dhaka-1342.

Ms. Rubayat Parveen

Department of Computer Science  
and Engineering,  
Jahangirnagar University, Savar,  
Dhaka-1342.

Md. Al-Amin Bhuiyan

Department of Computer Science  
and Engineering,  
Jahangirnagar University, Savar,  
Dhaka-1342.

**Abstract**— This paper presents a novel method for detecting human faces in an image with complex backgrounds. The approach is based on visual information of the face from the template image and is commenced with the estimation of the face area in the given image. As the genetic algorithm is a computationally expensive process, the searching space for possible face regions is limited to possible facial features such as eyes, nose, mouth, and eyebrows so that the required timing is greatly reduced. In addition, the lighting effects and orientation of the faces are considered and solved in this method. Experimental results demonstrate that this face detector provides promising results for the images of individuals which contain quite a high degree of variability in expression, pose, and facial details.

**Keywords**- Face Detection; Genetic Searching; Fitness Function; Cross-over; Mutation; Roulette Wheel Selection.

## I. INTRODUCTION

Face detection has many real world applications, like human-computer interface, surveillance systems, video-conferencing, forensic applications, pedestrian detection, image databases, etc. However, the development of a reliable system for human face detection in a complex scene is very difficult due to variation in illumination, variability in scale, location, orientation (up-right, rotated) and pose (frontal, profile). Furthermore, facial expression, occlusion and lighting conditions change the overall appearance of face.

The face detection problem has been faced up with various approaches over the last few decades: neural network, principal components, independent components, skin color based methods [1], [2]. Each of them imposes some **constraints**: frontal view, expressionless images, limited variations of lighting conditions, hairstyle dependence, uniform background, and so on. Yokoo *et al.* [3] proposed a face detection method by searching an ellipse using the genetic algorithm.

As the performance of edge detection is quite dependent on lighting conditions, so an ellipse may not occur in a true face. Yang *et al.* [4] utilized a hierarchical knowledge-based system which consists of three levels of detection. Level 1 and level 2 are based on mosaic images, so the method is difficult to locate face regions accurately. Furthermore, these methods are unable to detect a rotated human face.

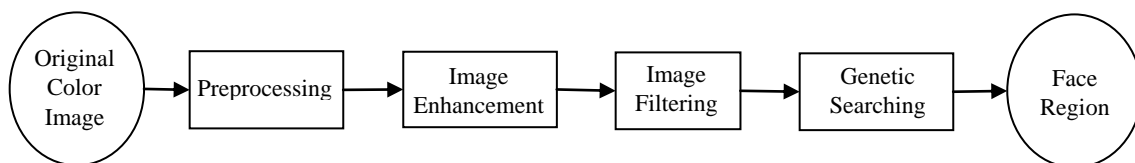
The main objective of this work is to propose a reliable method to detect human faces at different orientations. Possible face regions are selected by means of the Genetic Algorithm (GA) [5] and the fitness function is based on their projections on the template image. Experimental results indicate that the system is capable of detecting human faces in a complex scene with a high degree of variability in expression, pose, and facial details.

## II. FACE DETECTION METHODOLOGY

Face detection is concerned with determining which part of an image contains face. This is the first step of face recognition which requires both high- and low-level visual and geometric information processing. This work presents genetic searching for detecting human faces in a complex background. Face detection is achieved by employing template matching between a known face image and the input image. The main steps employed for the face detection process are shown in **Fig. 1**.

### A. Preprocessing

The original image is obviously a color image. For detecting the face area, the image content is first converted from RGB to HSV color space. In the HSV color model, a color is described by three attributes: hue, saturation, and value. The conversion from RGB to HSV has been accomplished using the following equations [6], [7]:



**Fig. 1** Fundamental steps employed for face detection

$$H_1 = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{\sqrt{(R-G)^2 + (R-B)(G-B)}} \right\} \quad (1)$$

Ranging  $[0, 2\pi]$ , where  $H = H_1$  if  $B \leq G$ , otherwise

$$H = 360^\circ - H_1;$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \quad (2)$$

$$V = \frac{\max(R, G, B)}{255} \quad (3)$$

where R, G, B are the red, green, and blue component values which exist in the range  $[0, 255]$ .

While registering images, the eyes, tip of the nose, and the corners of the mouth of each face is labeled. These points are then used to normalize each face to same scale, orientation and position. The normalization is performed by mapping the facial features to some fixed locations in an  $M \times N$  image. Each normalized image is then subjected to some image processing operations like image enhancement and filtering to account for different lighting conditions and contrast.

### B. B. Image Enhancement

The face images may be of poor contrast because of the limitations of the lighting conditions. To make an efficient

$$C_{g,rms} = \left[ \frac{1}{n-1} \sum_{i=1}^n (x_{g,i} - \bar{x}_g)^2 \right]^{1/2} \quad (5)$$

$$C_{b,rms} = \left[ \frac{1}{n-1} \sum_{i=1}^n (x_{b,i} - \bar{x}_b)^2 \right]^{1/2} \quad (6)$$

where  $x_{r,i}$ ,  $x_{g,i}$ ,  $x_{b,i}$  are the normalized illumination due to red, green, and blue color components, respectively such that  $0 < x_{r,i} < 1$ ,  $0 < x_{g,i} < 1$ ,  $0 < x_{b,i} < 1$ , and  $\bar{x}_r$ ,  $\bar{x}_g$ ,  $\bar{x}_b$  are the mean normalized illuminations due to red, green, and blue color components. With this definition, images captured at different lighting conditions will have the same contrast if their rms contrasts are equal. The rms contrast does not depend on spatial frequency contrast of the image or the spatial distribution of contrast in the image. All images are maintained with the same luminance and same rms contrast using the following equation:

$$\mathbf{g}_r = \alpha_r \mathbf{f}_r + \beta_r, \quad \mathbf{g}_g = \alpha_g \mathbf{f}_g + \beta_g, \quad \mathbf{g}_b = \alpha_b \mathbf{f}_b + \beta_b \quad (7)$$

where  $\alpha_r$ ,  $\alpha_g$ ,  $\alpha_b$ , are the contrast due to red, green, and blue color components, respectively, and  $\beta_r$ ,  $\beta_g$ ,  $\beta_b$  are the brightness to be increased or decreased from the respective red, green, and blue components  $\mathbf{f}_r$ ,  $\mathbf{f}_g$ ,  $\mathbf{f}_b$  of the original image to the new image  $\mathbf{g}$ . Fig. 2 (Courtesy: University of Hall, UK) shows some color face images before and after the illumination and rms contrast equalization process.



(a) Images with different illumination and contrast



(b) Images with same illumination and same rms contrast

Fig.2 Illumination and rms contrast equalization

and psychologically meaningful representation, and make the image illumination invariant in terms of sunny or cloudy environments, it is processed with fixed rms contrast and illumination equalization. For this purpose the following equations [7], [8] are employed in this method:

$$C_{r,rms} = \left[ \frac{1}{n-1} \sum_{i=1}^n (x_{r,i} - \bar{x}_r)^2 \right]^{1/2} \quad (4)$$

### C. Image Filtering

Various sources of noise may exist in the input image. The fine details of the image represent high frequencies which mix up with those of noise. So low-pass filters are used to obliterate some details in the image. In this experiment, Prewitt filter is used to suppress the noise.

### D. Genetic Searching

Genetic algorithm is a blind search technique which is used for searching possible facial regions in an image. Each generated solution for a problem is called a chromosome which

is defined by four parameters to specify a face region. The parameters are the location  $(x, y)$ , the face size, and the angle of rotation,  $\theta$  as illustrated in Fig.3. In this method, a population of possible face regions of different locations, sizes, and slopes is generated randomly.

the next generation. A new population is then generated by means of genetic operations: cross-over and mutation. This evolution process is iterated until a near-optimal solution is obtained or a given number of generations are reached. However, different steps employed in the genetic algorithm are shown in Fig. 4.

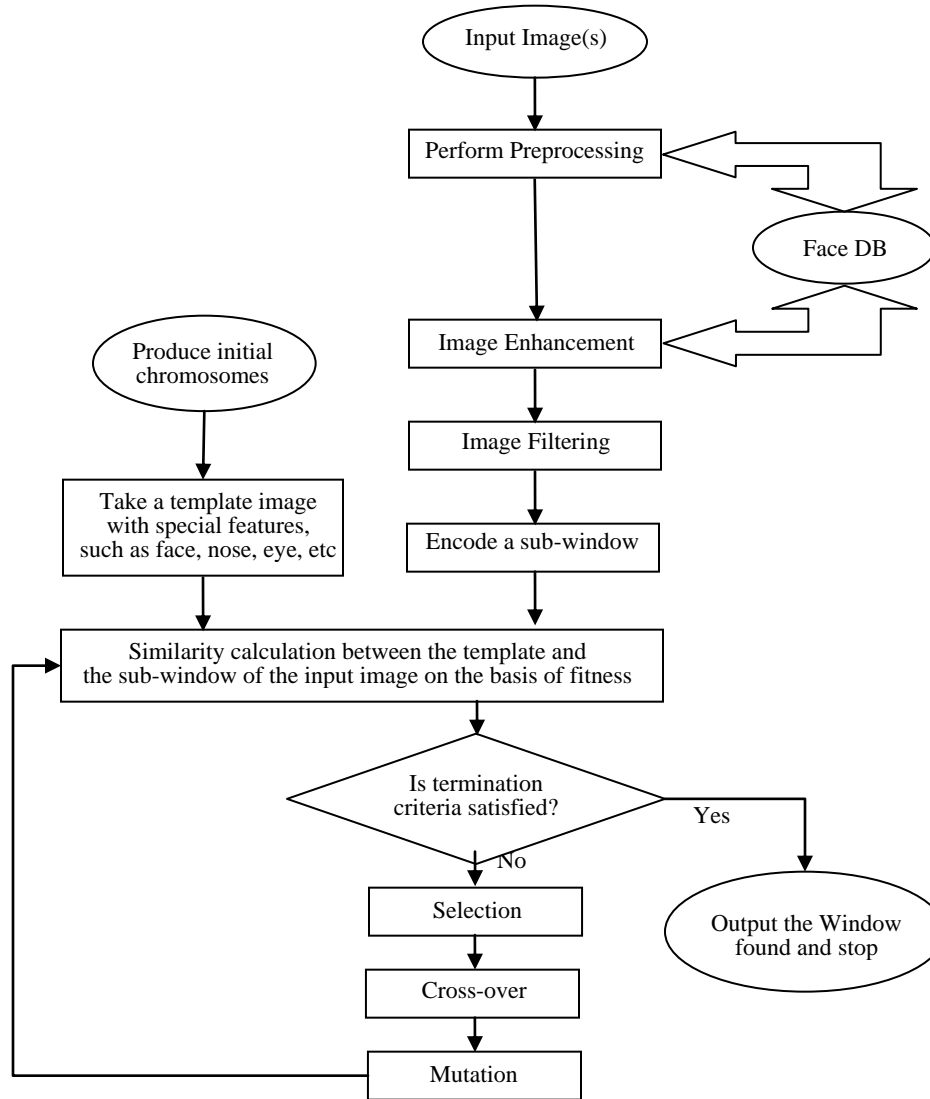


Fig. 4 Flowchart of the genetic algorithm based face detection process

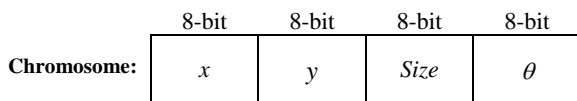


Fig. 3 Structure of a chromosome

The algorithm starts with an initial set of random solutions called the population. Each individual in the population, known as chromosome, is assigned a fitness value depending on how good its solution to the problem is. After fitness allotment, the natural selection is executed and the ‘survival of the fittest chromosome’ can prepare to breed for

To apply GA for face detection, a template of the face image obtained from averaging the gradation level of pixels of a number of similar looking face images of several persons is constructed. The template face image is then moved through the whole image to find the location where the most suitable match exists. The genetic algorithm and different genetic operations are given below.

E. Fitness Function

To determine where a selected region is a face or not, a function needs to assign a degree of fitness to each chromosome in every generation. The fitness of a chromosome

is defined as the function of the difference between the hue of the input image and that of the template image measured for the expected location of the chromosome. That is, for each chromosome  $n$ , fitness function is defined as [9], [10]:

$$f(n) = 1 - \frac{\sum_{(x,y) \in W} |f(x,y) - f_{n,t}(x,y)|}{H_{\max} \times xSize \times ySize} \quad (8)$$

where  $H_{\max}$  is the maximum hue of the image;  $xSize$  and  $ySize$  are the number of pixels in the horizontal and vertical directions of the template image,  $f$  and  $f_{n,t}$  are the hue values of the original image and the template image when it is justified for the  $n$ -th position of the chromosome, respectively.

#### F. Selection

Selection operator selects highly fit chromosomes that contribute their gene-inherited knowledge to breed for the next generation. This research uses conventional elitist selection scheme to select an elitist chromosome with the highest fitness value, which is copied directly into the new population of next generation. The other chromosomes are selected by a roulette-wheel selection process, where the selection probability of each individual is proportional to its fitness value.

#### G. Cross-over

Cross-over operator randomly chooses a crossover point where two parent chromosomes ‘break’, and then exchanges the chromosome parts after that point. As a result, two offspring are generated by combining the partial features of two chromosomes. If a pair of chromosomes does not cross over, then the chromosome cloning takes place, and the offspring are created as exact copies of each parent. This research employs single point cross-over, two point cross-over and uniform cross-over operators. The cutting points are selected randomly within the chromosome for exchanging the contents.

#### H. Mutation

Mutation operator alters a randomly selected gene of chromosome with a very low probability,  $P_M$ . For each chromosome, generate a random value between [0, 1]. If the random value is less than  $P_M$ , choose a bit at a random location to flip its value from 0 to 1 or 1 to 0. The parameter settings in our approach are shown in **Table I**.

TABLE I. PARAMETER SETTINGS

Chromosome length	32 bits
Population size	150
Number of generation	300
Cross-over probability	0.7
Mutation probability	0.01

The basic steps of the genetic algorithm are as follows:

**Step 1 Initial Population:** Generate randomly a population of chromosomes of size  $N$ :  $x_1, x_2, \dots, x_N$ . Assign the crossover probability  $P_C$  and the mutation probability  $P_M$ .

**Step 2 Evaluation Function:** Evaluate the fitness function for each chromosome in the population.

**Step 3 Selection:** Select a pair of chromosomes for mating. Use the roulette wheel selection procedure, where each chromosome is given a slice of a circular roulette wheel. The area of the slice within the wheel is equal to the chromosome fitness ratio. Obviously, the highly fit chromosomes occupy the largest areas, where the chromosomes with least fit have much smaller segments in the wheel. To select a chromosome for mating, a random number is generated in the interval [0, 100], and the chromosome whose segment spans the random number is selected.

**Step 4 Cross-over:** Produce two off-springs from two parent chromosomes.

**Step 5 Mutation:** Apply the conventional mutation operation to the population with a mutation rate  $P_M$ .

**Step 6 Termination Test:** If a predefined termination condition is satisfied, go to **Step 7**, else go to **Step 2**.

**Step 7 Preservation:** Keep the best chromosome and Stop.

### III. EXPERIMENTAL RESULTS

The effectiveness of this approach is justified using different images with various kinds of expressions captured under complex lighting conditions. When a complex image is subjected in the input, the face detection result highlights the facial part of the image, as shown in **Fig. 5**. For multiple faces, the system finds the dominant face only. Images of different persons are taken at their own working places and at different environments both in shiny and gloomy weather.





**Fig. 5** Face detection for the persons at different places under different lighting conditions  
Input images: (a1)-(a10)  
Detected face images: (b1)-(b10)

#### IV. PERFORMANCE ANALYSIS

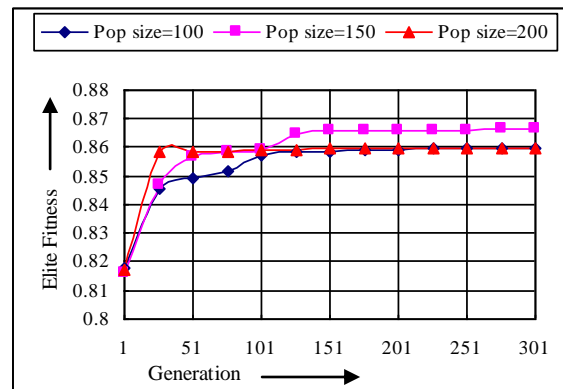
In this paper, face detection has been implemented using genetic algorithm to search for the face of a particular individual in an image. The algorithm is examined with a total of 260 images of more than 80 different persons. These images are taken under different environment: some are under sunny environment; some are under cloudy environment; some are dark images; some are bright images, and so on. In all cases, experiments have shown promising results. **Table II** summarizes the performance analyses for GA based face detection under all possible environments.

**TABLE II:** PERFORMANCE ANALYSES FOR GA BASED FACE DETECTION

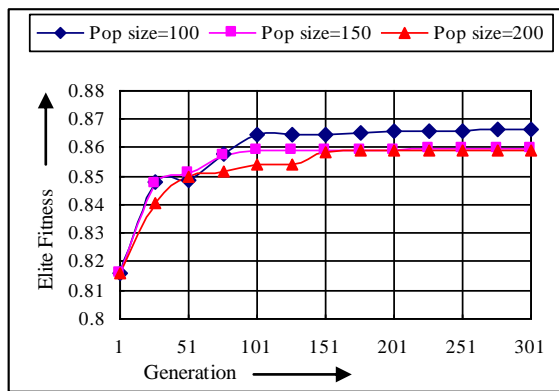
Lighting Condition		No. of Images	Correctly Detected	Success Rate
Outdoor Images	Sunny	40	39	98%
	Cloudy	60	57	95%
	Dark	27	26	96%
Indoor Images	Bright	40	39	98%
	Dark	60	57	95%
Images with Occlusion		33	30	91%

It is observed that among 260 face images only 12 faces are found false and the maximum success rate of the implemented system is 98% under sunny (Outdoor) and bright (Indoor) environments. Thus, the overall success rate of above 95%  $((248 / 260) \times 100\% = 95.38\%)$  is achieved in this method.

The genetic algorithm is examined using single point and uniform cross-over with different population sizes and the results are illustrated graphically in **Fig. 6** and **Fig. 7** respectively. **Fig. 6** reveals that larger population size offer better performance because of the larger pool of diverse schemata available in the chromosome but the inertia of larger population also boils down a problem of poorer initial. **Fig. 7** shows that smaller population size is better for uniform cross-over. So, a trade off is always taken between population size and the way of cross-over. Therefore, this research adopts single point cross-over with a population size of 150 during face detection process.



**Fig. 6** Elite fitness versus generation  
(Single point cross-over)



**Fig. 7** Elite fitness versus generation  
(Uniform cross-over)

The Experiments are performed on a Pentium IV 1.70GHz PC using Visual C++. The processing time for locating faces in a picture of size 320×240 is 2s to 10s. Some experimental results are shown in **Fig. 5**.

## V. CONCLUSION

Detection of faces and facial features using machine vision techniques has many useful applications. Though human beings accomplish these tasks countless times a day, they are still very challenging for machine vision. Most of the researchers attack this kind of problem with face localization and feature selection with frontal view faces and without facial expressions and normal lighting conditions although the variation between the images of the same face is too large due to facial expression, hair style, pose variation, lighting conditions, make-up, etc. In this paper, the effectiveness of the face detection algorithm has been tested both in simple and complex backgrounds for different types of face and non-face images of 320×240 resolution. The algorithm is capable of detecting the faces in the images with different backgrounds. A rotated human face can also be detected, even if the face is under shadow, wearing glasses, or under bad lighting conditions.

## REFERENCES

[1] H. A. Rowley, S. Beluja, T. Kanade, Neural Network-Based Face Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, Jan. 1998, pp. 23-38.  
[2] R. Hsu, M. Mottleb, A. K. Jain, Face Detection in Color Images, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, May 2002.

[3] Y. Yokoo, M. Hagiwara, Human Faces Detection Method using Genetic Algorithm, in Proc. IEEE Int. Conf. on Evolutionary Computation, pp. 113-118, May 1996.  
[4] G. Yang, T. S. Huang, Human Face Detection in a Complex Background, Pattern Recognition, Vol. 27, No. 1, pp. 53-63, 1994.  
[5] D. E. Goldberg, Genetic Algorithms in Search Optimization, and Machine Learning, Addison-Wesley, 1989.  
[6] Wang, Jianguo, Tan, Tieniu, A new face detection method based on shape information, Pattern Recognition Letters, 21(6), 1999, 463-471.  
[7] Bhuiyan, Md. Al-Amin, Ampornaramveth, Vuthichai, Muto, Shin-yo, Ueno, Haruki, Face detection and facial feature extraction, Int. Conf. on Computer and Information Technology, p. 270-274, 2001.  
[8] Yuille, Alan, Cohen, David, Hallinan, Peter, Feature extraction from faces using deformable templates, Proc. IEEE Computer Soc. Conf. on computer Vision and Pattern Recognition, p. 104-109, 1989.  
[9] Talay, A. Cagatay, An approach for eye detection using parallel genetic algorithm, V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3516. p. 1004-1007, 2005, Springer-Verlag Berlin Hiedelberg 2005.  
[10] Z. Suna, G. Bebis, R. Millerb, Object detection using feature subset selection, Elsevier, the Journal of Pattern Recognition, p. 581-589, March 2004.

## AUTHORS PROFILE



Md. Golam Moazzam completed his B.Sc. in Electronics and Computer Science from Jahangirnagar University in 1997 and M.S. in Computer Science and Engineering from the same University in 2001, respectively. He joined as a Lecturer in the Dept. of CSE, Jahangirnagar University, Savar, Dhaka, Bangladesh in 2001. His research interests include Digital Systems Design, Digital Image Processing, Artificial Intelligence, Computer Graphics, Neural Networks, Computer Vision and so on.



Ms. Rubayat Parveen completed her B.Sc (Hons) and M.Sc. in Applied Physics and Electronics from University of Dhaka, Bangladesh. She is currently doing her PhD research in the Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh. Her research interests include Artificial Intelligence, Digital Image Processing, Neural Networks, Pattern Recognition, and so on.



Md. Al-Amin Bhuiyan received his B.Sc (Hons) and M.Sc. in Applied Physics and Electronics from University of Dhaka, Dhaka, Bangladesh in 1987 and 1988, respectively. He got the Dr. Eng. degree in Electrical Engineering from Osaka City University, Japan, in 2001. He has completed his Postdoctoral in the Intelligent Systems from National Informatics Institute, Japan. He is now a Professor in the Dept. of CSE, Jahangirnagar University, Savar, Dhaka, Bangladesh. His main research interests include Image Face Recognition, Cognitive Science, Image Processing, Computer Graphics, Pattern Recognition, Neural Networks, Human-machine Interface, Artificial Intelligence, Robotics and so on.



# Ear Recognition using Dual Tree Complex Wavelet Transform

Rajesh M Bodade

Military College of Telecommunication Engineering  
Mhow, Indore, India

Sanjay N Talbar

S.G.G.S. Institute of Technology and Engineering  
Nanded, India

**Abstract**—Since last 10 years, various methods have been used for ear recognition. This paper describes the automatic localization of an ear and its segmentation from the side poses of face images. In this paper, authors have proposed a novel approach of feature extraction of iris image using 2D Dual Tree Complex Wavelet Transform (2D-DT-CWT) which provides six sub-bands in 06 different orientations, as against three orientations in DWT. DT-CWT being complex, exhibits the property of shift invariance. Ear feature vectors are obtained by computing mean, standard deviation, energy and entropy of these six sub-bands of DT-CWT and three sub-bands of DWT. Canberra distance and Euclidian distance are used for matching. This method is implemented and tested on two image databases, UND database of 219 subjects from the University of Notre Dame and own database created at MCTE, of 40 subjects which is also used for online ear testing of system for access control at MCTE. False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER) and Receiver's Operating Curve (ROC) are compiled at various thresholds. The accuracy of recognition is achieved above 97 %.

**Keywords**—Ear recognition; ear detection; ear biometrics; DT-CWT; complex wavelet transform; Biometrics; Pattern Recognition; Security; Image Processing; Bioinformatics; Computer vision.

## I. INTRODUCTION

Ear recognition has received considerably less attention than many alternative biometrics, including face, fingerprint and iris recognition. Ear-based recognition is of particular interest because it is non-invasive, and also is not affected by other factors such as mood, health, and/or clothing. Also, the appearance of an auricle (outer ear) is relatively unaffected by ageing, making it better suited for long-term identification[1]. Also, ear images can be easily taken from a distance without knowledge of the person concerned. Therefore ear biometric is suitable of surveillance, security, access control and monitoring applications.

As compared to face biometrics [2]-[4] ears have several advantages over complete face like reduced spatial resolution, a more uniform distribution of color, and less variability with expressions and orientation of the face. It's deep three-dimensional structure makes it very difficult to counterfeit. Moreover, in face recognition there can be problems of illumination variation, pose variation and facial expressions[4].

Ear was first used for recognition of human being by Iannarelli, who used manual techniques to identify ear images.

Samples of over 10,000 ears were studied to prove the distinctiveness of ear. Structure of ear does not change radically over time. The medical literature provides information that ear growth is proportional after first four months of birth and changes are not noticeable in the age group from 8 to 70 [1].

Thus, this ensures that the ear will occupy a special place in conditions requiring a high degree of perfection.

The remainder of this paper consists of: existing ear recognition techniques, localisation and normalisation of ear, feature extraction using DT-CWT, matching, experimental results and conclusions covered in Section-2, 3, 4, 5, 6 and 7 respectively.

## II. EXISTING EAR RECOGNITION TECHNIQUES

Major work on automatic ear localisation [2]-[11] has been done recently in past 10 years. Automatic ear recognition using Voronoi diagrams to take care of adverse effects of lighting, shadowing and occlusion has been presented by Burge and Burger [6]. In [9], Active Contour Model (or Snakes) is used to segment the ear from the side image of a face. Hurley, Nixon and Carter [7] have used force field transformations for ear localisation. [8][12] and [13] make use of 3-D range images to extract the ear from the image of a human. However, the tougher challenge is to detect the ear from an intensity image. A shaped model-based technique for locating human ears from face range images is proposed in [8]. In this method, the ear shape model is represented by a set of discrete 3D vertices corresponding to ear's helix and anti-helix parts. Ansari and Gupta have proposed an approach based on edges of outer ear helices by exploiting the parallelism between the outer helix curves of the ear to localize the ear[10]. Skin-color and contour information has been exploited for ear detection by Yuan and Mu [11]. In [13], authors has presented a distance transform and template based technique for automatic ear localization from a side face image. This technique first segments skin and non-skin regions in a face and then uses template based approach to find the ear location within the skin regions.

Victor et al. [4] and Chang et al. [2] have researched use of PCA and FETET for ear recognition. Moreno et al. [5] used 2D intensity images of ears with three neural net approaches for ear recognition. Hurley [14] developed an invertible linear transform which transforms an ear image into a force field by pretending that pixels have a mutual attraction proportional to

their intensities and inversely proportional to the square of the distance between them. Yan and Bowyer [15] have developed a fully automatic ear biometric system using ICP based 3D shape matching for recognition, and using both 2D appearance and 3D depth data for automatic ear extraction which not only extracts the ear image but also separates it from hair and earrings. In [16], Anupama Sana et al. presented an ear biometric system based on discrete Haar Wavelet Transform whereas Wang and Yuan [17] used Gabor wavelets and general discernment analysis. Wang Xiaoyun et al. [19] proposed block segmentation based approach whereas modular neural network architecture has been proposed by Gutierrez et al. [18].

### III. PROPOSED SYSTEM

The block diagram of proposed system is shown in Fig 1. It consist of an image acquisition module, preprocessing and automatic ear localization module, DT-CWT based feature extraction module and matching module.

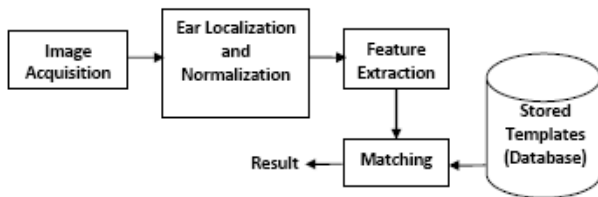


Figure 1. Block diagram of proposed system

#### A. Image Acquisition Module

Sony DSC-HX1 (15 Megapixel and optical zoom of 20x) camera is used for image acquisition. Database of 240 images of 40 subjects for left and right ears created at MCTE and UND database is also used.

#### B. Preprocessing and ear localisation module

The raw image is not suitable for feature extraction due to its large background. Thus some pre-processing is required to make it suitable. The important steps involved are: gray scale conversion, ear detection and scale normalization which has been presented in section III .

#### C. Feature extraction module

After successful ear localization, features are extracted using DT-CWT. The details of the same are stated in section IV.

#### D. Matching module

Energy, Entropy, Mean and Standard Deviation of each sub-band of DT-CWT is calculated to create a feature vector. Euclidian distance and Canberra distance as given by equation (1) and (2) are used as similarity measures for matching the feature vectors of the test image with that of the images stored in database(1:N) match.

Euclidian

$$d_E(x,y) = \sqrt{\sum_{i=1}^d (x_i - y_i)^2} \quad (1)$$

Distance:

Canbera Distance:

$$d_c(x,y) = \sum_{i=1}^d \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad (2)$$

### IV. AUTOMATIC EAR LOCALISATION

This was a very challenging task as most of the work carried out on this aspect is in experimental stage. The algorithm so designed includes the finer points of various algorithms and additional measures to try and further enhance and improve the ear localization results. The algorithm works as under.

(i) Take a side face image of an individual (under varied background and lighting conditions).

(ii) Creates the color transformation structure from RGB to LAB and apply it to the test image. The input is RGB which consists of three layers, Red, Green, and Blue. The output is LAB which consists of L\*, a\*, and b\* layers. L\* represents brightness and has range from 0 to 100. a\* represents degree of redness-greenish, having range from -100 to 100. (Positive values for redness, and negative values for greenish). b\* represents degree of yellowish-bluish and has the same ranges as a\*.

(iii)The technique is adaptable for different skin colors and various lighting conditions. Since RGB representation of color images is not suitable for characterizing skin-color, it first converts the RGB color space to chromatic color space, and then uses the chromatic color information for further processing. In RGB color space, the triple components (R, G & B) represent not only color but also luminance which may vary across a person's face due to the ambient lighting and is not a reliable measure in separating skin from non-skin regions. Luminance can be removed from the color representation in the chromatic color space.

(iv)Apply threshold to the resultant image and convert to binary.

(v) A few results under varied background conditions are shown from Figure 2 to Figure 4.

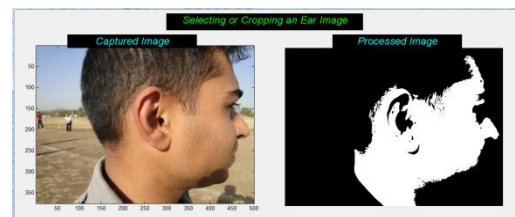


Figure 2. Side face image with open sky background

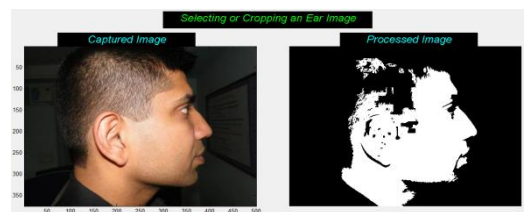


Figure 3. Side face image with class room background

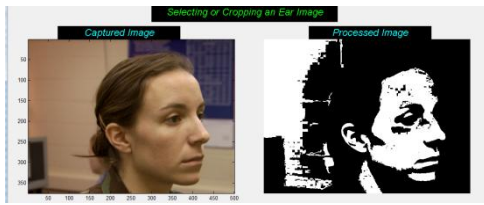


Figure 4. Side face image with class room background from UND database

(vi) Once the background is removed from an image, we determine the nose tip of the subject and assuming an approximate distance between the subject's nose tip & ear pit and the average size of human ear, we crop the ear part from the side face of an image.

(vii) Thereafter we take a four pronged approach to determine the edge variations from top (at Helix), from side (at Concha and at Helix posterior) and from bottom (at Lobulo) to determine the ear edges and thus crop out the exact ear from an image.

(viii) A two resulting ear image localized automatically from side face images are shown in Figure 5.

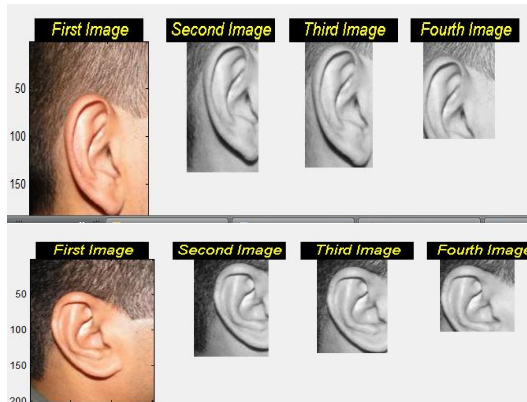


Figure 5. Automatic cropping for Ear localization

(ix) The cropped ear images may be of varying sizes and so the feature set of images may also vary. Hence the images are normalized to a constant size by resizing technique (Bilinear method). If two images are of different sizes, e.g. one is of size  $(x', y')$  and the other is of size  $(x'', y'')$ , then the two images are mapped to a constant size.

$$\begin{aligned} I(x', y') &= I(x, y) \\ I(x'', y'') &= I(x, y) \end{aligned}$$

## V. FEATURE EXTRACTION

DT-CWT is formulated by Kingbury and Selesnick[20], [21] using two trees (real and imaginary trees) of DWT with different filter real coefficients for imaginary tree filters designed from the coefficients of real tree filters to overcome the limitations (Poor directionality, shift variance and absence of phase information) of DWTs. Kokare et al. [22] has rotated the 2-D non-separable filters to obtain 2-D non separable Rotated Complex Wavelet Filters (RCWF) and used DT-CWT and RCWF for content based texture image retrieval with excellent retrieval results compared to many other methods. Same concept with different set of filters has been used by

Bodade et al. [23], [24] for iris recognition and obtained a comparable result with the benchmarked Daughman's method[25]. The details of DT-CWT and feature extraction are stated in following subsections.

### A. Dual Tree Complex Wavelet Transform

In dual-tree, two real wavelet trees are used as shown in Figure 6, each capable of perfect reconstruction (PR). One tree generates the real part of the transform while the other is used in generating complex part[20]. As shown,  $\{H_0(z), H_1(z)\}$  is a Quadrature Mirror Filter (QMF) pair in the real-coefficient analysis branch. For the complex part,  $\{G_0(z), G_1(z)\}$  is another QMF pair in the analysis branch.

All filter pairs are orthogonal and real-valued. It has been shown [21] that if filters in both trees be made to be offset by half-sample, two wavelets satisfy Hilbert transform pair condition and an approximately analytic wavelet is given by Eq(1).

$$\psi(x) = \psi_h(x) + j\psi_g(x) \quad (1)$$

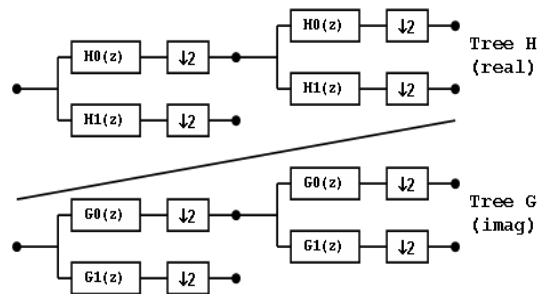


Figure 6. Selesnick's Dual Tree DWT

Thus, if  $G_0(\omega) \cong H_0(\omega) \times e^{-i\theta(\omega)}$  and  $\theta(\omega) \cong \omega/2$

$$\begin{aligned} \text{then } \psi_g(\omega) &\cong -j\psi_h(\omega), \omega > 0 \\ &\cong j\psi_h(\omega), \omega < 0 \end{aligned} \quad (2)$$

From Eq(1) and (2), low pass filters after the first stage and at first stage respectively are given by Eq(3):

$$\begin{aligned} g_0(n) &= h_0(n-0.5) \text{ and} \\ g_1(n) &= h_0(n-1) \end{aligned} \quad (3)$$

Similar relations also hold true for high pass filters of both the trees.

In this algorithm, (10, 10)-Tap near orthogonal wavelet filters are used in first stage and 'db7' filters are used for higher stages in the real tree (i.e.  $h_0$  and  $h_1$ )[20].The imaginary low pass filter is derived from the above half sample delayed condition. The high pass filter is the quadrature-mirror filter of the low pass filter. The reconstruction filters are obtained by time reversal of decomposition filters. All the filters used are of same length based on Selesnick's approach [20], [21], [23], [24] unlike Kingsbury's approach.

The 2D separable DWT can be written in terms of 1D scaling functions ( $\phi$ ) and wavelet functions( $\psi$ ) as:

$$\begin{aligned} \psi^0(x, y) &= \phi(x)\psi(y) \\ \psi^{90}(x, y) &= \psi(x)\phi(y) \\ \psi^{\pm 45}(x, y) &= \psi(x)\psi(y) \end{aligned} \quad (4)$$

Oriented non-separable 2D wavelet transform is derived by combining the sub-bands of two separable 2D DWTs. The pair of conjugate filters are applied to two dimensions (x and y), which can be expressed by Eq(5) as given below:

$$(h_x + jg_x)(h_y + jg_y) = (h_x h_y - g_x g_y) + j(h_x g_y + h_y g_x) \quad (5)$$

The filter bank structure of 2D DT CWT, to implement Eq(5) is shown in Figure 7.

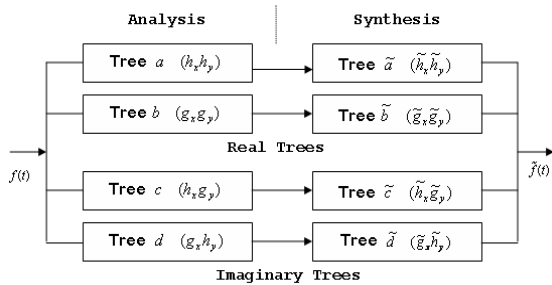


Figure 7. Filter bank structure of 2D DT CWT

Tree-a and Tree-b is combined to compute the Real part of Eq(5) i.e Real (2D DWT) tree of CWT as shown in Figure 8. Similarly, Imaginary (2D DWT) tree of CWT can be obtained from tree-c and tree-d i.e.  $(h_x g_y - g_x h_y)$ , as per Eq(5).

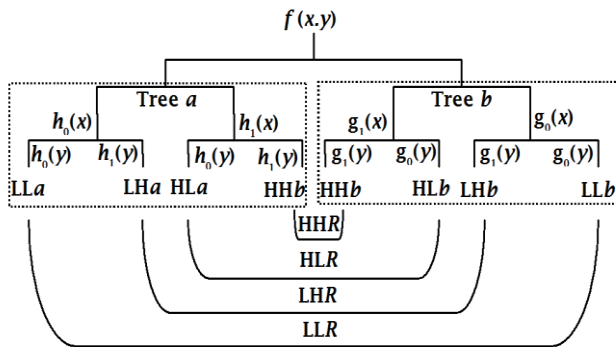


Figure 8: Formation of Real Tree DT CWT

Thus, the decomposition for each mode is performed in a standalone mode, in subsequent stages i.e. total of 6 detailed coefficients are derived at each stage; three each for real and imaginary trees. When 3-stage decomposition is performed, at each stage, coefficients are oriented towards their respective

directions as stated in Eq(4). Following six wavelets, as given by Eq(6), are used to obtain oriented 2-D separable wavelets [20]:

$$\begin{aligned} \psi_{1,1}(x, y) &= \phi_h(x)\psi_h(y), & \psi_{2,1}(x, y) &= \phi_g(x)\psi_g(y), \\ \psi_{1,2}(x, y) &= \psi_h(x)\phi_h(y), & \psi_{2,2}(x, y) &= \psi_g(x)\phi_g(y), \\ \psi_{1,3}(x, y) &= \psi_h(x)\psi_h(y), & \psi_{2,3}(x, y) &= \psi_g(x)\phi_g(y), \end{aligned} \quad (6)$$

where,  $\psi_{1,i}$  corresponds to the coefficients derived from the real tree and  $\psi_{2,i}$  corresponds to the coefficients derived from the imaginary tree. They can thus be combined by Eq(7) to form complex wavelet coefficients.

$$\begin{aligned} \psi_i(x, y) &= \frac{1}{\sqrt{2}}(\psi_{1,i}(x, y) - \psi_{2,i}(x, y)), \\ \psi_{i+3}(x, y) &= \frac{1}{\sqrt{2}}(\psi_{1,i}(x, y) + \psi_{2,i}(x, y)) \end{aligned} \quad (7)$$

Normalization by  $1/\sqrt{2}$  is used so that the sum/difference operation constitutes an ortho-normality. These six wavelet sub-bands of the 2-D DT-CWT are strongly oriented in  $\{+15^\circ, +45^\circ, +75^\circ, -15^\circ, -45^\circ, -75^\circ\}$  direction as shown in fig(5) by red lines and it captures image information in those directions.

Thus, in particular, 2D dual-tree wavelets are not only approximately analytic but are also oriented and shift invariant because of their analytic structure[20].

The impulse responses of three 2-D sub-bands (2-D non separable filters for detailed coefficients) of DWT and six sub-bands (2-D non separable filters for detailed coefficients) of DT-CWT are shown in Figure 9.

### B. Feature Extraction

Ear analysis using DWT provides singularities (edges) in only three directions (0, 45, 90) and without phase information which is improved by finding the singularities, with phase information, in six directions (0, +/-15, +/-30, +/-45, +/-60, +/-75, 90) and at many freq bands using DT-CWT to achieve shift invariant features for better accuracy and efficiency at less computational cost as compared to existing methods.

From the detailed study of prevalent techniques already employed for ear recognition, it is realized that nobody had made use of Complex Wavelets for ear recognition. This realization laid the foundation of utilizing this approach to determine whether or not the said approach can further enhance and improve the recognition rates already achieved by other methods. As an advantage of using CWT vis-à-vis DWT, it is imperative to employ Dual Tree - Complex Wavelet Transform (Selesnick) (DT-CWT(S)) for this work.

The DT-CWT(S) algorithm is used to design and implement the Dual Tree structure (up to Level 2) using MATLAB, employing first and second stage low and high pass filter coefficients given by Selesnick. The impulse responses of three 2-D sub-bands (2-D non separable filters for detailed coefficients) of DWT and six sub-bands (2-D non separable filters for detailed coefficients) of DT-CWT are shown in Figure 9.

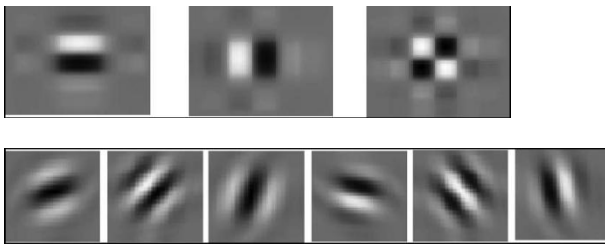


Figure 9. Impulse responses of sub-bands of DWT and DT-CWT.

DT-CWT has 06 directional wavelets oriented at angles of  $\pm 15, \pm 45, \pm 75$  in 2-Dimension. We get these six directions by passing the 2-D signal (image) through a real tree structure using the filter coefficients of both real and imaginary trees. The wavelet coefficients of each image which formed part of the Training Database were thus obtained in all the six directions and stored for further matching and testing. These directions can be seen clearly from the Figure 10 which represents the Level 1 decomposition of an image.

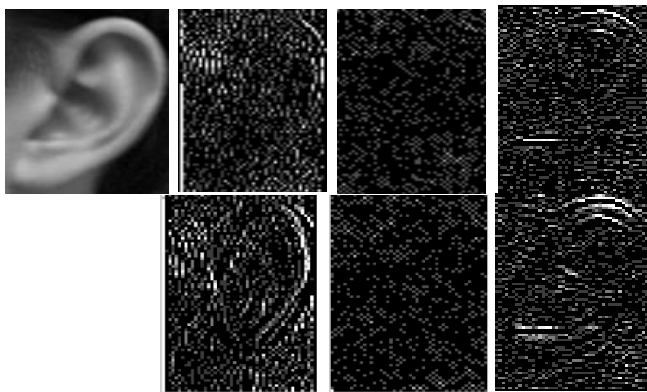


Figure 10. Real and imaginary tree wavelet sub-band images

## VI. EXPERIMENTAL RESULTS

All the Training images of both the databases (MCTE database of 240 images of 40 subjects for right and left ears and UND database of 219 subjects under J-collection and G-collection) are processed and their respective wavelet coefficients at Level 1 and Level 2 are calculated. Energy, Entropy, Mean and Standard Deviation of each image's wavelet coefficient are then calculated and stored in an MS Access database. Thereafter images from the Test Set and random images were matched with these stored values using Euclidean and Canberra distance matching techniques and results for False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER) and Receiver's Operating

Curve (ROC) compiled at various thresholds. All the results are stated in Table 1.

Figure 11 to 14 shows the FAR, FRR and ROC of best and worst case of Canberra distance and best and worst case of Euclidian distance when tested on database with following details.

Name Of Data-Base : UND – Collection G

No of Images in the Training database: 20

No of Images in Test Database: 90

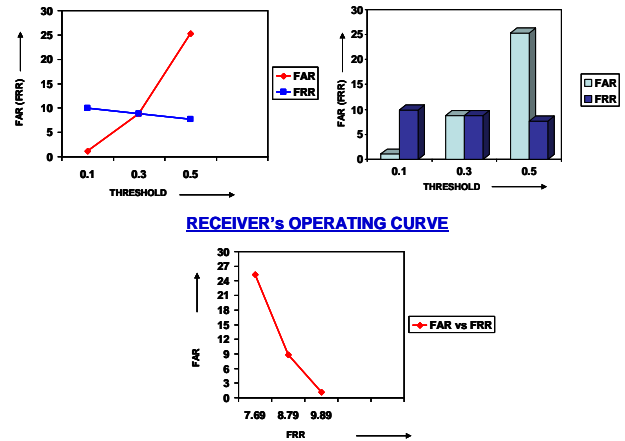


Figure 11. Results using Canberra distance and feature vector of energy only (Worst case of Canberra Distance)

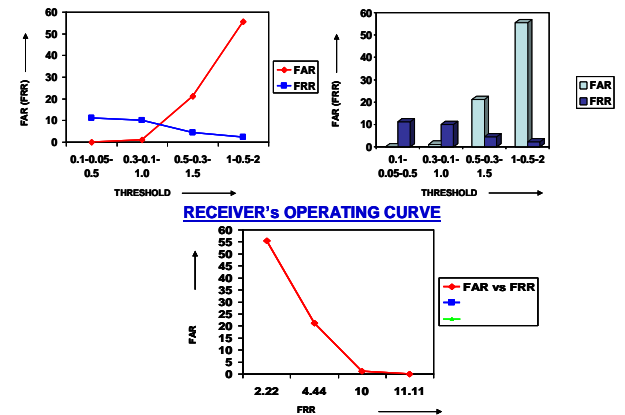


Figure 12. Results using Canberra distance and feature vector of energy + std deviation + entropy (Best case of Canberra Distance)

Table 1. Compiled results of Avg FAR, Avg FRR and Avg Recognition rate using Canberra and Euclidian distance for different feature vectors

Feature Vector / Distance	Canberra Distance			Euclidian Distance		
	Avg FAR	Avg FRR	Recognition Rate	Avg FAR	Avg FRR	Recognition Rate
Energy only	8.79 %	8.79 %	91.21 %	16.66 %	5.55 %	83.34 %

Entropy only	6.59 %	8.79 %	93.41 %	12.22 %	6.66 %	87.88 %
Standard Deviation only	8.79 %	8.79 %	91.21 %	21.11 %	4.44 %	78.89 %
Energy+ Entropy	3.29 %	8.79 %	96.71 %	3.33 %	8.88 %	96.67 %
Energy+ Standard Deviation	5.49 %	4.39 %	94.51 %	4.44 %	6.66 %	95.56 %
Entropy+ Standard Deviation	5.49 %	8.79 %	94.51 %	6.66 %	4.44 %	93.34 %
Energy + Entropy + Standard Deviation	2.19 %	7.69 %	97.81 %	3.33 %	7.77 %	96.67 %

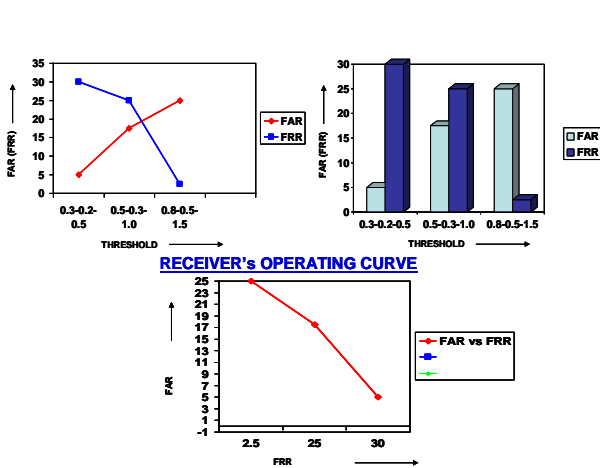


Figure 13. Results using Euclidian distance and feature vector of Std Deviation only (Worst case of Euclidian Distance)

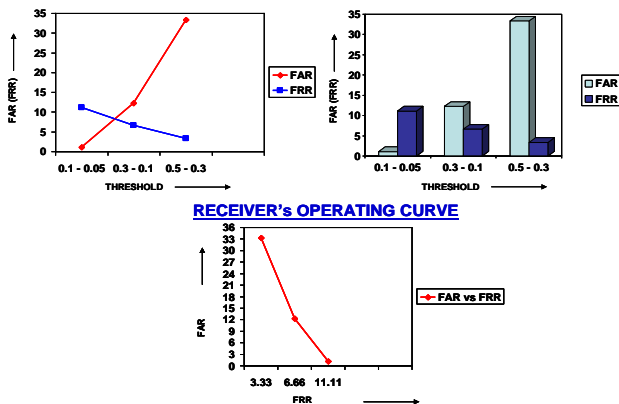


Figure 14. Results using Euclidian distance and feature vector of energy + std deviation + entropy (Best case of Euclidian Distance)

The maximum recognition rate of 81% is obtained when DWT is used for feature extraction and Canberra distance is used as similarity metric for combined vector of energy, std. deviation and entropy. FAR, FRR and ROC for it is shown in figure 15.

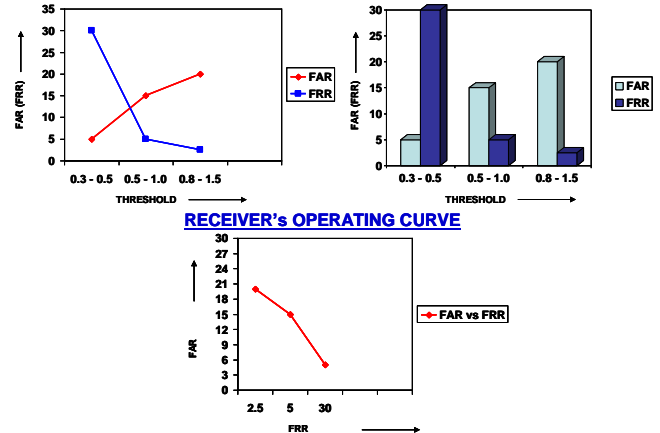


Figure 15. Results of DWT using Canberra distance and feature vector of energy + std deviation + entropy (Best case of DWT)

## VII. CONCLUSIONS

The authors have introduced a new 2D DT CWT for ear recognition first time because of its ability to capture shift invariant features in 06 orientations. The experimental results have demonstrated the effectiveness of the proposed method in terms of improving the recognition rate.

Canberra distance has shown better results than Euclidian distance because it normalises the individual feature components before finding the distance between the two images.

The best recognition rate of over 97% has been achieved using Canberra distance when feature vectors of energies, standard deviation and entropy of sub-bands of DT-CWT are used together.

The authors are working on improving the recognition rate by using the RCWF [22], [23], [24] in combination with DT-CWT to obtain features in 12 orientations (06 by DT-CWT and 06 by RCWF).

## REFERENCES

- [1] Alfred Iannarelli, "Ear Identification," Forensic Identification Series, Paramont Publishing Company, Fremont, California, 1989.
- [2] K. Chang., K.W. Bowyer, S. Sarkar, and B. Victor, "Comparison and Combination of Ear and Face Images in Appearance-Based Biometrics," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 9, pp. 1160-1165, September 2003.

- [3] D.J. Hurley, M.S. Nixon, J.N. Carter, "A New Force Field Transform for Ear and Face Recognition," Proc. IEEE 2000 International Conference on Image Processing (ICIP 2000), IEEE Press, pp. 25-28.
- [4] B. Victor, K.W. Bowyer, S. Sarkar, "An evaluation of face and ear biometrics" Proc. International Conference on Pattern Recognition, August 2002, pp. 429-432.
- [5] B. Moreno, A. Sánchez, J.F. Véllez, "On the Use of Outer Ear Images for Personal Identification in Security Applications," Proc. IEEE 33 Annual International Carnahan Conference on Security Technology, 1999, pp. 469-476.
- [6] M. Burge and W. Burger, "Ear biometrics in computer vision," Proc. International Conference on Pattern Recognition, (ICPR' 00), vol. 02, 2000, pp. 822-826.
- [7] D. J. Hurley, M. S. Nixon, and J. N. Carter, "Automatic ear recognition by force field transformations," Proc. IEEE Colloquium: Visual Biometrics, 2000, pp. 8/1-8/5.
- [8] H. Chen and B. Bhanu, "Human ear detection from side face range images," Proc. International Conference on Pattern Recognition, (ICPR' 04), vol. 3, IEEE Computer Society Press, 2004, pp. 574-577.
- [9] L. Alvarez, E. Gonzalez, and L. Mazon, "Fitting ear contour using an ovoid model," in IEEE International Carnahan Conference on Security Technology, (ICCSST' 05), 2005, pp. 145-148.
- [10] S. Ansari and P. Gupta, "Localization of ear using outer helix curve of the ear," Proc. (International Conference on Computing: Theory and Applications, ICCTA' 07), IEEE Computer Society Press, 2007, pp. 688-692.
- [11] L. Yuan and Z.-C. Mu, "Ear detection based on skin-color and contour information," Proc. International Conference on Machine Learning and Cybernetics, (ICMLC' 07), vol. 4, (Hong-Kong), 2007, pp. 2213-2217,
- [12] Hui Chen and Bir Bhanu, "Human Ear Recognition in 3D", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol 29, No 4, April 2007
- [13] Surya Prakash, Umarani Jayaraman and Phalguni Gupta, "Ear Localization from Side Face Images using Distance Transform and Template Matching", Proc. IEEE Int'l Workshop on Image Processing Theory, Tools and Applications, (IPTA 2008), Sousse, Tunisia, Nov 2008 pp. 1-8.
- [14] D. Hurley, M. Nixon, and J. Carter, "Force Field Energy Functionals for Ear Biometrics" Computer Vision and Image Understanding, vol.98, 2005, pp.491-512.
- [15] Ping Yan and Kevin W. Bowyer, "Biometric Recognition Using 3D Ear Shape" IEEE Transactions on pattern analysis and machine intelligence, vol. 29, no. 8, August 2007.
- [16] A. Sana, F. Gupta and R. Prukait, "Ear Biometrics: A new approach," Proc. Sixth International Conference on Advances in Pattern Recognition, Jan 2007, pp.46-50.
- [17] Xiaoyun Wang and Weiqi Yuan, "Gabor wavelets and General Discriminant analysis for ear recognition," Proc. 8th World Congress on Intelligent Control and Automation (WCICA 2010), IEEE Press, 2010, pp. 6305 - 6308
- [18] L. Gutierrez, P. Melin, and M. Lopez, "Modular neural network integrator for human recognition from ear images," Proc. International Joint Conference on Neural Networks (IJCNN 2010), IEEE Press, 2010, pp. 1-5.
- [19] Wang Xiaoyun and Yuan Weiqi, "Human ear recognition based on block segmentation," Proc. International Conf on Machine Learning and Cybernetics, 2009, vol. 4, pp. 2418 - 2422.
- [20] Ivan W. Selesnick, Richard G. Baraniuk, and Nick G. Kingsbury, "The Dual Tree Complex Wavelet Transform: A Coherent Framework for Multiscale Signal and Image Processing", IEEE Signal Processing Magazine, Nov 2005, pp. 123-151.
- [21] N.G. Kingsbury, "The dual-tree complex wavelet transform: A new technique for shift invariance and directional filters," in Proc. 8th IEEE DSP Workshop, Utah, Aug. 9-12, 1998, paper no. 86.20.
- [22] M. Kokare, P.K. Biswas, and B.N. Chatterji, "Rotation invariant texture features using rotated complex wavelet for content based image retrieval," in Proc. IEEE Int. Conf. Image Processing, Singapore, Oct. 2004, vol. 1, pp. 393-396.
- [23] Rajesh Bodade and Sanjay Talbar, "Iris Recognition Using Multi-Directional Wavelets: A Novel Approach" Journal of Advances in Engineering Sciences Sect. C (3), July-September 2008, (special issue on Image Processing) ISSN: 0973-9041
- [24] R. Bodade and S. Talbar, "Iris Recognition using Combination of Dual Tree Rotated Complex Wavelet and Dual Tree Complex Wavelet" Proc. IEEE International Conference on Communication-2009 (ICC-2009), IEEE Press, June 2009.
- [25] J. Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence," IEEE Trans. PAMI, vol. 15, no. 11, Nov. 1993, pp. 1148-1161.

# SUCCESeR: Simple and Useful Multi Color Concepts for Effective Search and Retrieval

Satishkumar L. Varma

Department of Computer Engineering  
Pillai's Institute of Information Technology  
Navi Mumbai, India

Sanjay N. Talbar

Electronics and Telecommunication Engineering  
SGGS Institute of E&IT  
Nanded, India

**Abstract**—The image quality depends on level of intensities used in images. Image consists of various types of objects. Objects in the images are distinguishable because of various intensity levels used. The concentration of intensity levels so called energy can be extracted from image using discrete cosine transform (DCT). In this paper we apply DCT 8x8 block coefficients separately on three different color planes of three different color models namely RGB, HSV and YCbCr. The different elements of ten DCT coefficient matrices are used to form feature vectors. The different feature vectors are formed using these ten elements. These feature vectors are used to index all images in the database. The system was tested with Coral Image database containing 1000 natural images having 10 different classes of images. The image retrieval using these indices is giving comparatively better results.

**Keywords**—color model; discrete cosine transform; image indexing; image retrieval;

## I. INTRODUCTION

With the explosive growth of the need of the information, the large amount of storage, manipulation and transmission of data also needs to be kept indexed.

Content Based Image Retrieval (CBIR) is a technique which uses visual contents in an image to retrieve images. The visual contents are called features. These features are used to search images from large scale image databases according to users' requests in the form of a query image [1], [8], [9].

The system is tested with Coral Image database used in [8] containing 1000 natural images having 10 classes of images. These images have also been used in SIMPLicity [7], Columbia VisualSEEK and WebSEEK [16], and Stanford WBIIS [16].

### A. Related Work and Systems

In the commercial domain, IBM QBIC [19] is one of the earliest systems. Recently, additional systems have been developed at IBM T.J. Watson [20], VIRAGE [21], NEC AMORA [22], Bell Laboratory [23], and Interpix. In the academic domain, MIT Photobook [24], [25] [26] is one of the earliest. Berkeley Blobworld [27], Columbia VisualSEEK and WebSEEK [16], CMU Infromedia [28], UCSB NeTra [29], UCSD [30], University of Maryland [31], Stanford EMD [32], and Stanford WBIIS [16] are some of the recent systems. For

example, [2] describes a method for image retrieval purely based on color and texture.

Ngo et al. developed an image indexing algorithm via reorganization of DCT coefficients in Mandala domain, and representation of color, shape and texture features in compressed domain [13]. Feng et al. proposed an indexing approach by direct extraction of statistic parameters in DCT domain to combine the nature of texture and shape into an integrated feature [14]. Ladert and Guerin-Dugue proposed an algorithm for extracting the global distribution of local dominant orientation from DCT domain [15].

### B. Contribution and Paper Organization

The major contribution of this paper is not only to suggest feature extraction method to index images for faster and efficient digital image retrieval but also propose a clearly designed architecture that can be used for image classification.

The paper is organized as follows. In the following Section II the three color spaces and their conversion from one space to another is presented. The very next Section III explains the proposed CBIR model for image indexing and retrieval. The experimental results are discussed in Section IV. The summary and conclusion of the paper is presented in Section V. The future scope is suggested in Section VI.

## II. COLOR MODEL

Color is perhaps the most expressive of all the visual features and has been extensively studied in the image retrieval research during the last decade.

### A. RGB Color Model

The quantity of light is generated by the pixel results from the sum of Red, Green and Blue that are stated by the computer. The color model is shown in Fig. 1.

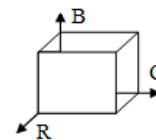


Figure 1. RGB Color Space



### B. HSV Color Model

Let  $r, g, b$   $[0, 1]$  be the red, green, and blue coordinates, respectively, of a color in RGB space. Let  $\max$  be the greatest of  $r, g,$  and  $b,$  and  $\min$  the least. To find the hue angle  $h$   $[0, 360]$  for either HSI or HSV space, compute:

$$h = \begin{cases} 0 & \text{if } \max = \min \\ \left(60^\circ \times \frac{g-b}{\max-\min} + 0^\circ\right) \bmod 360^\circ, & \text{if } \max = r \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ, & \text{if } \max = b \end{cases} \quad (1)$$

HSI and HSV have the same definition of hue, but the other components differ. The values for  $s$  and  $v$  of an HSV color are defined as follows:

$$s = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max - \min}{\max} = 1 - \frac{\min}{\max}, & \text{otherwise} \end{cases} \quad (2)$$

$v = \max$

The HSV color space [2] is a popular choice for manipulating color. The HSV color space is developed to provide an intuitive representation of color and to approximate the way in which humans perceive and manipulate color. RGB to HSV is a nonlinear, but reversible, transformation. The hue (H) represents the dominant spectral component—color in its pure form, as in green, red, or yellow.

### C. YCbCr Color Model

RGB to YCbCr [2] conversion is the most commonly used color coordinate system for the compression of image and video signals. Y is the luminance component and Cb and Cr are the chrominance components. The primary red, green, blue inputs (R, G, and B) can be converted as follows:

$$\begin{aligned} Y &= 0.299 * R + 0.587 * G + 0.114 * B \\ Cb &= -0.169 * R - 0.331 * G + 0.500 * B \\ Cr &= 0.500 * R - 0.419 * G - 0.081 * B \end{aligned} \quad (3)$$

## III. PROPOSED CBIR

In this paper, we discuss the design and implementation concept of multi color spaces used in our CBIR system for picture libraries. An experimental system, the SUCCE-SeR (Simple and Useful Multi Color Concepts for Effective Search and Retrieval) system, has been developed to validate the methods.

### A. System Architecture

The generalized architecture of the SUCCESeR retrieval system is presented in Fig. 2. Consider  $s$  number of color space with  $p$  number of color planes in each color space. In each color plane  $n \times n$  square sub blocks are constructed. There are  $b$  sub blocks in each  $p$  plane. On every  $b$  sub blocks the selected energy compaction transform is applied.

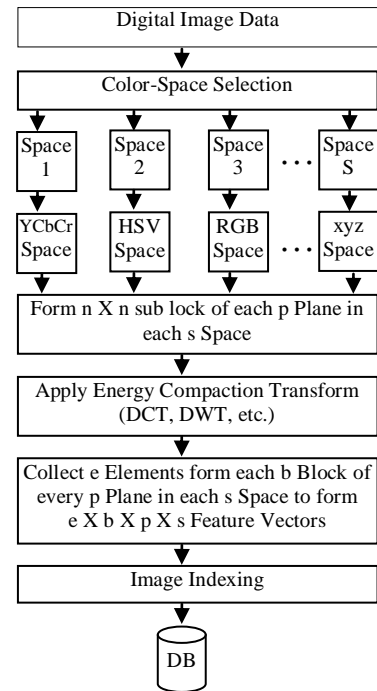


Figure 2. The architecture of feature indexing process

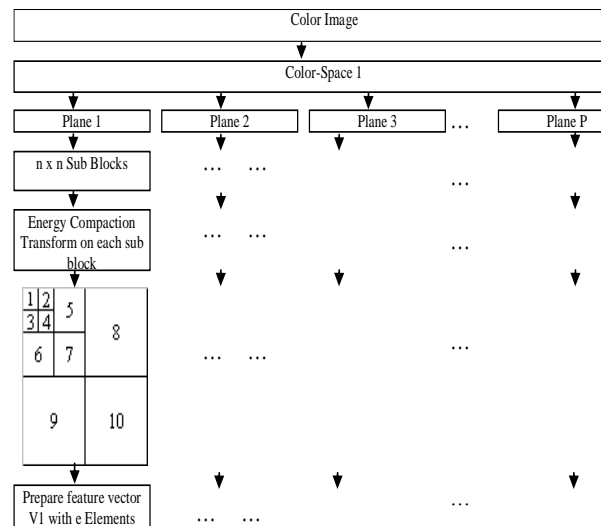


Figure 3. The plane-wise architecture of feature indexing process

The  $e$  number of elements in each  $b$  sub block is considered to form feature vector. So, the feature vector can be form by selecting  $e$  elements from each  $b$  sub blocks of each  $p$  plane in every  $s$  space. The different set of  $e$  elements is formed to test the retrieval results. The interface program helps to select different elements for each color space to check the retrieval results. For different color space the procedure remains the same. The procedure is presented in Fig. 3.

For a given color space, the feature vector  $V1$  can be formed by in manner given in Fig. 3.

B. Feature Extraction and Indexing

To be precise the different color models used for extracting features in SUCCESeR system are RGB, HSV and YCbCr. As we know there are three color planes in each color space. Energy in image is due to color component. The energy of the image is extracted using discrete cosine transform (DCT). Discrete cosine transform is applied on each 8x8 sub blocks of each color plane in each color space of every image in the database. The maximum energy of the image is concentrated in DC component of DCT and it reduces as we move down diagonally in each sub block.

1	2	5	8
3	4		
6		7	
9		10	

Figure 4. The 10 elements in of a 8x8 DCT coefficient

The 10 compiled elements shown in Fig. 4 from each 8x8 DCT coefficients for each color plane are used in the process of forming feature vectors. Two feature vectors are used to for image indexing. The first feature vector is formed by selecting only one element i.e. first element in each color plane; the second feature vector is formed by selecting first four elements of each color plane, and so on. This is shown in Fig. 5. To understand better the above concept is put in another form which is given in Fig. 6.

Vector	Description of Vectors	Color Space		
		RGB	HSV	YCbCr
V <sub>111</sub>	Vector elements include one coefficient form every 8x8 sub-block in each color space	1	1	1
V <sub>112</sub>	Vector elements include one coefficient in RGB and HSV, and two coefficient in YCbCr form every 8x8 sub-block	1	1	1 2
V <sub>121</sub>	Vector elements include one coefficient in RGB and YCbCr, and two coefficient in HSV form every 8x8 sub-block	1	1 2	1
V <sub>122</sub>	Vector elements include one coefficient in RGB, and two coefficient in HSV and YCbCr form every 8x8 sub-block	1	1 2	1 2

Figure 5. Description of elements in a vector

Feature Vector	Number of Elements Used in each Feature Vector Formation (Out of 10 DCT coefficient in every plane of 3 color spaces)								Number of Elements in Each Feature Vector	
	r	g	b	h	s	v	y	cb		cr
Vector V <sub>111</sub>	1	1	1	1	1	1	1	1	1	9
Vector V <sub>112</sub>	1	1	2	1	1	2	1	1	2	12
Vector V <sub>121</sub>	1	2	1	1	2	1	1	2	1	12
Vector V <sub>122</sub>	1	2	2	1	2	2	1	2	2	15

Figure 6. The elements used in vector formation

C. Equations

The chi-square distance measure [17] between features of query object and database object is used in [18] to match and display images. It is given in (4).

$$dissimilarity = \sum_{i=1}^{12} \left( \frac{Fq(i) - Ft(i)}{Fq(i) + Ft(i)} \right)^2 \tag{4}$$

The dissimilarities are sorted in descending order. Lower the dissimilarity closest the match. The closest matched images are our candidate images and they are displayed using proposed SUCCESeR System interface.

IV. RESULT AND ANALYSIS

In this paper, the implemented CBIR system using different color models can be deployed to retrieve images in picture libraries.

A. Image Database

The SUCCESeR system results were compared with the existing SIMPLIcity system results given in [7] which was already evaluated and compared with WBIIS [16] based on a same subset of the COREL database, formed by 10 image categories shown in Table I (a), each containing 100 pictures used in [7] and [16]. Within this database, it is known whether any two images are of the same category.

The SUCCESeR system was also evaluated based on a subset of the COREL database, formed by 10 image categories shown in Table I (b), each containing 100 pictures. In particular, a retrieved image is considered a match if and only if it is in the same category as the query. This assumption is reasonable since the 10 categories were chosen so that each depicts a distinct semantic topic.

Table I. COREL Categories of Images Tested

ID	Category Name	ID	Category Name
1	Africa people and villages	1	Butterfly
2	Beach	2	Beach
3	Landscape with buildings	3	Landscape with buildings
4	Buses	4	Buses
5	Dinosaurs	5	Dinosaurs
6	Elephants	6	Elephants
7	Flowers	7	Flowers
8	Horses	8	Horses
9	Mountains and glaciers	9	Mountains and glaciers
10	Food	10	Food

(a)

(b)

B. Evaluation of the System

To provide numerical results, we tested 30 sample images chosen randomly from ten categories, each containing three of the images. A retrieved image is considered a match if it belongs to the same category of the query image. The categories of images tested are listed in Table I. Most categories simply include images containing the specified objects. The precision is calculated using (5).

$$\text{Precision} = \frac{\text{Number of relevant documents retrieved}}{\text{Total number of documents retrieved}} \quad (5)$$

C. Performance of SUCCESeR versus Simplicity

For each of the ten image categories shown in Table I (a), the average precision performance comparison of our SUCCESeR system (using feature vector V111) with existing SIMPLIcity [7] system based on the three sample images are plotted in Fig. 7.

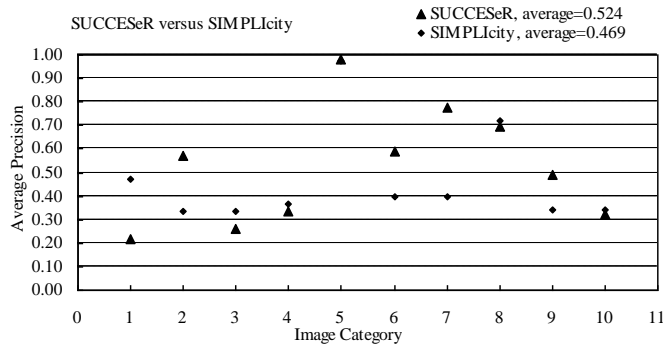


Figure 7. The average precision using vector V<sub>111</sub>

For each of the ten image categories shown in Table I (a), the average precision comparison based on the three sample images using feature vectors V112 and V122 are also plotted in Fig. 8 and Fig. 9 respectively.

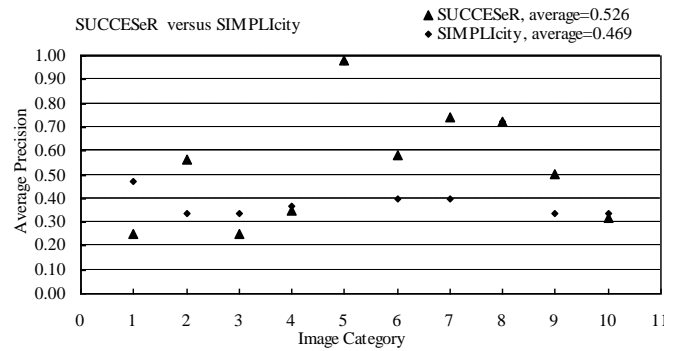


Figure 9. The average precision using vector V<sub>122</sub>

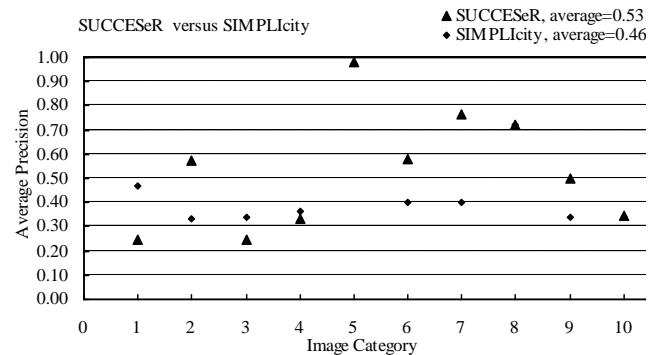


Figure 8. The average precision using vector V<sub>112</sub>

D. Performance of SUCCESeR

The retrieval results with different set of image categories in a given database are also comparable with existing SIMPLIcity [7] and WBIIS [16] systems.

Table II. COREL Categories of Images Tested

Class ID	Average Precision Using		
	Feature Vector V <sub>111</sub>	Feature Vector V <sub>112</sub>	Feature Vector V <sub>122</sub>
1	0.59	0.59	0.25
2	0.53	0.55	0.56
3	0.23	0.21	0.25
4	0.32	0.31	0.35
5	0.98	0.98	0.98
6	0.49	0.48	0.58
7	0.60	0.59	0.74
8	0.65	0.63	0.72
9	0.46	0.49	0.50
10	0.29	0.27	0.32
Average	0.514	0.510	0.526

Beside the performance comparison of our SUCCESeR system (using feature vectors V111, V112 and V122) for each of the ten image categories shown in Table I (a), the average precision based on the three sample images using SUCCESeR feature vectors V111, V112, and V122 for each of the ten

image categories shown in Table I (b) were also measured and it is given in Table II.

The performance in terms of the average precision using graph is also shown in Fig. 10, Fig 11 and Fig. 12 for each of the ten image categories shown in Table I (b) based on the three sample images using SUCCESeR feature vectors V111, V112, and V122.

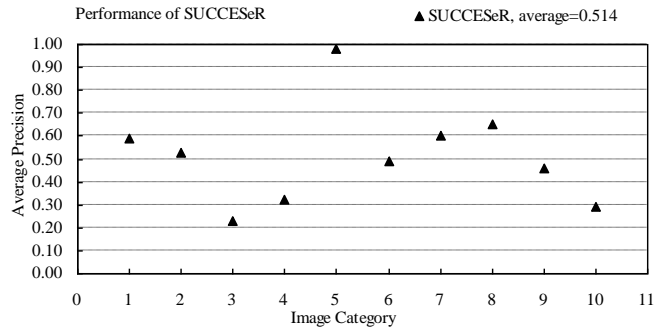


Figure 10. The average precision using vector V<sub>111</sub>

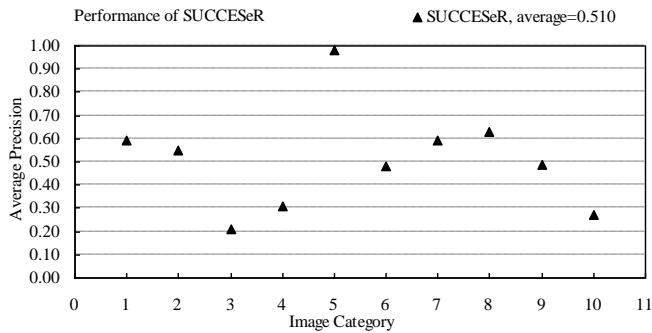


Figure 11. The average precision using vector V<sub>121</sub>

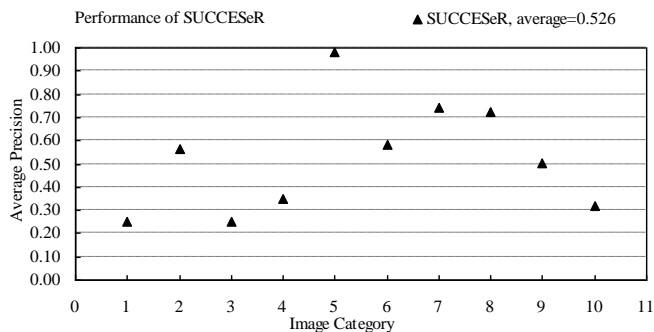


Figure 12. The average precision using vector V<sub>122</sub>

### E. Performance of SUCCESeR in terms of Speed

The algorithm has been implemented on a Pentium IV 2.4GHz PC using the Win-dows XP operating system. To compute the feature vectors for the 1000 color images of size 384 x 256 resized to 80 x 80 to increase speed in our general-purpose image database requires approximately 294.598 seconds for indexing images. The indexing using feature vector V111 takes less time as compared to feature vector V122.

The matching speed is very fast. When the query image is given from the database, it takes about 2.5:2.9 seconds of CPU time on average to extract and display all the 20 images in the indexed 1000-image database using the similarity measure given in (5).

### F. Retrieval Results

The query image is the image at the upper-left corner. The numbers below the images are the ID numbers of the images in the database. The other numbers next to image ID are the values of the similarity measure given in (5) between the query image and the matched image.

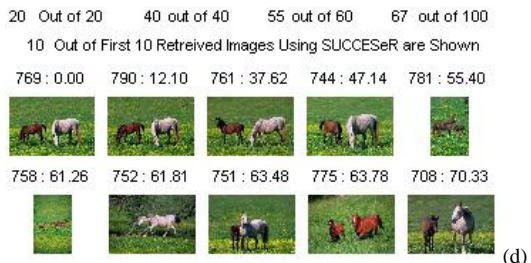
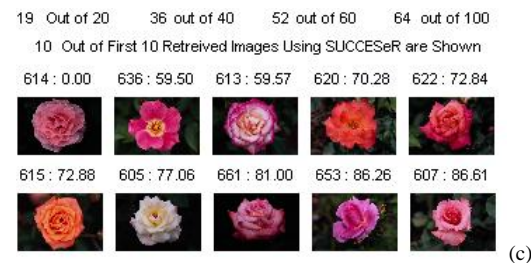
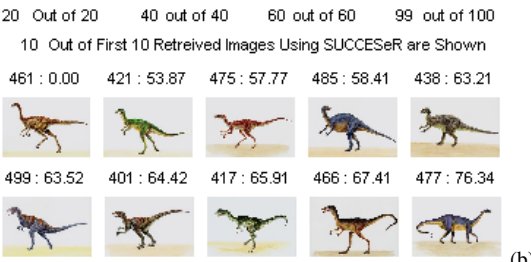
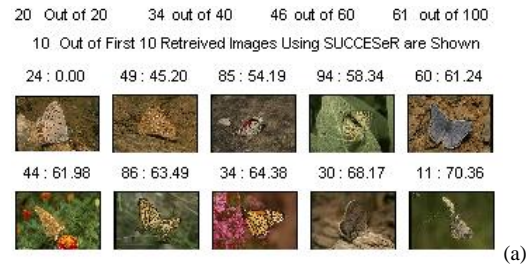


Figure 13. Retrieval Results using feature vector V<sub>111</sub> (a) Butterfly (b) Dinosaurs (c) Flowers and (d) Horses.

The precision within the first 20 retrieved images are calculated. However the precision within the first 40, 60 and 100 retrieved images are also displayed.

There is no big difference between the retrieval results using the feature vector V111 and feature vector V122. The results using feature vector V111 using our SUCCESeR

program interfaces are shown in Fig. 13. The results using feature vector V122 using our SUCCESeR program interfaces are shown in Fig. 14.

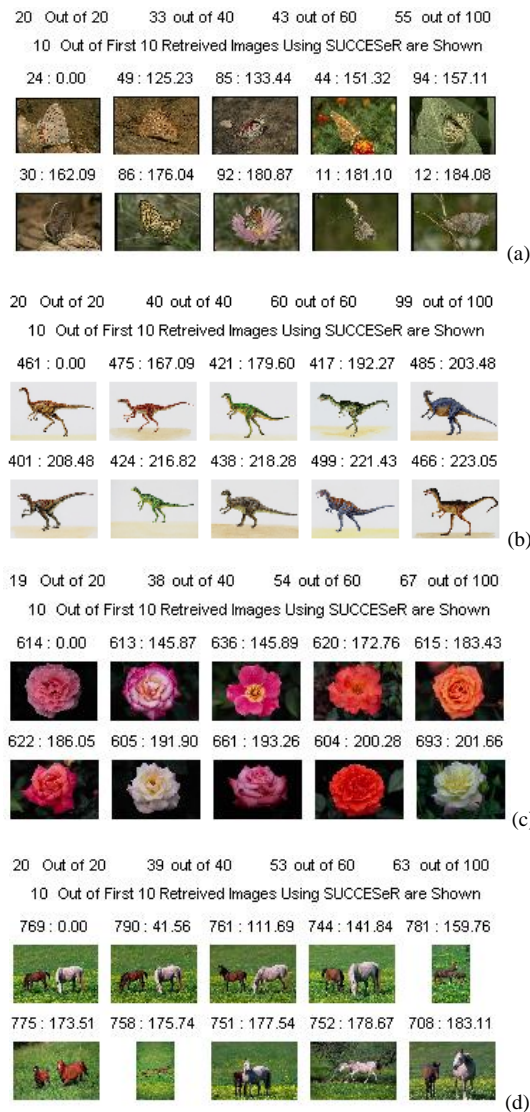


Figure 14. Retrieval Results using feature vector  $V_{122}$  (a) Butterfly (b) Dinosaurs (c) Flowers and (d) Horses.

## V. SUMMARY AND CONCLUSION

The SUCCESeR system gives average precision 1 within first 10 retrieved images with butterfly, beach, dinosaurs, flowers, and food image categories. The average precision within first 20 retrieved images with butterfly, beach, dinosaurs, flowers, and food image categories also tends to 1. The indexing using feature vector V111 takes less time as compared to feature vector V122.

## VI. FUTURE SCOPE

One possible solution to improve the results could be of considering the combination of other color models. Again this may take more time for indexing images. Also the results can be improved by including more elements in a new feature vectors than the elements in feature vector V122. Again this

may take more time for indexing images and retrieval. Here, the elements in feature vector V111 and feature vector V122 are 9 and 15 respectively.

## ACKNOWLEDGMENT

We propose our sincere thanks to one and all those who have helped and encouraged us to think in a possible direction to work and devise a useful method that helps our society to move one step ahead.

## REFERENCES

- [1] Swain, M. J. & Ballard, D. H. (1991) Color Indexing, International Journal of Computer Vision, 7, 1, 11-32.
- [2] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, and Akio Yamada, "Color and Texture Descriptors". In: IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 6, June 2001, pp. 70-75.
- [3] Remco C. Veltcamp, Mirela Tanse, "Content Based Image Retrieval Systems". A Survey, Technical Report UU-CS-2000-34, October 2000, pp. 1-62.
- [4] Yining Deng, B. S. Manjunath, Charles Kenney, Michael S. Moore, Student Member, Hyundoo Shin, "An Efficient Color Representation for Image Retrieval" IEEE Transactions on Image Processing, 2001.
- [5] J. R. Smith and S.-F. Chang. "Tools and techniques for color image retrieval", In Symposium on Electronic Imaging: Science and Technology - Storage & Retrieval for Image and Video Databases IV, volume 2670, San Jose, CA, February 1996. IS&T/SPIE.
- [6] James Hafner, Harpreet S. Sawhney, Will Equits, Myron Flickner and Wayne Niblack, "Efficient Color Histogram Indexing for Quadratic Form Distance Functions", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 17, No. 7, July 1995.
- [7] James Z. Wang, Jia Li, Gio Wiederhold, "SIMPLiCity: Semantics-sensitive Integrated Matching for Picture Libraries," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol 23, no.9, pp. 947-963, 2001.
- [8] S. Nandagopalan, Dr. B. S. Adiga, and N. Deepak, "A Universal Model for Content-Based Image Retrieval," PWASET Vol. 36, December 2008, ISSN 2070-3740.
- [9] P. S. Hiremath, Jagadeesh Pujari, "Content Based Image Retrieval using Color, Texture and Shape features," 15th International Conference on Advanced Computing and Communications, IEEE Computer Society 2007, pp. 780-784.
- [10] Se-Hwan Kim, Woontack Woo and Yo-Sung Ho, "Image Retrieval Using Multi-Scale Color Clustering," IEEE, 2001, pp. 666-669.
- [11] J. Geusebroek, D. Koelma, A.W.M. Smeulders and T. Gevers, "Image Retrieval and Segmentation based on Color Invariants," IEEE, pp. 1-2, 2000.
- [12] H. Nezamabadi-pour and S. Saryazdi, "Object-Based Image Indexing and Retrieval in DCT Domain using Clustering Techniques," PWASET Vol. 3, January 2005, ISSN 1307-6884, pp.98-101
- [13] C.W.Ngo, T.C.Pong and R.T.Chin, "Exploiting image indexing techniques in DCT domain", pattern Recognition, 2001, vol. 34, pp. 1841-1851.
- [14] G.Feng and J.Jiang, "JPEG compressed image retrieval via statistical features", Pattern Recognition, 2003, vol. 36, pp. 977-985.
- [15] A.Vailaya, A.K.Jain and H.J.Zhang, "On image classification: city vs. landscape", Pattern Recognition, 1998, vol. 31, pp. 1921-1935.
- [16] J. Z. Wang, G. Wiederhold, O. Firschein, and X.W. Sha, "Content-Based Image Indexing and Searching Using Daubechies' Wavelets," Int'l J. Digital Libraries, vol. 1, no. 4, pp. 311-328, 1998.
- [17] Text of ISO/IEC 15 938-3 Multimedia Content Description Interface—Part 3: Visual. Final Committee Draft, ISO/IEC/JTC1/SC29/WG11, Doc. N4062, Mar. 2001.
- [18] Roger Weber and Michael Mlivoncic, "Efficient Region-Based Image Retrieval", ACM CIKM '03 November 3-8, 2003, USA.
- [19] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom et al. "Query by Image and Video Content: The QBIC System," IEEE Computer, vol. 28, no. 9, 1995.

- [20] J.R. Smith and C.S. Li, "Image Classification and Querying Using Composite Region Templates," *Int'l J. Computer Vision and Image Understanding*, vol. 75, nos. 1-2, pp. 165-174, 1999.
- [21] A. Gupta and R. Jain, "Visual Information Retrieval," *Comm. ACM*, vol. 40, no. 5, pp. 70-79, May 1997.
- [22] S. Mukherjea, K. Hirata, and Y. Hara, "AMORE: A World Wide Web Image Retrieval Wngine," *Proc. World Wide Web*, vol. 2, no. 3, pp. 115-132, 1999.
- [23] S. Stevens, M. Christel, and H. Wactlar, "Informedia: Improving Access to Digital Video," *Interactions*, vol. 1, no. 4, pp. 67-71, 1994.
- [24] A. Pentland, R.W. Picard, and S. Sclaroff, "Photobook: Tools for Content-Based Manipulation of Image Databases," *Proc. SPIE*, vol. 2185, pp. 34-47, Feb. 1994.
- [25] R.W. Picard and T. Kabir, "Finding Similar Patterns in Large Image Databases," *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing*, vol. 5, pp. 161-164, 1993.
- [26] T.P. Minka and R.W. Picard, "Interactive Learning Using a Society of Models," *Pattern Recognition*, vol. 30, no. 3, p. 565, 1997.
- [27] J.Z. Wang, J. Li, R.M. Gray, and G. Wiederhold, "Unsupervised Multiresolution Segmentation for Images with Low Depth of Field," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 85-91, Jan. 2001.
- [28] S. Stevens, M. Christel, and H. Wactlar, "Informedia: Improving Access to Digital Video," *Interactions*, vol. 1, no. 4, pp. 67-71, 1994.
- [29] W.Y. Ma and B. Manjunath, "NaTra: A Toolbox for Navigating Large Image Databases," *Proc. IEEE Int'l Conf. Image Processing*, pp. 568-571, 1997.
- [30] R. Jain, S.N.J. Murthy, P.L.-J. Chen, and S. Chatterjee, "Similarity Measures for Image Databases," *Proc. SPIE*, vol. 2420, pp. 58-65, Feb. 1995.
- [31] E.G.M. Petrakis and A. Faloutsos, "Similarity Searching in Medical Image Databases," *IEEE Trans. Knowledge and Data Eng.*, vol. 9, no. 3, pp. 435-447, May/June 1997.
- [32] Y. Rubner, L.J. Guibas, and C. Tomasi, "The Earth Mover's Distance, Multi-Dimensional Scaling, and Color-Based Image Retrieval," *Proc. DARPA Image Understanding Workshop*, pp. 661- 668, May 1997.
- [33] B. M. Mehtre, M. S. Kankanhalli, A. D. Narasimhalu and G. C. Man, "Color matching for image retrieval," *Pattern Recognition Letters*, Vol. 16, pp. 325-331, March 1995.

#### AUTHORS PROFILE



**Satishkumar L. Varma** received B.Tech. and M.Tech. degrees from Dr. Babasaheb Ambedker Technological University, Lonere, Maharashtra, India in June 2000 and January 2004 respectively. He is pursuing Ph.D. degree from the SRTM University, Nanded, India. He has also worked with DBIT as an Assistant Professor and Head in the department of Information Technology. Currently he is Assistance Professor in the department of Computer Engineering in Pillai's Institute of Information Technology, New Mumbai, India. He has published one Journal paper and more than 12 papers in referred National as well as International Conferences including IEEE, Springer and IET. His research interest includes Multimedia Systems, Digital Image and Video Processing and Computer Vision.



**Sanjay N. Talbar** received his B.E and M.E degrees from SGGGS Institute of Technology, Nanded, India in 1985 and 1990 respectively. He obtained his PhD from SRTM University, Nanded, India in 2000. He received the "Young Scientist Award" by URSI, Italy in 2003. He had Collaborative research programme at Cardiff University Wales, UK. Presently he is working as Professor and Head, Department of Electronics & Telecommunication Engg., SGGGS Institute of Engineering & Technology Nanded, India. He has published 12 journal papers and more than 65 papers in referred National as well as International Conferences. His research interests includes Image processing, Multimedia Computing and Embedded System Design. He is a member of many prestigious committees in academic field of India.

# Automatic License Plate Localization Using Intrinsic Rules Saliency

Chirag N. Paunwala

Asst. Prof, Electronics and Communication Engg Dept  
Sarvajani College of Engg and Tech.  
Surat, India

Dr. Suprava Patnaik

Professor, Electronics Engg Dept.  
S.V. National Inst of Tech  
Surat, India

**Abstract**— This paper addresses an intrinsic rule-based license plate localization (LPL) algorithm. It first selects candidate regions, and then filters negative regions with statistical constraints. Key contribution is assigning image inferred weights to the rules leading to adaptability in selecting saliency feature, which then overrules other features and the collective measure, decides the estimation. Saliency of rules is inherent to the frame under consideration hence all inevitable negative effects present in the frame are nullified, incorporating great deal of flexibility and more generalization. Situations considered for simulation, to claim that the algorithm is better generalized are, variations in illumination, skewness, aspect ratio and hence the LP font size, vehicle size, pose, partial occlusion of vehicles and presence of multiple plates. Proposed method allows parallel computation of rules, hence suitable for real time application. The mixed data set has 697 images of almost all varieties. We achieve a Miss Rate (MR) = 4% and False Detection Rate (FDR) = 5.95% in average. Also we have implemented skew correction of the above detected LPs necessary for efficient character detection.

**Keywords**- License plate localization; Salient rules; Connected Region Analysis; statistical inconsistency; skew correction.

## I. INTRODUCTION

License Plate Localization (LPL) is being paid significance day by day, due to the exponential increase of traffic, requiring installation of smart traffic monitoring system. Applications like automatic toll collection, criminal chase, security control of restricted area and traffic law enforcement have been benefited from LPL system. Although experts have undertaken vast and in-depth research over the past many years for developing LPL technique, it is still driven forward due to demand for handling complex conditions arisen due to collective adverse happenings. To highlight, some of them are variations in LP size and associated font styles, time to time alteration in the standardization and vehicles with dissimilar standard LPs sharing the traffic, variations in road curvature, lanes and lighting, non standard vehicle with occlusion and difficulties in camera positioning.

This work aims at developing intelligent LPL algorithm for monitoring Indian traffic. By law all LPs in India are required to be of standard format consisting of 9 to 12 characters as shown in figure 1. The first two letters identify the state code, followed by two numbers to identify the district. This is often followed by a series code, e.g. 14E, is the fourteen series of private cars and 2M is the second series of motor bikes.

Recently many states have been adapting a dual letter series for example CA, CB, CC for cars and MA, MB, MC for bikes. Finally a 4 digit number is used to uniquely identify the vehicle. In some states like Delhi tertiary and Gujarat initial '0' of the district code is omitted.

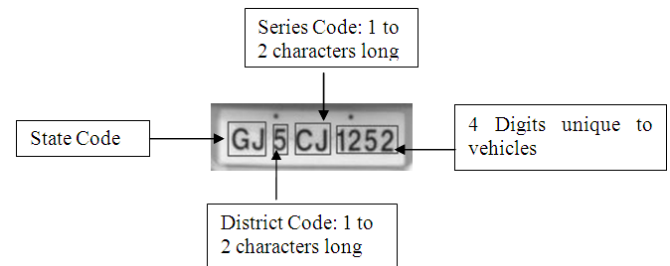


Figure 1. Typical LP pattern in India

## II. STATE OF THE ART

Prototype LPL systems came into functioning in early eighties. The era began with electronic circuitual approach and gradually bowed into computer based processing for real time application. This field has been researched rigorously for more than three decade and many commercial systems are in use in developing countries. However, robust solutions able to face real-world scenarios still need to be proposed. Most of the existing commercial solutions are only applicable to controlled conditions of illumination, viewpoint, or require high-resolution and specialized imaging hardware, possibly combined with infrared strobe lights.

Techniques based upon combinations of edge statistics and mathematical morphology [1-4] featured good results. Main disadvantage of this hybrid technique is that edge based methods alone can hardly be applied to complex images, since they are too sensitive to unwanted edges, which may also show a high edge magnitude or variance (e.g., the radiator region in the front view of the vehicle). When combined with morphological steps that eliminate unwanted edges in the processed images, the LP extraction rate becomes relatively high and fast. Though morphological deal along with edge analysis gives good results, it is limited to the fixed distance and angle between camera positions to vehicle as well as it is fail for plates of different size and more than one vehicle in single frame.

In [5], a block based recognition system was proposed to extract and recognize license plates of motorcycles and vehicles on highways only, in which as a first stage, a block-difference method was used to detect moving objects and then a screening method, based on the projection of edge magnitudes was used to find two peaks in the projection histograms to find license plates. But main shortcoming of this method is high FDR because of projection of edges.

Former class of algorithm uses knowledge guided train and test approach like support vector machine (SVM) [6] and Adaboost [7]. In SVM, the system uses a window of small size to scan frame and classify the center pixel either for LP or background region by analyzing properties like color and texture. In Adaboost [7] method, first training is planned for classifier using a set of LP and non-LP images and then test frame is fed in to the system, to detect the location of plate in the test image. Knowledge guided training approach is highly depended on training data set and it is not clear how well the detector will generalize in other situations. Haar like features are generally use for vertical and horizontal direction analysis in association with classifiers. Training based methods are also slower compared to edge-based methods.

In [8,9], wavelet feature analysis is applied to identify LP regions, which is helpful to significantly highlight the vertical edges of license plates because of its multi-scale property which enables detection of plates in different scales. Then, LP can easily be extracted by first-order local recursive Otsu segmentation [10] and orthogonal projection histogram analysis.

In Hough transform (HT) based method for license plate extraction, edges in the input image are detected first. Then, HT is applied to detect the LP regions. In [11], a combination of Hough transform and contour algorithm was applied on the edge image. Then the lines that cross the plate frame were determined and a rectangular-shaped object that matched the license plate was extracted. In [12] scan and check algorithm was used followed by radon transform for skew correction.

Representation of color in an invariant manner is of main objectives for color-based object recognition system and therefore based on human perception this feature is very powerful for object recognition. However, the sensitivity of this feature to the parameters such as color of car, illumination condition and the quality of imaging system has been restricted its usage as well as it is computationally complex [13].

Foundation of all the above literatures is vertical edge analysis and hence they are not robust to all natural variations in traffic imaging conditions. Standalone algorithm to handle high range of ambient light, skewness, font size and aspect ratio with no external association is indispensable. Additional features like low computation complexity leading to real time processing is appreciable.

Section-3 of proposed algorithm describes the pre-processing necessary for probable candidate region selection, followed by section-4, illustrating rules and adaptive saliency measure approach. Results obtained under various adverse conditions, to justify the robustness claimed earlier are shown

in section-5. Section 6 deals with skew correction part of detected LPs. Section-7 derive the conclusion.

### III. PREPROCESSING

LPL step is responsible for transformation of data between the real environment and information system. Aim is to extract semantics accurately, for which preprocessing becomes obligatory. We have considered a two-step preprocessing. In first stage for contrast enhancement, algorithm proposed in [14] is applied in which statistical analysis was done for reducing the processing time and if outcome is low, sigmoid transformation is applied for contrast enhancement and then second stage searches for probable candidate region based on vertical edge density and region connectivity. Figure 2 shows the flow chart of proposed algorithm.

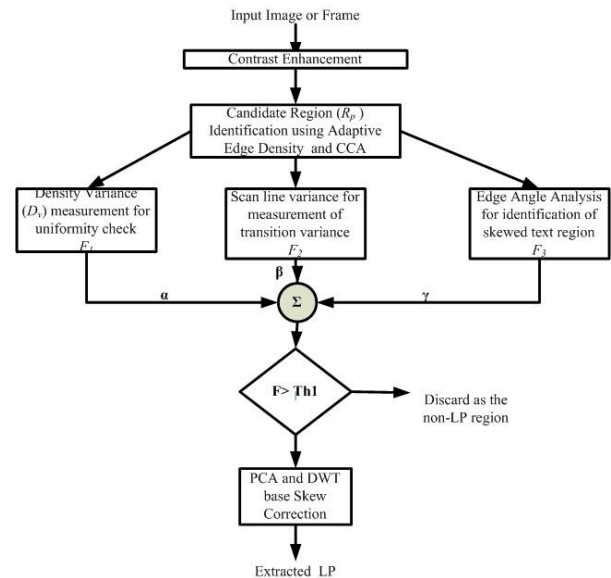


Figure 2. Flow Chart of Proposed Algorithm

#### A. Candidate Region Identification

Edges display irregularities in intensity due to presence of salient features. In this application the vertical gradient image is obtained using the x-direction Sobel operator, which is further binarized using Otsu's concept [10] of optimal threshold  $k^*$ , to maximize the between class variance  $\sigma_B^2(k^*)$

$$\text{Where } \sigma_B^2(k^*) = \max_{0 \leq k \leq L-1} \sigma_B^2(k)$$

The LP also contains the characters of almost equal shape and size and is placed at regularly spaced intervals. In this method a mask of pre-defined size ( $w \times h$ ) is made to budge over the entire image. The edge density is calculated using (1).

$$\text{Edge\_den} = \frac{1}{(w \times h)} \sum_{x_i}^{x_i+w} \sum_{y_j}^{y_j+h} \text{edge}(x_i, y_j)$$

$$\text{where } x_i = x_{i-1} + k_1; y_j = y_{j-1} + k_2; \quad (1)$$

$$\text{and } \text{edge}(x_i, y_j) \in \{0, 1\}$$



Where  $k_1$  and  $k_2$  decides the mask overlapping along horizontal and vertical direction respectively. If  $Edge\_den$  is higher than threshold, region covered by mask is considered as probable candidate of LP. The size of mask is decided according to the dynamic range between camera and traffic locality under analysis, which also decides the probable distance of vehicle from the camera and aspect ratio of LP and character size. The height of the mask is computed experimentally from maximum expected plate character height  $Max\_char\_ht$  and minimum expected plate character height  $Min\_char\_ht$  as shown in (2). Purpose of choosing two different parameters is to consider two row plates, which are very common in Indian scenario.

$$mask\_ht = h = \left\{ \frac{(Max\_char\_ht + Min\_char\_ht)}{2} * \eta \right\}$$

$$mask\_width = w = \frac{h}{AR\_lp} \quad (2)$$

Where,  $0.5 < \eta < 0.9$

As shown in figure 3, white rectangle is the LP region; gray rectangles are masks with different  $\eta$  values. Small values of  $\eta$  will upshot missing of LP, if vehicle is close to camera. Similarly for large value of  $\eta$  probable LP region will come with additional back ground, which may affect the detection accuracy.

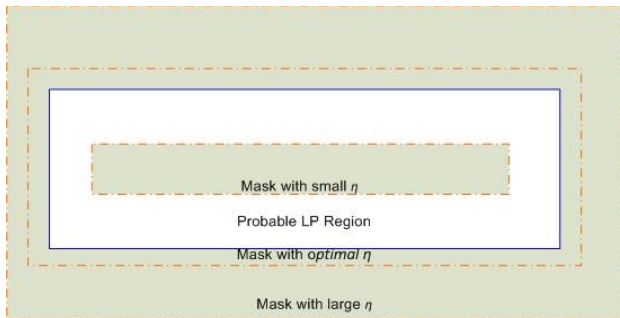


Figure 3: Importance of  $\eta$

Figure 4 shows the vertical-gradient image as a thermo graphic map, where “hot” spots denote areas with high vertical edge density which includes all LP regions.

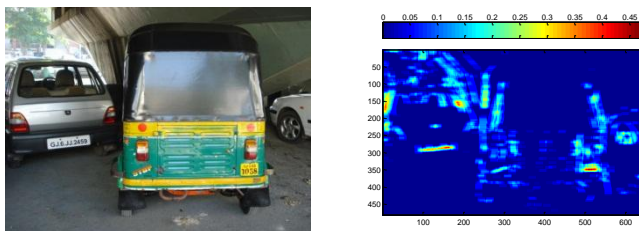


Figure 4. Original image and its thermo graphic image

### B. Connected Region Analysis (CRA)

Connected Region Analysis is performed on probable regions, with 8- connectivity test performed for edge pixels of the regions. If  $R_n$  and  $R_{n+N}$  (where  $1 < N < 8$ ) are two connected

regions, CRA will merge them to one region of larger size. Figure 5 shows the possible region connectivity and the outcome after CRA for two blocks.

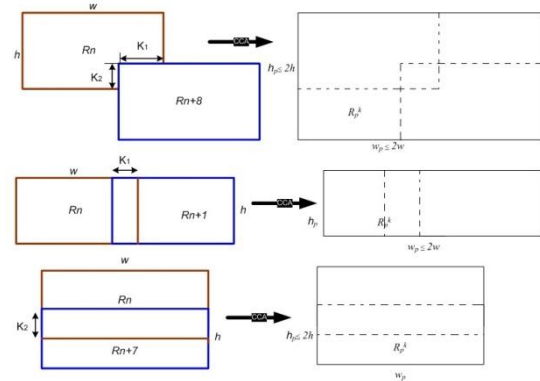


Figure 5. Formation of  $R_p^k$  after CCA

## IV. PROPOSED RULES AND SALIENCY EVALUATION

Next task is to filter out the non-LP regions from the connected probable regions and this is analyzed by setting required statistical properties. We have considered three rules, sufficient to handle almost all complex situations which can be processed in parallel for real time implementation.

### A. Density Variance Rule

License plate consists of not only the copious edge information, but the foreground characters are also usually distributed with relatively even interval. Therefore, for block containing text, the foreground pixels distributed uniformly in the blocks and this feature is useful to discriminate text blocks from graphic blocks. Figure 6 shows the vertical gradient distribution both in LP and non-LP regions.

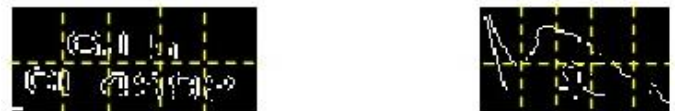


Figure 6. Signature of gradient variation in LP and Non-LP region

For finding out density variance, we segregate block  $R_p$  in to  $n$  equal sized sub blocks  $B_i$ , where  $1 < i < n$  and  $n=10$  for our case. Then density variance and mean value  $B_{mean}$  of block  $R_p$  is calculated using (3) and (4) respectively.

$$D_v = \frac{\sum_{i=1}^n |B_{mean} - B_i|}{B_{mean}} \quad (3)$$

and

$$B_{mean} = \frac{1}{n} \sum_{k=1}^n B_k \quad (4)$$

From (3),  $D_v$  remains low for uniform gradient distribution across the block  $R_p$ .

### B. Scan Line Variance Rule

LPs consists of English alphabets subscribed on contrast background, hence possess a definite edge distribution pattern. The transition between plate background and characters, such as the ones shown in figure 7, are used for inference derivation. To reduce complexity as well as to improve accuracy, top and bottom 20% of rows are discarded with an assumption that characters on bounding box are centrally located.



Figure 7. Concept of scan line calculation

Eq. (5) formulates the scan line variance, where  $T_i$  is the number of transitions in the  $i$ -th row.

$$scan\_vr = \frac{\sum_{i=1}^L (T_i - T_{avg})}{T_{avg}} \quad (5)$$

Where 
$$T_{avg} = \frac{1}{L} \sum_{i=1}^L T_i$$

Main limitation of this parameter is that it is fail to detect skew LP where numbers of transitions may be less due to oblique view leading to fusion of characters.

### C. Edge Orientation Rule

The distribution of edges is more directional in text blocks, i.e. edge direction histogram corresponding to text block will have maxima at  $0^\circ$  and  $90^\circ$ . For each edge pixel direction  $\psi$  is obtained using (6)

$$\psi = \arctan(g_v / g_h) \quad (6)$$

Here  $g_v$  and  $g_h$  are gradient magnitudes of edge pixels. Gradient angle  $\psi$  is quantized  $Q(\psi)$  into integer number of bins. Direction bin counters ( $dir_i$ ) are then incremented if  $Q(\psi)$  falls in the bin. For example eight possible directions,  $\theta_1$  to  $\theta_8$ , with bin span of  $22.5^\circ$ , are as shown in figure 8.

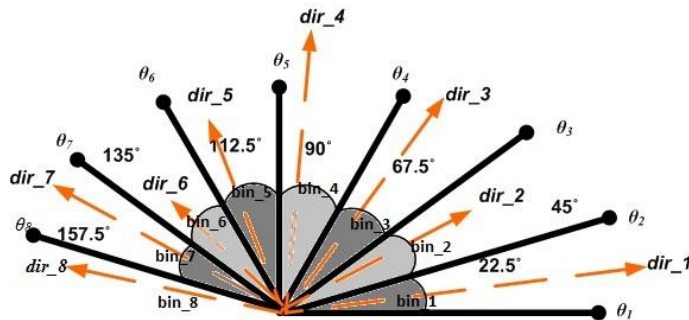


Figure 8. Edge Direction Counters

A selection means employed on each edge pixel to increment the corresponding counter is,

$$\begin{aligned} \theta_i &= Q(\psi / \theta_s) \text{ for } \psi \leq 180^\circ \\ &= Q\{(\psi - 180^\circ) / \theta_s\} \text{ for } \psi > 180^\circ \quad (7) \end{aligned}$$

where  $\theta_s = 180^\circ / 8$

For LP block, assignment of edge pixels for  $dir_1$  (horizontal edges) and  $dir_5$  (vertical edges) are maximum compared to other counters as shown in figure 9. Based on this we have used the third rule as given in (8), which supports extraction of tilted LPs.

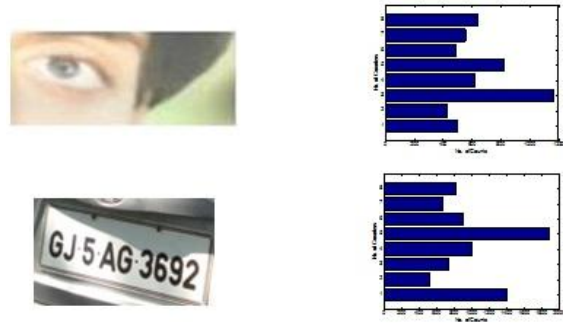


Figure 9. Histogram of counters for Non-text and Text blocks

$$edge\_ori\_vr = \frac{1}{Mean\_count} \left\{ \begin{aligned} &dir\_1 + dir\_{\frac{M}{2}+1} \\ &- \sum_{n=1, n \neq \frac{M}{2}+1}^M dir\_n \end{aligned} \right\} \quad (8)$$

where  $Mean\_count = \frac{1}{n} \sum_{n=1}^M dir\_n$

### D. Saliency Evaluation of Rules

After defining the mathematical formulation of three inconsistency measures, namely (1) density variance ( $F_1$ ), (2) scan line variance ( $F_2$ ) and (3) edge orientation variance ( $F_3$ ), next task is to examine rule saliency and assign related weightage  $\alpha$ ,  $\beta$  and  $\gamma$  respectively.

The dynamic range for selection of  $\beta$  is computed using (9) using the ratio of count( $T$ ) of scan lines with transitions above a predefined value to the total number of scan lines ( $N$ ), and aspect ratio of probable region under consideration.

$$\beta = \frac{1}{3} * \left\{ \frac{T}{N} + \frac{h_p}{w_p} \right\} \quad (9)$$

Indian LPs consist of 9 to 12 characters therefore block consisting of few text lines can pass through rule  $F_3$  but it will be blocked by rule  $F_2$  due to insufficient edge information. Therefore  $\gamma$  is determined taking edge orientation into account as shown in (10).

$$\text{If } \frac{dir\_1 + dir\_{\frac{M}{2}+1}}{Total\_count} > 0.5$$

$$\text{then } \gamma = \frac{1}{3} * \left\{ \frac{\text{dir}_{-1} + \text{dir}_{-(\frac{M}{2} + 1)}}{\text{Total\_count}} + \frac{h_p}{w_p} \right\} \quad (10)$$

else  $\gamma=0.2$

' $\gamma$ ' is minimum for erected view and maximum for oblique view. Finally factor  $\alpha$  for  $F_1$  is decided to satisfy the normalization criteria as

$$\alpha = 1 - (\beta + \gamma) \quad (11)$$

Amalgamated rule  $F$  then becomes,

$$F = \alpha * F_1 + \beta * F_2 + \gamma * F_3 \quad (12)$$

If block under consideration has  $F$  more than experimentally set threshold (in our case it is 5.9) then it is selected as LP region, else is discarded.

### V. EXPERIMENTATION RESULTS

Our database for experimentation consists of 237 low contrast images, consisting of 442 LPs that includes commercial or cargo transport vehicles, motor cycles, auto rickshaw and more than two vehicles in a frame, named as set-2. In order to compare the performance unchanged method is also applied on another set consisting of fine images with 707 LPs that is set-1. Two performance measure parameters are:

1. Miss Rate (MR), ratio of number of miss plates (NM) out of total number of detected plates (NDP).

$$MR(\%) = \frac{NM}{NDPs} * 100 \quad (13)$$

2. False Detection Rate (FDR) which checks accuracy of system by taking ratio of number of false detection (NFD) to NDP.

$$FDR(\%) = \frac{NFD}{NDPs} * 100 \quad (14)$$

Table 1 gives the performance measure of proposed algorithm on both sets and figure 10 shows the extraction of plate for some unfavorable cases. We have compared proposed algorithm with [15], as our inclination is more towards Indian conditions and achieved impressive improvement.

TABLE I. ANALYSIS OF RESULTS USING PROPOSED ALGORITHM

Param- eters	Analysis on set-1		Analysis on set-2	
	Proposed work	Algorithm of [15]	Proposed work	Algorithm of [15]
% MR	3.68	6.5	4.43	8.5
%FDR	5.1	8.2	6.8	10.2

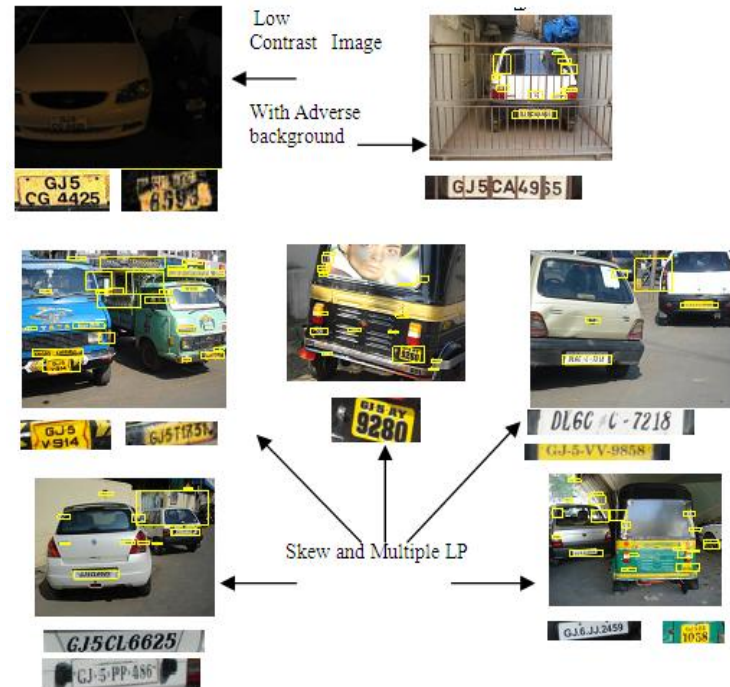


Figure 10. Results with some typical cases

### VI. SKEW CORRECTION TECHNIQUE

The skewed license plate affects badly on the accurate character segmentation and recognition. To solve the problem, an efficient approach for skew correction of license plate is proposed based on wavelet transform and principal component analysis. Figure 11 shows the representation of proposed algorithm for skew correction.

By decomposing an extracted LP with 2D-DWT with 'Haar' as a mother wavelet, four frequency sub bands  $LL_1$ ,  $LH_1$ ,  $HL_1$  and  $HH_1$  are obtained. Again by taking second level of decomposition we have achieved bands  $LL_2$ ,  $LH_2$ ,  $HL_2$  and  $HH_2$ . Main criterion over here is to find out exact discontinuity in the image, i.e. no edge will be missing. From experimentation, we have short listed 'Daubechies' (DB3 onwards) and 'Haar' wavelets out of other available wavelet families. The final scaling value in the 'Daubechies' transform is not the average of the data set, as it is in the case of the Haar transform, this results that the low pass filter part of the 'Daubechies' transform does not produce as smooth a result as the Haar transform. Finally, we have selected 'Haar' as mother wavelet because of the fact that processing has to be done on binary image and there are only two transitions in 'Haar' wavelet. 'Haar' wavelet will be well suited to binary images and it does not require any complex calculation. Also computational complexity is also very less in 'Haar' compare to 'Daubechies'.

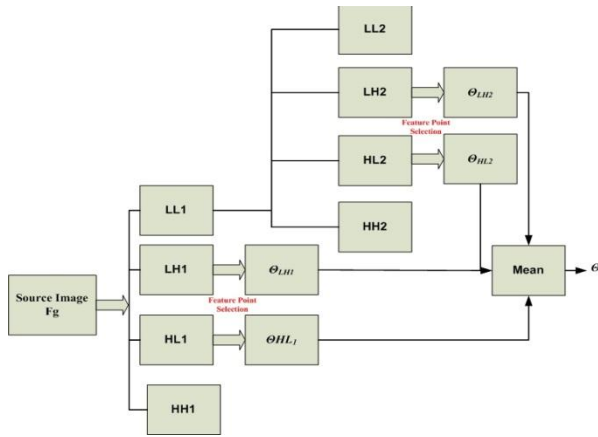


Figure 11. Representation of Skew Correction Algorithm.

Here we are using bands LH<sub>1</sub>, HL<sub>1</sub>, LH<sub>2</sub> and HL<sub>2</sub> for feature point selection, as it preserves horizontal and vertical details contained in the image, as well as we get good approximation of the results. Next step is to set a threshold value to binarize the levels obtained after wavelet analysis. White pixels in the image are the extracted feature points as shown in figure 12. Measurement of the two variables  $x_1$  and  $x_2$  are considered and measured data under consideration is within the coordinate system  $(x_1/ x_2)$ , and then the transform it into new coordinate system  $(z_1/ z_2)$  using principal component analysis (PCA). The new set of values is examined; the direction in which data are maximal is the direction of principal component  $z_1$  and the one with smaller variations is  $z_2$ . Therefore component with the smaller variations is neglected.



Figure 12. Feature Points of different Bands

For PCA, initially we have to compute the mathematical expectation for feature points in both x and y coordinates using (15) and using it covariance matrix  $C_x$  will be calculated using (16). Finally eigen values  $\lambda_j$  and feature vectors  $P_j$  are computed using (17) and by arranging eigen values in descending order gives the transformation kernel matrix H using (18).

$$M_{x1} \approx \frac{1}{N} \sum_{k=1}^N x_{1k}, M_{x2} \approx \frac{1}{N} \sum_{k=1}^N x_{2k} \quad (15)$$

$$C_x = \frac{1}{N} \sum_{k=1}^N (x_{1k} - M_{x1})(x_{2k} - M_{x2})^T \quad (16)$$

$$|\lambda_j I - C_x| = 0 \quad (17)$$

Where  $I$  is an identity matrix,  $\lambda_j = (\lambda_1, \lambda_2, \dots, \lambda_j)$  and  $\lambda_1 > \dots > \lambda_j > \dots > \lambda_k$ . The corresponding feature vector of  $\lambda_j$  is  $P_j$  and  $P_j = [P_{j1}, P_{j2}, \dots, P_{jk}]^T$ .

$$H = \begin{bmatrix} P_1^T \\ \dots \\ P_k^T \end{bmatrix} = \begin{bmatrix} P_{11} & \dots & P_{1k} \\ \dots & \dots & \dots \\ P_{k1} & \dots & P_{kk} \end{bmatrix} \quad (18)$$

The new vector set Y is obtain by performing PCA as shown in (19)

$$Y = H(X - Mx) \quad (19)$$

Since mathematical expectation of (19) is zero, its origin of coordinates has moved to the central position which results in to new covariance matrix  $C_{x\_new}$ , as mention in (20).

$$C_{x\_new} = \begin{bmatrix} \lambda_1 & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & \lambda_k \end{bmatrix} \quad (20)$$

The feature vector  $P_1$ , correspondence to most significant eigen value  $\lambda_1$  (which has maximum value) gives the direction of principal component  $z_1$  as shown in figure 13. PCA turns the original coordinates into the variance maximum direction and enables the variance of each pixel coordinate projection along projected direction to achieve the maximum.

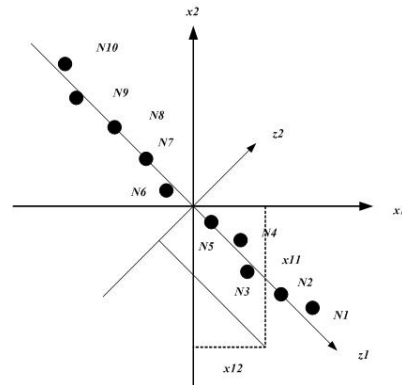


Figure 13. Plot of N measurements of principal components with two (horizontal and vertical) variables

After 1st level of decomposition (for LH<sub>1</sub>) by computing and arranging eigen values  $\lambda_1$  and  $\lambda_2$  in descending order and computing eigen vectors  $P_1$  and  $P_2$ , rotation angle  $\theta_{LH1}$  will be obtained using (21).

$$\theta_{LH1} = \cos^{-1}(P_1) \quad (21)$$

Similar procedure has to do for HL<sub>1</sub>, LH<sub>2</sub> and HL<sub>2</sub> and compute  $\theta_{HL1}$ ,  $\theta_{LH2}$  and  $\theta_{HL2}$  respectively. Finally rotation by using bilinear interpolation is taken place by taking mean of all four angles. Table 2 shows the computation of rotation angles for image under consideration and figure 14 gives the original skewed and de-skewed LP after skew correction. Figure 15(a) and (b) shows the results of positive and negative skew correction respectively

TABLE II. COMPUTATION OF ROTATION ANGLE

Comp.	LH <sub>1</sub>	HL <sub>1</sub>	LH <sub>2</sub>	HL <sub>2</sub>
$\lambda_1$	470.061	449.483	130.880	109.652
$\lambda_2$	1476.443	1411.645	391.877	362.326
$P_1$	[-0.9654 -0.2607]	[-0.9951 0.0986]	[-0.9687 - 0.2481]	[-0.9929 0.1192]
$P_2$	[-0.2607 0.9654]	[0.0986 0.9951]	[-0.2482 0.9687]	[0.1192 0.9929]
$\theta$	15.1092	5.6576	14.3692	6.8471



Figure 14 (a) Original Image (b) De skewed Image



Figure 15 (a) Positive Skew Correction (b) Negative Skew Correction

## VII. CONCLUSIONS

This work presents LPL method based on statistical rules and saliency of rules which gives comparable performance under complex adverse condition. The robustness of proposed method is demonstrated by applying it on a mixed data set with high diversity. Advantage of the proposed algorithm is that the detection box has high precision and it is independent of the detecting license plates with different styles, scales, poses, and partial occlusion. Additional merit is proposed method supports parallel processing and suitable for real time computation. For modifying proposed algorithm for specific country, changes have to be made in *Max\_char\_ht*, *Min\_char\_ht* and threshold selection for deciding *F*.

In this paper we have also suggested post processing of detected LP regions to nullify the skew error. Skew corrected images then can readily be passed through character detectors for LP number extraction. Image features are view point dependent. Hence normalization of extracted LPs to prescribed size along with orientation would improve the performance. We leave this issue to be considered in future study. Our parallel feature filtering approach reduces the computation time reasonably, however in order to make the system applicable to real time in less restrictive environment feature selection need to be image intrinsic and independent of view point.

## REFERENCES

[1] F. Martin, M. Garcia and J. L. Alba. "New methods for Automatic Reading of VLP's (Vehicle License Plates)," in Proc. IASTED Int. Conf. SPPRA, pp: 126-131, 2002.

[2] C. Wu, L. C. On, C. H. Weng, T. S. Kuan, and K. Ng. "A Macao License Plate Recognition system," in Proc. 4th Int. Conf. Machine Learning Cybernetics, pp: 4506-4510, 2005.

[3] Feng Yang and Fan Yang. "Detecting License Plate Based on Top-hat Transform and Wavelet Transform", ICALIP, pp:998-2003, 2008

[4] Feng Yang and Zheng Ma. "Vehicle License Plate Location Based on Histogramming and Mathematical Morphology", Automatic Identification Advanced Technologies, 2005. pp:89 - 94, 2005

[5] H.-J. Lee, S.-Y. Chen, and S.-Z. Wang. "Extraction and Recognition of License Plates of Motorcycles and Vehicles on Highways," in Proc. ICPR, pp. 356-359, 2004.

[6] K. I. Kim, K. Jung and J. H. Kim, "Color Texture-Based Object Detection: An Application to License Plate Localization", vol. 2388, Springer-Verlag, pp. 293-309.

[7] Xiangdong Zhang, Peiyi Shen, Jianhua Bai, Jing Lei, Yang Hu, Yuli Xiao, Bo Li and Dongpo Qi, "License Plate Detection Using location using AdaBoost", IEEE international conference on Information and Automation, pp:1705 - 1710, 2010.

[8] C.T.Hsieh, Y.-S.Juan and K.-M.Hung, "Multiple license plate detection for complex background", International Conference on Advanced Information Networking and Applications, pp:389-392, 2005.

[9] Chirag Paunwala and Suprava Patnaik, "An Improved License Plate Extraction Technique Based on top-hat transformation and Prolonged Haar Wavelet Analysis", ACM International conference and workshop on Emerging Trends in Technology, pp: 618-622, 2010

[10] N.Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE Transactions on System Man and Cybernetics, vol.9(1), pp:62-66, 1979.

[11] T. D. Duan, T. L. H. Du, T. V. Phuoc and N. V. Hoang, "Building an automatic vehicle license-plate recognition system" , in Proceedings International Conference on Computer Science, pp. 59-63, 2005.

[12] J. Kong, X. Liu, Y. Lu, and X. Zhou. "A novel license plate localization method based on textural feature analysis," in Proc. IEEE Int. Symp. Signal Process. Inf. Technol., Athens, Greece, pp. 275-279, 2005.

[13] X.Shi, W.Zhao and Y.Shen, "Automatic license plate recognition system based on color image processing", LNCS, vol.3483, pp:1159-1168, 2005.

[14] Chirag Paunwala, Suprava Patnaik, A Novel Multiple License Plate Extraction Technique for Complex Background in Indian Traffic Conditions, Int. Journal of Image Processing, Computer Science Journals, vol.4(2), pp:106-118, 2010.

[15] Prathamesh Kulkarni, Ashish Khatri, Prateek Banga, Kushal Shah, "A Feature Based Approach for Localization of Indian Number Plates", IEEE International Conference on Information Technology, pp: 157-162, 2009

## AUTHORS PROFILE



Prof. Chirag Paunwala received his M.E. in Digital Techniques and Instrumentation from SGSITS, Indore, India in 2006. Presently, he is assistant professor at Electronics and Communication Engineering Department, SCET, Surat, India. His research interest includes Image Processing, Pattern Recognition and Digital Watermarking.



Dr Suprava Patnaik received M.Tech. in Electronic & Communication Engineering from NIT, Rourkela, in 1992, and Ph.D. in Electrical, Electronics & Communication Engineering Deptment from the IIT, Kharagpur in 2004. She is Associate Professor in the Electronics Engineering Department, SVNIT, Surat (Gujarat) India. Her research interests include Signal and Image processing; Image Segmentation, Pattern Recognition, Digital Watermarking, Image Registration & Video Compression.

# Performance Comparison of SVM and $K$ -NN for Oriya Character Recognition

Sanghamitra Mohanty, Himadri Nandini Das Bebartta  
Department of Computer Science and Application  
Utkal University, Vani Vihar  
Bhubaneswar, India

**Abstract**—Image classification is one of the most important branch of Artificial intelligence; its application seems to be in a promising direction in the development of character recognition in Optical Character Recognition (OCR). Character recognition (CR) has been extensively studied in the last half century and progressed to the level, sufficient to produce technology driven applications. Now the rapidly growing computational power enables the implementation of the present CR methodologies and also creates an increasing demand on many emerging application domains, which require more advanced methodologies. Researchers for the recognition of Indic Languages and scripts are comparatively less with other languages. There are lots of different machine learning algorithms used for image classification nowadays. In this paper, we discuss the characteristics of some classification methods such as Support Vector Machines (SVM) and  $K$ -Nearest Neighborhood ( $K$ -NN) that have been applied to Oriya characters. We will discuss the performance of each algorithm for character classification based on drawing their learning curve, selecting parameters and comparing their correct rate on different categories of Oriya characters. It has been observed that Support Vector Machines outperforms among both the classifiers.

**Keywords**-Recognition; Features; Nearest Neighbors; Support Vectors.

## I. INTRODUCTION

During the past thirty years, substantial research efforts have been devoted to character recognition that is used to translate human readable characters to machine-readable codes. Immense effort has been made on character recognition, as it provides a solution for processing large volumes of data automatically in a large variety of scientific and business applications. OCR deals with the recognition of optically processed characters rather than magnetically processed ones. OCR is a process of automatic recognition of characters by computers in optically scanned and digitized pages of text [3]. OCR is one of the most fascinating and challenging areas of pattern recognition with various practical applications. It can contribute immensely to the advancement of an automation process and can improve the interface between man and machine in many applications [1] [2]. The term image classification refers to the labeling of images into one of a number of predefined categories. Although it seemed not a very difficult task for humans, it has proved to be a difficult problem for machines. Therefore, image classification challenge is very important for OCR. One of the important tasks in machine learning is the electronic reading of

documents. All official documents, magazines and reports can be converted to electronic form using a high performance OCR. From past many years, many academic laboratories and companies are involved in research on printed recognition. The increase in accuracy of printed processing results from a combination of several elements i.e. the use of complex systems integrating several kinds of information., the choice of relevant application domains, and new technologies such as high quality high speed scanners and inexpensive powerful CPU's. In Character recognition system we required two things i.e. processing on data set and decision making algorithms. We can categorize preprocessing into three categories: the use of global transforms local comparisons and geometrical or topological characteristics. There are various kinds of decision methods have been used such as: various statistical methods, neural networks, structural matching and stochastic processing. Many recent methods developed by combining several techniques existing together in order to provide a better reliability to compensate the great variability of document processing.[4] To address this problem, designing and implementation of automatic image classification algorithms has been an important research field for decades. The methods popularly used in the early stage of OCR research and development are template matching and structural analysis [4]. The templates or prototypes in these early methods were simple design methods are insufficient to accommodate the shape variability of samples, and so, are not able to yield high recognition accuracies. For large sample data, the character recognition community has turned attention to classification methods  $K$ -NN and SVMs, are also actively studied and applied in pattern recognition.

In this paper, we discuss the strengths and weaknesses of classification methods that have been widely used, identify the needs of improved performance in character recognition, and suggest some research directions of classification that can help meet with these needs. We will focus on the classification of isolated (segmented) characters, though classification method.

## II. CHALLENGES IN ORIYA CHARACTER SET

The Oriya script has descended from the Brahmi script sometime around the 11th century AD. Oriya is the official national language of India, most frequently used by common people in Orissa. Oriya alphabets consist of 268 symbols (13 vowels, 36 consonants, 10 digits and 210 conjuncts).



Longest-run Features: For computing longest-run features from a character image, the minimum square enclosing the image is divided into 25 rectangular regions. In each region, 4 longest-run features are computed row wise, column wise and along of its major diagonals. The row wise longest-run feature is computed by considering the sum of the lengths of the longest run bars that fit consecutive black pixels along each of all the rows of a rectangular region, as illustrated. The three other longest-run features are computed in the same way but along all column wise and two major diagonal wise directions within the rectangular separately. Thus in all, 25x4=100 longest-run features are computed from each character image.

### B. Support Vector Machines

The objective of any machine capable of learning is to achieve good generalization performance, given a finite amount of training data, by striking a balance between the goodness of fit attained on a given training dataset and the ability of the machine to achieve error-free recognition on other datasets. With this concept as the basis, support vector machines have proved to achieve good generalization performance with no prior knowledge of the data. The principle of an SVM is to map the input data onto a higher dimensional feature space nonlinearly related to the input space and determine a separating hyper plane with maximum margin between the two classes in the feature space[15].

An SVM in its elementary form can be used for binary classification. It may, however, be extended to multi class problems using the one-against-the-rest approach or by using the one-against-one approach. We begin our experiment with SVM's that use the Linear Kernel because they are simple and can be computed quickly. For a two class support vector machine let m-dimensional inputs  $x_i$  ( $i = 1 \dots M$ ) belongs to Class 1 or 2 and the associated labels be  $y_i = 1$  for Class 1 and  $-1$  for Class 2. Let the decision function be

$$D(x) = W^t x + b \quad (1)$$

where  $w$  is an m-dimensional vector,  $b$  is a scalar, and

$$y_i D(x_i) \geq 1 - \xi_i \text{ for } i=1, \dots, M. \quad (2)$$

Here  $\xi_i$  are nonnegative slack variables. The distance between the separating hyper plane  $D(x) = 0$  and the training datum, with  $\xi_i = 0$ , nearest to the hyper plane is called margin. The hyper plane  $D(x) = 0$  with the maximum margin is called optimal separating hyper plane. To determine the optimal separating hyper plane, we minimize

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^M \xi_i \quad (3)$$

subject to the constraints:

$$y_i (w^t x_i + b) \geq 1 - \xi_i \text{ for } i = 1, \dots, M \quad (4)$$

where,  $C$  is the margin parameter that determines the tradeoff between the maximization of the margin and minimization of the classification error. The data that satisfy the equality in (4) are called support vectors.

A priori, one does not know which value of soft margin will yield the classifier with the best generalization ability. We optimize this choice for best performance on the selection portion of our data.

### V. PERFORMANCE COMPARISON AND EVALUATION

The experiments of character recognition reported in the literature vary in many factors such as the sample data, pre-processing technique, feature representation, classifier structure and learning algorithm. A better scheme to compare classifiers is to train them on a common feature representation.

#### A. Kernel method Vs statistical method

For Kernel methods like SVMs, the parameters of one class are trained on the samples of all classes. For statistical classifiers, the parameters of one class are estimated from the samples of its own class only. We compare the characteristics of two kinds of classifiers in the following respects.

- 1) *Complexity of training:* The parameters of K-NN classifiers are generally adjusted by distance measure. By feeding the training samples a fixed number of sweeps, the training time is linear with the number of samples. SVMs are trained by quadratic programming (QP), and the training time is generally proportional to the square of number of samples.
- 2) *Flexibility of training:* The parameters of K-NN classifiers can be adjusted in Feature weighting improve classification accuracy for global performance and also easy to add a new class to an existing classifier. On the other hand, SVMs can only be trained at the level of holistic patterns. This classifier is proportional to square of the number of classes, and to guarantee the stability of parameters, adding new classes or new samples need re-training with all samples.
- 3) *Classification accuracy:* SVMs have been demonstrated superior classification accuracies to K-NN classifiers in many experiments. When training with enough samples, SVM classifiers give higher accuracies than statistical classifiers.
- 4) *Complexity of training storage and execution complexity:* At same level of classification accuracy, SVM learning by Quadratic Programming often results in a large number of SVs, which should be stored and computed in classification. K-NN classifiers have much less parameters, and are easy to control. In a word, K-NN classifiers consume less storage and computation than SVMs.



B. Evaluation

We tested the performance on Oriya characters. As till date no standard dataset is available for Oriya Characters we have collected and created the dataset within our organization. For this a corpus for Oriya OCR consisting of data base for machine printed Oriya characters has been developed. Mainly samples have been gathered from laser print documents, books and newspaper containing variable font style and sizes. A scanning resolution of 300 dpi is employed for digitization of all documents. The training and testing set comprises of 10, 000 samples each. Fig. 2. Shows some sample characters of various fonts of Oriya script used in the experiment.

We have performed experiments with different types of images such as normal, bold, thin, small, big, etc. having varied sizes of Oriya characters. We have considered gray scale images for collection of the samples. This database can be utilized for the purpose of document analysis, recognition, and examination. The training set consists of binary images of 297 Oriya letters. We have kept the same data file for testing and training for all types of different classifiers to analyze the result.

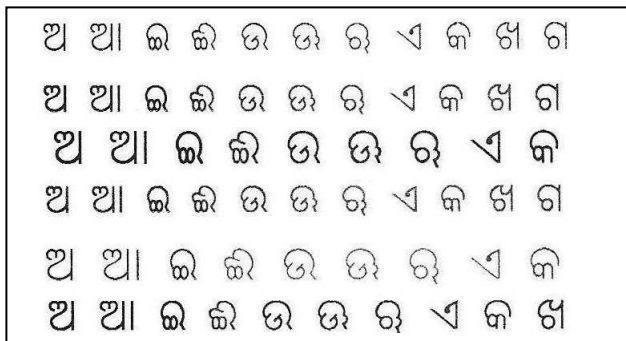


Figure 2: Samples of machine printed Oriya characters used for training

Below in Table 1 we can see the accuracy rate that we have obtained after testing the data set.

TABLE I. CLASSIFIERS ACCURACIES ON ORIYA CHARACTERS

Classifier	Training set	Test set	Accuracy Rate (in %)
SVM	10,000	10,000	98.9
KNN	10,000	10,000	96.47

Regarding the effect on accuracy by considering the different classifiers with different types of the images used for characters, for Oriya-Bold and big characters the accuracy rate is high in case of support vector machines and it has nearly 98.9 percentage of accuracy. The accuracy rate decreases for the thin and small size characters. Table 2 shows the effect on accuracy for Oriya by considering different character sizes with different types of the images using Support Vector Machines.

TABLE II: EFFECT ON ACCURACY BY CONSIDERING DIFFERENT CHARACTER SIZES WITH DIFFERENT TYPES OF THE IMAGES USED FOR ORIYA CHARACTERS.

Image type	Size of the samples	Accuracy percentage
ଅ ଆ ଇ ଈ	Bold and small	92.78%
ଅ ଆ ଇ ଈ	Bold and big	98.9%
ଈ ଉ ଊ ଋ	Normal and small	96.98%
ଅ ଆ ଇ	Normal and Bold	97.12%

VI. CONCLUSION

The results obtained for recognition of Oriya characters are quite encouraging and show that reliable classification is possible using SVMs. Nearest neighbor classifiers consume less storage and computation than that of SVMs. Biggest limitation of the support vector approach lies in the choice of kernel. Second limitation is in speed and size, both in training and testing. We applied SVMs and K-NNs classifiers on same feature data. Further we will find out the accuracy rate by taking a different set of samples for the test set and this work in future can be extended with degraded, noisy machine printed and italic text.

ACKNOWLEDGMENT

We are thankful to DIT, MCIT and my colleague Mr. Tarun Kumar Behera for their help and support.

REFERENCES

- [1] J. Mantas, "An overview of character recognition methodologies", Pattern Recognition, vol. 19, pp. 425-430, 1986.
- [2] V. K. Govindan, and A. P. Shivaprasad, "Character recognition –a review", Pattern Recognition, vol. 23, pp. 671-683, 1990.
- [3] Pal, U., and B. B. Chaudhuri, "Indian script character recognition: a survey", Pattern Recognition, vol. 37, pp. 1887-1899, 2004.
- [4] S. Mori, C.Y. Suen, and K.Yamamoto, "Historical review of OCR research and development", Proceedings of the IEEE, vol. 80, pp. 1029-1058, July 1992.
- [5] Muzaffar Khan, Tirupati Goskula, Mohmmmed Nasiruddin , and Ruhina Quazi, "Comparison between k-nn and svm method for speech emotion recognition", International Journal on Computer Science and Engineering, vol. 3, pp. 607- 611, February 2011.
- [6] Amendolia et. al., "A comparative study of k-nearest neighbour, support vector machine and multi-layer perceptron for thalassemia screening" Chemometrics and Intelligent Laboratory Systems, vol. 69, pp. 13-20, August 2009.
- [7] N.Suguna1, and Dr. K. Thanukodia, "An improved k-nearest neighbor classification using genetic algorithm", IJCSI International Journal of Computer Science Issues, vol. 7, pp. 18-21, July 2010
- [8] K. Fukunaga, Introduction to Statistical Pattern Recognition, 2nd ed, Academic Press, 1990.
- [9] R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification, 2nd ed, Wiley Interscience, 2001.
- [10] V. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New Work, 1995.

- [11] C.J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition, Knowledge Discovery and Data-Mining, 1998, pp. 1-43.
- [12] U. Kressel, Pairwise Classification and Support Vector Machines, 1999, pp. 255-268.
- [13] A. Bellili, M. Gilloux, and P. Gallinari, "An MLP-SVM combination architecture for online handwritten digit recognition: reduction of recognition errors by support vector machines rejection mechanisms", International Journal of Document Analysis and Recognition, vol. 5, pp. 244-252, 2003.
- [14] J.X. Dong, A. Krzyzak, C.Y. Suen, "High accuracy handwritten chinese character recognition using support vector machine", Proceedings of the International Workshop on Artificial Neural Networks for Pattern Recognition, 2003.
- [15] C. J. C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998, pp. 121-167.



Himadri nandini Das Bebartta is continuing her PhD under the guidance of Prof. (Dr.) Sanghamitra Mohanty in the Department of Computer Science and Application, Utkal University, Bhubaneswar, Odisha, India. She has been a research scholar since April 2007. And has worked as a Junior research fellow in the project development of robust document analysis and Recognition system for printed Indian scripts, DIT, MCIT, India.

#### AUTHORS PROFILE



Prof. (Dr.)Sanghamitra Mohanty is the Professor and Head of the Department of Computer Science and Application, Utkal University, Bhubaneswar,. Odisha, India. Her research interests are Speech processing, Image processing and Natural Language Processing. She has published 132 papers in International Journals and Conferences. She has guided many Ph. D. students and has 10 IPRs to her credit.