EDGE BASED IMAGE SEGMENTATION

MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF

Master of Technology

In

Information Systems

Submitted By:

MAYANK SINGH

(2K12/ISY/18)

Under the Guidance of

Prof. O. P. Verma

(HOD, Department of IT)

DEPARTMENT OF INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(2012-2014)

CERTIFICATE

This is to certify that **Mayank Singh (2K12/ISY/18)** has carried out the major project titled **"Edge Based Image Segmentation"** in partial fulfillment of the requirements for the award of Master of Technology degree in Information Systems by **Delhi Technological University**.

The major project is a bonafide piece of work carried out and completed under my supervision and guidance during the academic session **2012-2014**. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Prof. O. P. Verma
Head of Department
Department of Information Technology
Delhi Technological University
Delhi-110042

ACKNOWLEDGEMENT

I take the opportunity to express my sincere gratitude to my project mentor Dr. O.P. Verma, Head of Department, Department of Information Technology, Delhi Technological University, Delhi for providing valuable guidance and constant encouragement throughout the project .It is my pleasure to record my sincere thanks to him for his constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

Mayank Singh

Roll No. 2K12/ISY/18

M.Tech (Information Systems)

E-mail: mayank8590@gmail.com

ABSTRACT

Image segmentation has emerged as a leading field of research due to its wide span of usage and applicability. The segmentation process involves partitioning of an image into various groups of disconnected regions with attributes such as intensity, colour, tone or texture which are both uniform and homogeneous in nature. Various techniques have been developed over time for improving segmentation of an image and the participation of a pixel within a region or cluster as a crisp or a fuzzy entity has been debated widely by various researches. K-Means, Fuzzy C-Means (FCM), Fuzzy Co-Clustering for Images (FCCI) have been widely regarded as some of the most efficient algorithms for the image segmentation process by various researchers and scholars. All these are iterative process in which results are improved over the course of various iterations based on an over heading optimization function. FCM has been widely acknowledged as a very efficient clustering method but it loses its efficiency over many situations like involvement of noise or distorted images. Many variations of FCM have been proposed to overcome these drawbacks. Edge based weight distribution is one of the most efficient variation of FCM given which ensures overcoming of FCM inefficiency with noisy images. FCCI is an improvement over FCM in terms of its optimization to ensure better and well defined segmentation of an image. In the proposed technique, a combination of FCCI and edge based weight distribution method has been explained which ensures efficient results as it incorporates the advantages of both the techniques. The proposed technique has been applied over 100 natural images taken from Berkeley's image segmentation database and the results have been compared with the above mentioned algorithms on the basis of mean square error as a quantitative measure.

Table of Contents

Figures and Tables

Chapter 1

INTRODUCTION

1.1 Digital Image Processing

Digital image processing is a branch of the electronic field wherein the image is transformed into an array of small numbers, called *pixels*, denoting a physical entity like scene radiance, kept in a digital memory, and processed by computer or other digital hardware. Digital image processing, works either as an enhancement tool for the user or to conduct autonomous analysis while offering various advantages in terms of cost, speed, and flexibility.

An image is depicted by a 2 dimensional functions of the form $f(x,y)$[1]. The value or amplitude of f at spatial coordinates (x,y) is a positive scalar quantity whose physical meaning is determined by the source of the image. In a digital image, (x,y), and the magnitude of f are all finite and discrete quantities.

It is difficult job to differentiate among the various areas of image processing and any other related domain such as computer vision. But the two differ in the type of output we obtain from them. **Computer vision** is a technical aspect of machines that can visualize objects in a digital form. Computer vision, as a scientific discipline is related with the technique for designing artificial intelligent systems that can retrieve information from images. The data from the images can be of various types, such as a video sequence, views from multiple cameras, or multidimensional data from a medical scanner. Computer vision takes a digital image as an input and provides some interesting information or features about it as output. Image processing is usually performed as a pre-processing step for computer vision. Image processing is a domain in which both input and output are images. The fundamental steps of digital image processing are shown in figure 1.1.

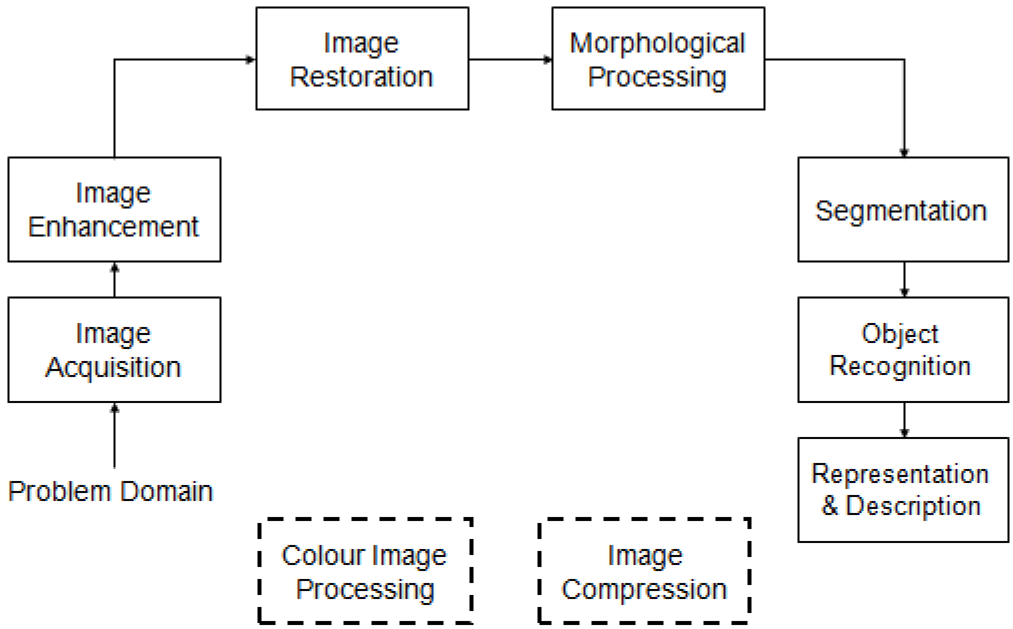


Figure1.1: Key Stages of Digital Image Processing

The design of a computer vision system is heavily dependent on its applicability. Some systems work independently to solve a specific measurement or detection problem and are termed as stand-alone application systems while others makes use of multiple sub-system, for example, some also use various sub-systems for controlling the mechanical actuators, planning, information databases, man-machine interfaces, etc. The explicit implementation of a computer vision system largely depends on whether its functionality is predefined or some part of it can be learned or changed during operation. However, there are some common functions which are found in almost every computer vision systems.

1. Image acquisition: A digital image is created by numerous image sensors which, besides various types of light-sensitive cameras, include range sensors, tomography devices, radar and ultra-sonic cameras. The output image can be an ordinary 2D image, a 3D volume, or an image sequence depending on the kind of sensor being used. The value of the pixel usually correspond to the intensity of light in various spectral bands depending on the type of image, but can also be related to various physical measures, such as depth, absorption or reflectance of sonic or electromagnetic waves, or nuclear magnetic resonance.

2. Pre-processing: It is essential to process the image before being fed into a computer vision system for extracting some necessary information about the image and to assure that it satisfies definite hypothesis implied by the technique. Examples are:

(a) Re-sampling is conducted to guarantee that the image coordinate system is accurate.

(b) Noise reduction is conducted to guarantee that sensor noise does not bring in false information.

(c) Contrast enhancement is conducted to guarantee that relevant information can be identified.

(d) Scale space representation is conducted to enhance image structures at locally appropriate scales.

3. Feature extraction: Image features can be obtained from the image at different stages of complexity. Some examples of extracted features are

(a) Lines, edges and ridges.

(b) Localized interest points such as corners, blobs or points.

More complex features that can be extracted are related to pattern, shape or motion and gesture.

4. Image Segmentation: At some point of time in the processing, a decision is made about which image points or regions of the image are important for further processing. Examples are

(a) Selection of a specific set of interest points

(b) Segmentation of one or more image parts which contain a specific area of interest.

5. High-level processing: Here, the input is generally a small set of numbers, for example a set of points or an image region which is assumed to contain a specific region. The remaining processing deals with, for example:

(a) Verification of data whether it satisfies model-based and application specific assumptions.

(b) Estimation of application specific parameters, such as object pose or object size.

(c) Classification of detected object into different categories

Hence it can be concluded that image segmentation forms a vital part of computer vision systems and is more a domain of computer vision than image processing.

1.2 Image Segmentation

Segmentation of an image entails the division or separation of the image into regions of similar attribute. The basic attribute for segmentation is image amplitude- luminance for a monochrome image and color components for a color image. Image edges and textures are also useful attributes for segmentation. 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. Segmentation does not involve classifying each segment. The segmentor only subdivides an image; it does not attempt to recognize the individual segments or their relationships to one another.

There is no theory of image segmentation. As a consequence, no single standard method of image segmentation has emerged. Rather, there are a collection of ad hoc methods that have received some degree of popularity. Because the methods are ad hoc, it would useful to have some means of assessing their performance. Haralick and Shapiro[2] have established the following qualitative guidelines for “good” image segmentation:

(a) Regions of the image segmentation should be uniform and homogeneous with respect to some characteristic such as gray tone or texture.

(b) Region interiors should be simple and without many small holes

(c) Adjacent regions of segmentation should have significantly different values with respect to the characteristic on which they are uniform.

(d) Boundaries of each segment should be simple, not ragged, and must be spatially accurate.

1.3 Literature Survey

Several techniques have been proposed for segmentation of color images. Region based segmentation of color is given by Yu-Ichi Ohta *et al* in [3]. The authors attempt to derive a set of effective color features by systematic experiments of region segmentation but it does not guarantee contiguity of the resulting regions. Edge detection based techniques [4] pose the difficulty of determining the boundary of an image due to the ambiguity of the response of a weak edge. Recently, Arbelaez *et al.* in [5] have proposed a hierarchical segmentation obtained from the output of a contour detector which overcomes the difficulties of weakly linked boundaries. In [6] color segmentation by region growing and merging is investigated. One drawback of the conventional region growing technique is the selection of the seed point and the order in which regions grow or merge. In [7], the problem of seed selection is solved by using the relaxation labeling technique which yields satisfactory results. Recent techniques for region growing use automated seed selection process as in [8] which use a fuzzy similarity and fuzzy distance based approach. In [9], after the region growing of similar color, Markov Random Fields (MRF) are applied to improve the results. However, it is observed that some homogeneous regions may get disconnected due to the MRF process. Blobworld[10], a popular image segmentation and retrieval algorithm groups pixels into regions by modeling the joint distribution of color texture and position features by a mixture of Gaussians with parameters being decided by the expectation maximization algorithm. However, the resulting blobs may not contain all the details of objects and also may not distinguish an object which is not visually distinct. Further an iterative post processing step is required to correct the mis-alignment of object boundaries.

Mean-Shift filtering [11] and Graph partitioning[12,13] methods and their hybrids[14] perform clustering in feature–space and are found to be effective for color segmentation. But they are very sensitive to the parameters like color bandwidth (Mean-Shift) and the threshold edge length (Graph method). Neural network based approaches [15,16] for image segmentation like Competitive Learning Neural Network (CLNN) and Self Organizing of Kohonen Feature Map (SOFM) avoid complex programming but usually consume a lot of training time. Other significant works on image segmentation include: Watershed technique[17] based on the morphological watershed transform, segmentation using K-nearest neighbor (K-NN) technique[18] which is sensitive to the choice of reference sample and JSEG[19]-a segmentation algorithm based on color and texture. Various combinations of popular segmentation algorithms like region merging and graph partitioning[20], mean-shift and region merging[21], watershed and Kohonen SOM[22] have been suggested together with their advantages. In [23], color segmentation is carried out by applying a set of fuzzy if-then-rules on 200 fixed color samples. The Fuzzy C-Means (FCM) clustering method, a popular choice for color segmentation has been investigated in the works of [24]. The results are quite good but for the computational complexity and sensitivity to the initialization. Several variants of FCM are summarized in [16]. In [25] fuzzy set theory and maximum fuzzy entropy principle are used to convert the image to the fuzzy domain and a Space Scale filter is used to analyze the homogeneity histogram to find the appropriate segments. Fuzzy co-clustering algorithm with its dual fuzzy (object and feature) membership functions was originally derived for document clustering, examples being FCCM, FCoDoK [26,27] and robust versions PFCC[28], RFCC[29]. Fuzzy Co-Clustering was then applied to images in 2013 by M.Hanmandalu *et al* as described in [30].

In order to make clustering more robust, many spatial clustering methods [31]–[34], which can deal with original image without filtering, have been proposed. Ahmed *et al*. [34] proposed bias-corrected fuzzy C-means (FCM) (BCFCM). In BCFCM, the label of a pixel is determined by both spectral features of the pixel and its mean-filtered neighbors, and a parameter α controls the effect of neighbors. Tarabalka *et al*. [31] proposed a spectral–spatial method for hyperspectral images. The homogeneous regions are obtained by combining the results of support vector machine and clustering using majority voting. Liew *et al*. [32] presented an adaptive FCM which utilizes local contextual information to impose local spatial continuity and allows the suppression of noise and resolves ambiguities. Dulyakarn and Rangsanseri [33] added a priori spatial information with FCM, and the remote sensing image experiments showed that the spatial information improved the segmentation results. As one of the best methods, Krinidis and Chatzis [35] presented a robust image clustering method called fuzzy local information C-means (FLICM). FLICM is a noise insensitive method without a priori knowledge. The clustering is dependent on both the spectral and local spatial information which cooperate by using a fuzzy factor. However, this method assumes that the label of one pixel is related to the labels of its spatial neighbors. Therefore, incorrect cluster labels may be assigned to the pixels near the edges of regions, and edges will be dislocated consequently. Therefore, an image spatial clustering method, called fuzzy C-means with edge and local information (FELICM), which reduces the edge degradation by introducing the weights of pixels within local neighbor windows is presented in [36] by Nan Li *et al*.

1.4 Problem Statement

Various image segmentation techniques have been proposed over time for improving segmentation of an image and the participation of a pixel within a region or cluster as a crisp or a fussy entity has been debated widely. K-Means, Fuzzy C-Means , FCCI have been widely regarded as some of the most efficient algorithms for the image segmentation process by various researchers and scholars. FCM (Fuzzy C-Means) has been widely acknowledged as a very efficient clustering method but one of the major shortcomings of FCM is that it considers only a single feature of the image for optimization purposes. FCCI on the other hand is a co-clustering approach which incorporates two membership functions which intern ensures more accurate and well defined regions/clusters. Both FCM and FCCI algorithms though being highly efficient in most cases, loose there efficiency to segment images adequately in the presence of noise. Many variations of FCM have been proposed to overcome this drawback. Edge based weight distribution is one of the most efficient variation of FCM given, which ensures overcoming of its inefficiency with noisy images. But as it's a variant of FCM, hence the pre-existing problem of a single feature optimization remains. In this thesis a new technique has been proposed which integrates the edge based weight distribution and FCCI methods ensuring efficient results by incorporating the advantages of both the methods.

1.5 Thesis Organization

The remainder of the thesis is organized as follows: chapter two gives a brief introduction about what is image segmentation and various techniques and applications of image segmentation. Then chapter three consists of description about edge based weight distribution and its effect on optimization of the objective function. The fourth chapter presents the introduction of previously implemented techniques for image segmentation. Similarly, chapter five provides a detailed discussion of the proposed approach and its advantages over previous techniques. Chapter six presents the results and validates the output clusters on the basis of cluster similarity factor. Also it presents some sample results for qualitative evaluation and discussion. Finally, chapter seven concludes the thesis and presents ideas for future work.

Chapter 2

IMAGE SEGMENTATION

2.1 Introduction

Image segmentation is one of the most significant tasks of image analysis. Segmentation is defined as a technique which subdivides an image into its constituent regions or objects depending on the problem being solved [1]. The process of image segmentation should stop when the object of interest is found. Every pixel in an image is allocated to one of the regions. The purpose of image segmentation is to divide an image into regions having high degree of correlation with objects of significance in the image. A good segmentation is typically one in which pixels in the same category have similar grey scale of multivariate values and form a connected region and neighboring pixels which are in different categories have dissimilar values.Segmentation can be mainly classified into complete and partial. The resultant disjoint regions of complete segmentation correspond exclusively with input image objects. While the resultant regions in partial segmentation do not correspond exclusively with input image. Image segmentation is often considered as a pattern recognition problem since classification of pixels is involved in the process of segmentation [38].

2.2 Techniques of Image Segmentation

On the basis of different properties of an image, the approaches to image segmentation can be classified into discontinuity based segmentation and similarity based segmentation [37]. In discontinuity based segmentation the image is divided on the basis of sudden change in intensity. This includes techniques like edge detection. While in similarity based segmentation the image is divided into regions which are similar depending on a set of predefined criteria. This includes techniques like thresholding, region growing and clustering.

2.2.1 Edge Detection Based Segmentation

Edge detection based segmentation attempts to resolve image segmentation by detecting the edges or pixels between different regions that have abrupt changes in intensity [37]. This results in a binary image. Edge based techniques do not require a priori information about the image content and is comparatively faster in computation. On the basis of theory edge based segmentation methods can be classified into gray histogram and gradient based method [39]. Region boundaries and edges are strongly related as there is often a sharp transition in intensity at the region boundaries. Another segmentation technique use edge detection techniques as the basis for segmentation process. The edges identified by edge detection are often disconnected. To segment an object from an image however, one needs closed region boundaries. The desired edges are the boundaries between such objects.

2.2.2 Threshold Based Segmentation

Thresholding based segmentation is easy but effective technique for segmenting images having light objects on dark background [37]. Thresholding algorithms are fast and economical in computation but require prior knowledge about image. Multilevel image is converted into a binary image by applying thresholding operation. A proper threshold T is chosen, to divide image into objects and split objects from background. Any pixel (x, y) is assigned to a region whose intensity is greater than or equal to threshold value T, else pixel is assigned to background [40]. On the basis of choosing the value of threshold, two types of thresholding techniques are global and local Thresholding. In global thresholding T is constant whereas in local thresholding there can be multiple thresholds to compensate uneven illumination. Threshold selection is generally done manually however it is possible to choose threshold by automatic thresholding selection algorithms. Limitation of thresholding method is that it is sensitive to noise as it does not take into account spatial features of an image.

2.2.3 Region Based Segmentation

Region based segmentation is relatively simple and has higher noise immunity as compared to edge detection method [39]. Region based segmentation is used to divide an image into regions that are similar according to some predefined criteria [37]. Segmentation techniques based on region mainly include following methods:

Region Growing: Region growing is a process [41] that group pixels in whole image into sub regions or larger regions based on predefined criterion [1]. Region growing can be achieved by performing the following steps:-

- Select a group of seed pixels in original image.
- Select a set of similarity criterion such as gray level intensity or color and set up a stopping rule.
- Grow regions by appending to each seed those neighbouring pixels that have predefined properties similar to seed pixels.
- Stop region growing when no more pixels met the criterion for inclusion in that region.

Region Splitting and Merging: Rather than choosing seed points, user can divide an image into a set of arbitrary unconnected regions and then merge the regions [39] in an attempt to satisfy the conditions of reasonable image segmentation. Region splitting and merging is usually implemented with theory based on quad tree data.

Let R represent the entire image region and select a predicate Q

- We start with entire image if Q(R) = FALSE [1], we divide the image into quadrants, if Q is false for any quadrant that is, if Q (Ri) = FALSE, We subdivide the quadrants into sub quadrants and so on till no further splitting is possible.
- If only splitting is used, the final partition may contain adjacent regions with identical properties. This drawback can be remedied by allowing merging as well as splitting i.e. merge any adjacent regions Rj & R_k for which, Q (Rj U Rk) = TRUE
- Stop when no further merging is possible.

2.2.4 Clustering Based Segmentation

Clustering is a straight forward technique for classification and easy to implement. Clustering is an unsupervised learning algorithm, where one needs to identify a finite set of categories known as clusters to classify pixels. Clustering use no training stages rather train themselves using available data. Clustering is mainly used when classes are known in advance. A similarity criteria is defined between pixels [42], and then similar pixels are grouped together to form clusters. The grouping of pixels into clusters is based on the principle of maximizing the intra class similarity and minimizing the inter class similarity. The quality of a clustering result depends on both the similarity measure used by the method and its implementation. Clustering algorithms are classified as hard clustering and soft clustering.

2.3 Applications

Some of the practical applications of image segmentation are:

1. Content-based image retrieval: Content-based image retrieval (CBIR), also known as query by image content (QBIC) and content-based visual information retrieval (CBVIR) is the application of computer vision techniques to the image retrieval problem, that is, the problem of searching for digital images in large databases. "Content-based" means that the search analyzes the contents of the image rather than the metadata such as keywords, tags, or descriptions associated with the image. The term "content" in this context might refer to colors, shapes, textures, or any other information that can be derived from the image itself. CBIR is desirable because searches that rely purely on metadata are dependent on annotation quality and completeness.

2. Machine vision: Machine vision (MV) is the technology and methods used to provide imaging-based automatic inspection and analysis for such applications as automatic inspection, process control, and robot guidance in industry. The primary uses for machine vision are automatic inspection and industrial robot guidance. Common machine vision applications include quality assurance, sorting, material handling, robot guidance, and optical gauging.

3. Medical imaging: Medical imaging is the technique, process and art of creating visual representations of the interior of a body for clinical analysis and medical intervention. Medical imaging seeks to reveal internal structures hidden by the skin and bones, as well as to diagnose and treat disease. Medical imaging also establishes a database of normal anatomy and physiology to make it possible to identify abnormalities.

- Locate tumors and other pathologies
- Measure tissue volumes

- Diagnosis, study of anatomical structure
- Surgery planning
- Virtual surgery simulation
- Intra-surgery navigation

4. Object detection: Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance.

- Pedestrian detection
- Face detection
- Brake light detection
- Locate objects in satellite images (roads, forests, crops, etc.)

Several general-purpose algorithms and techniques have been developed for image segmentation. To be useful, these techniques must typically be combined with a domain's specific knowledge in order to effectively solve the domain's segmentation problems.

Chapter 3

EDGE DETECTION

3.1 Introduction

There has been an abundance work on different approaches to the detection of one dimensional feature in images. The wide interest is due to the large number of vision applications which use edges and lines as primitives, to achieve higher level goals. Some of the earliest methods of enhancing edges in images used small convolution masks to approximate the first derivative of the image brightness function, thus enhancing edges; e.g., see (Prewitt, 1970; Sobel, 1990). These filters give very little control over smoothing and edge localization. In (Canny, 1983) Canny described what has since become one of the most widely used edge finding algorithms.

3.2 SUSAN Technique

The SUSAN edge finder has been implemented using circular masks (sometimes known as windows or kernels) to give isotropic responses as explained in [43]. The usual radius is 3.4 pixels (giving a mask of 37 pixels), and the smallest mask considered is the traditional three by three mask. The mask is placed at each point in the image and, for each point; the brightness of each pixel within the mask is compared with that of the nucleus.

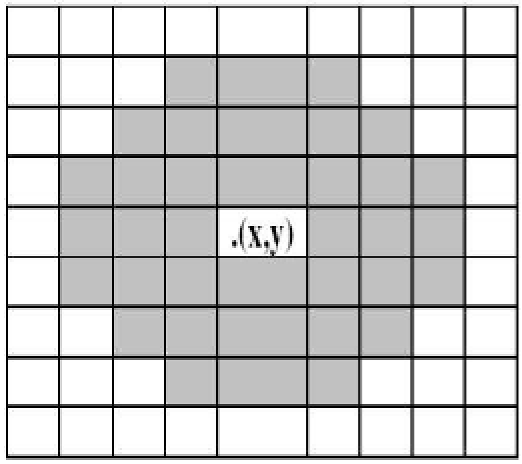


Figure 3.1: A circular mask

Originally a simple equation determined this comparison as in equation 1.

$$c(\vec{r}, \vec{r}_0) = \begin{cases} 1 & \text{if } |I(\vec{r}) - I(\vec{r}_0)| \le t \\ 0 & \text{if } |I(\vec{r}) - I(\vec{r}_0)| > t, \end{cases} \qquad (1)$$

where, r_o is the position of the nucleus in the two dimensional image, r is the position of any other point within the mask, $I(r)$ is the brightness of any pixel, t is the brightness difference threshold and c is the output of the comparison. This comparison is done for each pixel within the mask, and a running total, n, of the outputs (c) is made;

$$n(\vec{r}_0) = \sum_{\vec{r}} c(\vec{r}, \vec{r}_0). \qquad (2)$$

This total n is just the number of pixels in the USAN, i.e., it gives the USAN's area. As described earlier this total is eventually minimized. The initial edge response is then created by using the following rule:

$$R(\vec{r}_0) = \begin{cases} g - n(\vec{r}_0) & \text{if } n(\vec{r}_0) < g \\ 0 & \text{otherwise,} \end{cases} \qquad (3)$$

where, $R(r_o)$ is the initial edge response. This is clearly a simple formulation of the SUSAN principle, i.e., the smaller the USAN area, the larger the edge response. The algorithm as described gives quite good results, but a much more stable and sensible equation to use for *c* in place of Eq. (1) is

$$c(\vec{r}, \vec{r}_0) = e^{-\left(\frac{I(\vec{r}) - I(\vec{r}_0)}{t}\right)^6}. \qquad (4)$$

The form of Eq. (4) was chosen to give a "smoother" version of Eq. (1). This allows a pixel's brightness to vary slightly without having too large an effect on *c*, even if it is near the threshold position.

Computation of the edge direction is necessary for a variety of reasons. Firstly, if non-maximum suppression is to be performed the edge direction must be found. It is also necessary if the edge is to be localized to sub-pixel accuracy. Finally, applications using the final edges often use the edge direction for each edge point as well as its position and strength. In the case of most existing edge detectors, edge direction is found as part of the edge enhancement. As the SUSAN principle does not require edge direction to be found for enhancement to take place, a reliable method of finding it from the USAN has been developed. The direction of an edge associated with an image point which has a non zero edge strength is found by analyzing the USAN in one of two ways, depending on the type of edge point which is being examined.

With respect to the scale-space behaviour of the SUSAN edge detector, scale-space graphs showing edge localization against mask size (e.g., plotting a single horizontal line from the edge image against mask size, in the manner of Witkin (1983)) give vertical lines. (Most edge detectors do not give scale-invariant edge positions, thus producing curves in scale-space graphs.) This is obviously a desirable feature, as it means that accuracy does not depend on mask size. This is to be expected; the minimum USAN area when approaching an edge occurs on top of the edge regardless of the mask size.

In summary then, the algorithm performs the following steps at each image pixel:

1. Place a circular mask around the pixel in question (the nucleus).

2. Using Eq. (4) to calculate the number of pixels within the circular mask which have similar brightness to the nucleus. (These pixels define the USAN.)

3. Using Eq. (3) to subtract the USAN size from the geometric threshold to produce an edge strength image.

4. Use moment calculations applied to the USAN to find the edge direction.

5. Apply non-maximum suppression, thinning and sub-pixel estimation, if required.

Chapter 4

PIXEL WEIGHT DISTRIBUTION

4.1 Weight Distribution

Weight Distribution to the pixels in an image is done on the basis of procedure stated in [36]. If the straight line between two pixels is cut off by an edge, these two pixels belong to "different regions." Here, the "different regions" are used to find the most appropriate spatial neighbors, but they are not really different regions. Pixels in different regions will set weights which are different from those of pixels in the same regions.

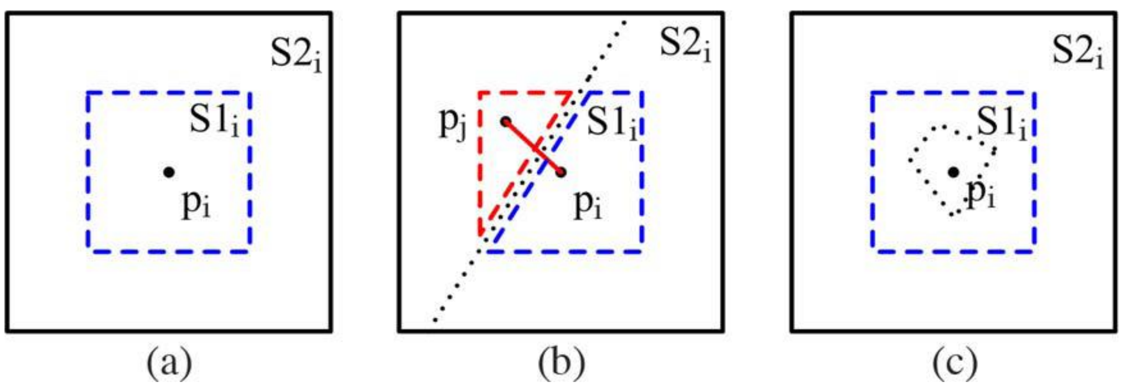


Figure 3.2: Three cases of the windows setting in weight distribution.

Figure 3.2 shows the three cases of the windows setting, where the dash–dot lines represent the edges detected SUSAN and $S1_i$ and $S2_i$ are the spatial neighbor windows with the same center pixel p_i.

The weights for pixels of local neighbors can be set by the following method:

1) Suppose that the pixels in $n \times n$ neighbor window may be affected by the label of the center pixel p_i, i.e., the size of spatial neighbor window $S1_i$ is $n \times n$, the subscript i means that the window center is p_i.

2) Get a $m \times m$ spatial neighbor window $S2_i$ which has the same center as window $S1_i$, where $m = 2\times n + 1$.

3) Count the number of neighbor pixels which are not separated from the center pixel p_i by edges in window $S2_i$, and set the total number as t.

4) If t is more than n^2, the weight of pixel p_j within window $S1_i$ is set as

$$w_{ij} = \begin{cases} 0.33 & \text{if } p_i \text{ and } p_j \text{ are separated by the edge} \\ 1 & \text{Otherwise.} \end{cases}$$

If p_i and p_j are separated by edges, the weight of pixel p_j should be set as zero. Here, we set the weight as 0.33 for weakening the possible errors in edge detection.

5) If t is less than n^2, the weight of pixel p_j within window $S1_i$ is $w_{ij} = 1$. Based on the aforementioned settings, there exist three cases.

Case 1) There is no edge within window $S1i$, as shown in Fig. 3.2(a). The weights of all the pixels in $S1i$ are set as one.

Case 2) One edge exists in window $S1_i$, and the total number t in window $S2_i$ is more than n^2, as shown in Fig. 3.2(b). Different weights will be set for the neighbor pixels in $S1_i$. As the neighbor pixel p_j and center pixel p_i are separated by the edge, these two pixels may belong to different regions, so the weight of neighbor pixel p_j is set as 0.33, and its impact on center pixel p_i has also been reduced.

Case 3) The edges exist in window $S1_i$, and the total number t in window $S2_i$ is less than n^2, as shown in Fig. 3.2(c). This case may be caused by the noise or the edge error, and the weights of all pixels in window $S1_i$ are set as one, so the effect of errors can be weakened by the spatial neighborhood.

According to the aforementioned weight setting method, different weights will be assigned to all pixels in the image, and then, the edges of regions can be retained well by using the weighted windows.

Chapter 5

CLUSTERING TECHNIQUES

5.1 Introduction

Few commonly used techniques for autonomous image segmentation are applied in this study and their results are compared with the proposed algorithm. Performance evaluation of segmentation methods is a tough task as different parameter settings can affect the results significantly. The problem of over-segmentation and under-segmentation is also quite crucial in this context. All of the segmentation methods used in this study is selected so that there are very few user-defined parameters required. We have used three traditional methods described in the following sections for image segmentation, namely k-means clustering, fuzzy c-means clustering and fuzzy co-clustering for images algorithm.

5.2 K-Means

K-means is a type of hard clustering algorithms. K-means clustering technique belongs to the category of unsupervised cluster analysis algorithms. Given 'n' number of observations, this algorithm groups these observations into clusters [44]. The observations that belong to same cluster are alike in nature and those belonging to different clusters are different in nature.

The number of clusters 'k' is assumed to be fixed. Each cluster has a leader called 'centroid'. Cluster centroids are initialized with random values. The sum of squares of distance between observation and cluster centroid is minimized iteratively. Centroid is then recalculated until convergence. The process of K-means clustering is illustrated in figure 4.1 with some random data points.

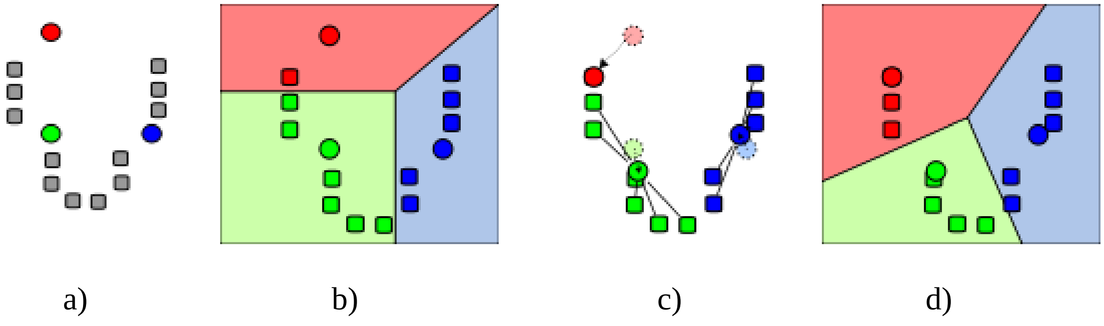


Figure 4.1: a) Initial clusters (k=3) b) all observations assigned to corresponding cluster c) calculation of new cluster centers d) steps 2 and 3 repeated until convergence.

Steps of the algorithm are:

Step 1: Cluster centroids are initialized with random values. These observations represent temporary cluster centers.

Step 2: For each observation, calculate sum of square of distance between centroid and the observation. Based on this distance, assign each observation to closest cluster. Mathematically it can be given by equation (1)

$$\min \sum_{i=1}^{k} \sum \| x_j - u_i \|^2 \qquad \text{.......(6)}$$

Where u_i are the cluster centroids and x_j represents other observations.

Step 3: A new centroid is calculated for each cluster and is replaced with previous cluster centroid.

Step 4: Repeat steps 2 and 3 until no cluster centroid changes.

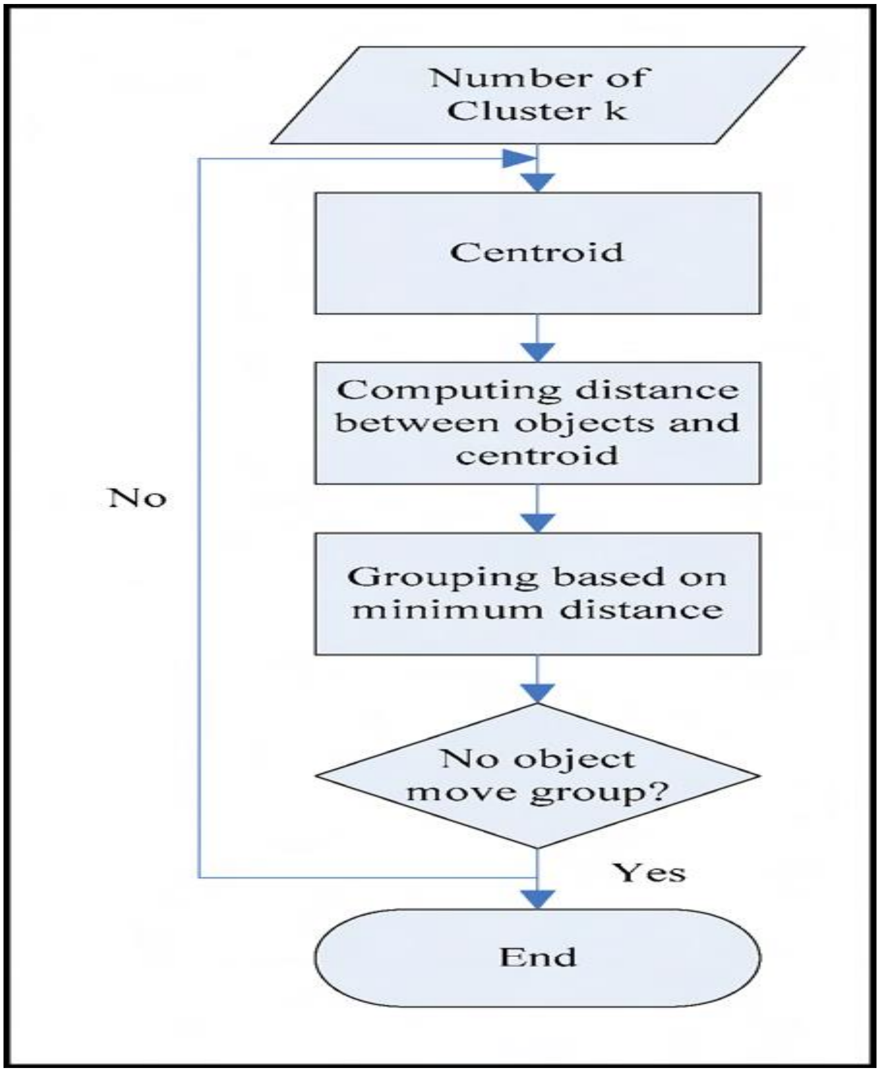


Figure 4.2: Flowchart for K-means Clustering Algorithm

5.3 Fuzzy C-Means

As discussed in previous section clustering assigns observations into different groups (clusters) such that observations belonging to the same cluster are more similar to one another than observations belonging to different clusters [45]. A large number of clustering schemes are proposed in literature. One major classification of schemes is hard clustering and the fuzzy clustering (or soft clustering). In hard clustering each observation or point in the dataset either belongs to a particular cluster or does not belongs. This leads to very crisp segmentation results. i.e., each pixel of the image belongs to exactly one class. Hard segmentation does not provide satisfactory results in images with issues like poor contrast, limited spatial resolution, overlapping intensities etc. Fuzzy clustering on the other hand is a soft clustering technique which allows partial belongingness of pixels into different clusters. This partial membership is calculated using membership functions. The sum of all membership degrees for any given data point is equal to 1. This method has better applicability to segmentation applications than hard clustering. Dunn *et al* first introduced FCM which was later extended by [45].

The algorithm finds c clusters by minimizing the objective function given by equation 7.

$$J_{FCM} = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^q d^2 (x_k, v_i) \quad \ldots\textbf{(7)}$$

where X = {x1, x2, …, xn} are the data points, n is the number of data items, c represents the number of clusters, the degree of membership of x_k in the i_{th} cluster is represented as u_{ik}, q is a weighting exponent on each fuzzy membership, vi represents the centre of cluster i, d2 (x_k, v_i) is a distance between data point x_k and cluster centre v_i. Objective function is minimized using iterative process as follows:

Step 1: Initialize parameters like c, q.

Step 2: Initialize the fuzzy partition matrix U = [u_{ik}].

Step 3: Initialize the loop counter b= 0.

Step 4: Calculate the c cluster centers v_i^b with U using equation 8:

$$v_i^{(b)} = \frac{\sum_{k=1}^{n} (u_{ik}^{(b)})^q x_k}{\sum_{k=1}^{n} (u_{ik}^{(b)})^q} \quad \ldots\textbf{(8)}$$

Step 5: Calculate the membership U $^{(b+1)}$. For k = 1 to n, calculate the following:

$$I_k = \{i \mid 1 \le i \le c, d_{ik} = \| x_k - v_i \| = 0\};$$

for the kth column of matrix, compute new membership values.

$$a) \quad if\, I_k \ne \phi, then$$

$$u_{ik}^{(b+1)} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{(q-1)}}}$$

$$b)\ else \quad u_{ik}^{(b+1)} = 0 \; for\, all\, i \notin I \; and \sum_{i \in I_k} u_{ik}^{(b+1)} = 1; next\, k.$$

Step 6: If $\| U^{(b)} - U^{(b+1)} \| < \varepsilon, stop; otherwise, b = b+1 and\ go\ to\ step\, 4.$

5.4 FCCI

Fuzzy Co-clustering for images (FCCI) [30] algorithm incorporates the distance between each feature data point and the feature cluster centre as the dissimilarity measure and the entropies of objects and features as the regularization terms in the objective function. In FCCI, the images captured are transformed from RGB to CIELAB color space.

The advantages offered by FCCI algorithm are as follows:

- It is insensitive to initialization and form distinct clusters. (Fuzzy clustering)
- It performs well in high dimensions and provides well defined clusters. (Co-clustering)
- It minimizes the impact of outliers to improve the accuracy of co-clustering. (ranking/feature memberships)
- Its objective function integrates the distance measure of input features w.r.t. feature centroid into the entropy regularization framework.
- It is reasonably fast enough.

The objective function J_D to be minimized in the FCCI algorithm is stated below in equation 9:

$$J_D = \sum_{c=1}^{C} \sum_{i=1}^{N} \sum_{j=1}^{K} \mu_{ci} v_{cj} D_{cij} + T_U \sum_{c=1}^{C} \sum_{i=1}^{N} \mu_{ci} log\mu_{ci} + T_V \sum_{c=1}^{C} \sum_{j=1}^{K} v_{cj} log v_{cj} \qquad \ldots\textbf{(9)}$$

The total memberships of all attributes in a cluster must sum to 1, hence the above equation is subjected to the constraints:

$$\sum_{c=1}^{C} \mu_{ci} = 1, \qquad \mu_{ci} \in [0,1], \quad \forall i = 1,\ldots,N$$

$$\sum_{j=1}^{K} v_{cj} = 1, \qquad v_{cj} \in [0,1], \quad \forall c = 1,\ldots,C$$

where, C is the number of clusters, N is the number of data points i.e., N = N1*N2(N1*N2 is the dimension of the input image), K is the number of color features, μ_{ci} is membership function of data point i to cluster c, ν_{cj} is membership function of feature j to cluster c, T_U and T_V are the weighting parameters that determine the degree of fuzziness and D_{cij} is the Euclidean distance between data point X_{ij} and cluster centroid p_{cj} given as:

$$D_{cij} = d^2(X_{ij}, p_{cj}) = (X_{ij} - p_{cj})^2$$

The constrained optimization problem for FCCI algorithm is now defined as:

$$F = \sum_{c=1}^{C} \sum_{i=1}^{N} \sum_{j=1}^{K} \mu_{ci} v_{cj} D_{cij} + T_U \sum_{c=1}^{C} \sum_{i=1}^{N} \mu_{ci} log\mu_{ci} + T_V \sum_{c=1}^{C} \sum_{j=1}^{K} v_{cj} log v_{cj} +$$
$$\sum_{i=1}^{N} \lambda_i \left(\sum_{c=1}^{C} \mu_{ci} - 1 \right) + \sum_{c=1}^{C} \gamma_c \left(\sum_{j=1}^{K} v_{cj} - 1 \right)$$

…**(10)**

After solving the optimization problem with the help of Lagrange's multipliers λ_i and $\gamma_{c,}$ the value of the membership functions and centroid is calculated as:

$$\mu_{ci} = \frac{\exp(-\sum_{j=1}^{K} \frac{v_{cj} D_{cij}}{T_U})}{\sum_{c=1}^{C} \exp(-\sum_{j=1}^{K} \frac{v_{cj} D_{cij}}{T_U})}$$

$$v_{cj} = \frac{\exp(-\sum_{i=1}^{N} \frac{\mu_{ci} D_{cij}}{T_V})}{\sum_{j=1}^{K} \exp(-\sum_{i=1}^{N} \frac{\mu_{ci} D_{cij}}{T_V})}$$

$$p_{cj} = \frac{\sum_{i=1}^{N} \mu_{ci} X_{ij}}{\sum_{i=1}^{N} \mu_{ci}}$$

The steps involved in the FCCI algorithm are stated below [30]:

Step 1: Read the input RGB color image.

Step 2: Convert RGB color image to CIELAB color space.

Step 3: Define the number of color features in color space as K=2, i.e. A and B.

Step 4: Perform the 2d to 1d transformation to generate data point X_{ij} in the j_{th} dimension, j=1, 2 for each pixel, I=1,…..,N where N is the size of the data.

Step 5: Initialize the value of T_u and $T_{v.}$as 10 and 10^6 respectively.

Step 6: Define the number of clusters as C=2.

Step 7: Obtain the corresponding value of μ_{ci} for C=2.

Step 8: Defuzzification: Convert fuzzy membership matrix to a binary image by defining the cut-off/threshold point. Our algorithm is run at a threshold point of μ_{ci} (cut-off) =0.5.

$$\text{Output} \quad \begin{cases} = 1, \text{ if } \mu \geq 0.5 \\ = 0, \text{ otherwise} \end{cases}$$

Step 9: Map the selected coordinates of clusters from that of original RGB image and display all clusters.

Step 10: Reconstruct the color image, for comparison purpose, by representing each cluster with its centroid (A, B) value and the original L value. The μ_{ci} (cut-off) =0.5 threshold is found to be generally acceptable for comparison of segmentation results.

Step 11: Calculate the segmentation error ISE as the number of object pixels that have not been selected, and OSE as the number of non object pixels that have been selected. Calculate the average segmentation error.

The flowchart for the above steps of FCCI Algorithm is shown below in figure 5.1.

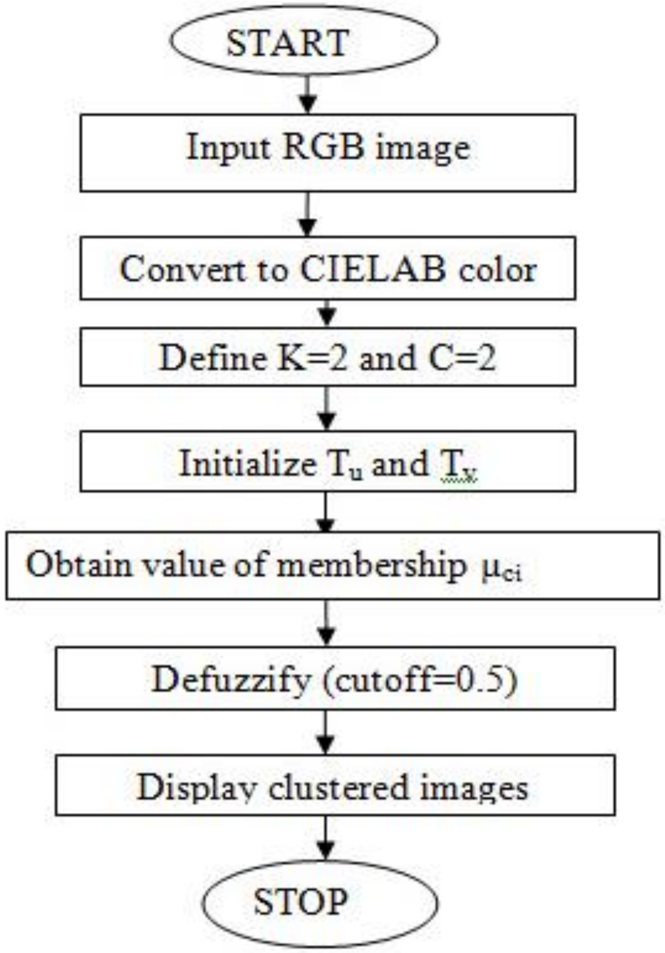


Figure 4.3: Flowchart of FCCI algorithm

5.5 Edge Based FCM

For an image of N pixels $P = \{p_1, p_2, . . . , p_N\}$, FELICM is an iterative clustering method, and its objective function can be defined as

$$J = \sum_{i=1}^{N} \sum_{k=1}^{c} [Q_{ki} + G_{ki}] \quad \text{…(11)}$$

where, N is the number of pixels, c is the number of clusters, and Q_{ki} is the distance between pixel p_i and the center of cluster k and is defined as

$$Q_{ki} = u_{ki}^m \|p_i - v_k\|^2 \quad \ldots\textbf{(12)}$$

where, m is the exponent of weight and it has been set to three in the back experiments from practical experience, u_{ki} is the membership degree of pixel p_i to the cluster k, and v_k is the prototype of the center of cluster k.

The distance G_{ki} between the spatial neighbors of pixel p_i and the center of cluster k is used to control the impact of noises and is defined as

$$G_{ki} = \sum_{\substack{j \in S1_i \\ i \neq j}} \frac{1}{d_{ij}+1} \left[w_{ij}(1-u_{kj})\right]^m \|p_j - v_k\|^2 \quad \ldots\textbf{(13)}$$

where, pixel p_j belongs to the window $S1_i$ whose center is pixel p_i, d_{ij} is the spatial Euclidean distance between p_i and p_j, and w_{ij} is the weight of pixel p_j in the window $S1_i$.

Local minimal extreme J is obtained iteratively as follows.

Step 1: Set c, m, and the stopping condition ε.

Step 2: Initialize the fuzzy partition matrix, and set all elements as $1/c$. Set the loop counter b = 0.

Step 3: Scan the image, and the cluster prototypes are calculated as

$$v_k = \frac{\sum_{i=1}^{N} u_{ki}^m p_i}{\sum_{i=1}^{N} u_{ki}^m}. \quad \ldots\textbf{(14)}$$

Step 4: The degree of membership is obtained by

$$u_{ki} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|p_i - v_k\|^2 + G_{ki}}{\|p_i - v_j\|^2 + G_{ji}} \right)^{\frac{1}{m-1}}}$$

…**(15)**

Step 5: If $\{U(b) - U(b+1)\} < \varepsilon$, then stop; otherwise, set $b = b + 1$ and go to step 3.

Step 6: Pixel *pi* is clustered into the cluster $c_i = \arg_k\{\max\{u_{ki}\}\}$

From aforementioned descriptions, it can be seen that *Gki* can be obtained without any parameter. It varies automatically according to the spectral feature of pixels in the spatial window and the fuzzy membership of pixels to the clusters. In other words, the local fuzzy membership balances the relationship between pixels and their neighbors. This preserves the details and edges in the image and improves the accuracy of clustering ultimately.

After the clustering, all pixels on the edges obtained by Susan operator will be assigned to their nearest adjacent regions, and the final clustering result is achieved.

Chapter 6

PROPOSED TECHNIQUE

6.1 Introduction

As described in the previous chapters various techniques like FCCI and Edge Based FCM have many advantages over the generally used FCM image segmentation technique. On one hand where FCCI optimizes the results over multiple objective functions, edge based FCM ensures efficiency of clusters during the presence of noise.

Hence, if we combine the above mentioned techniques it should ensure many advantages over the conventional FCM technique in various forms. Therefore, we propose edge based FCCI or EFCC algorithm as a method to ensure both optimizations over multiple functions as well as efficient clusters during the presence of noise.

6.2 Algorithm

The proposed algorithm can be applied over color images through the following steps:
Step 1: Read the RGB image.

Step 2: Detect the edges from the image using SUSAN edge detection algorithm.

Step 3: Assign weights to each pixel of the RGB image based on the technique described in chapter 4.

Step 4: Perform the clustering using FCCI as explained in chapter 5.

Step 5: Change the optimization function used in FCCI as follows:

$$J = \sum_{i=1}^{N} \sum_{k=1}^{c} [Q_{ki} + G_{ki}]$$ …**(16)**

Where

$$G_{ki} = \sum_{\substack{j \in S1_i \\ i \neq j}} \frac{1}{d_{ij} + 1} [w_{ij}(1 - u_{kj})]^m \|p_j - v_k\|^2$$ …**(17)**

$$Q_{ki} = \sum_{c=1}^{C} \sum_{i=1}^{N} \sum_{j=1}^{K} \mu_{ci} v_{cj} D_{cij} + T_U \sum_{c=1}^{C} \sum_{i=1}^{N} \mu_{ci} \log \mu_{ci} + T_V \sum_{c=1}^{C} \sum_{j=1}^{K} v_{cj} \log v_{cj}$$ …**(18)**

Subjected to constrains:

$$\sum_{c=1}^{C} \mu_{ci} = 1, \qquad \mu_{ci} \in [0,1], \quad \forall i = 1,\dots,N$$

$$\sum_{j=1}^{K} v_{cj} = 1, \qquad v_{cj} \in [0,1], \quad \forall c = 1,\dots,C$$

Where in

$$D_{cij} = d^2(X_{ij}, p_{cj}) = (X_{ij} - p_{cj})^2$$

$$p_{cj} = \frac{\sum_{i=1}^{N} \mu_{ci} X_{ij}}{\sum_{i=1}^{N} \mu_{ci}}$$

Step 6: Calculate the membership functions as:

$$u_{ki} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|p_i - v_k\|^2 + G_{ki}}{\|p_i - v_j\|^2 + G_{ji}} \right)^{\frac{1}{m-1}}} \quad \ldots\textbf{(19)}$$

Step 7: Follow the steps of the conventional FCCI using the optimization and membership functions described in the above two steps.

Step 8: Halt the iterative process if the maximum number of iterations or the minimum threshold difference between two consecutive iterations is achieved.

That is

If $\{U(b) - U(b+1)\} < \varepsilon$ or b > max iteration, then stop; otherwise, set $b = b + 1$ and continue to next iteration

Where

ε = *minimum threshold*

b = *number of iteration*

Step 9: Pixel *pi* is clustered into the cluster with center c_i as

$$c_i = \arg_k\{\max\{u_{ki}\}\}$$

Step 10: Display the clustered images.

Step 11: Calculate the mean square error as follows:

$$\text{MSE} = \sum_{i=1}^{k}\sum_{j=1}^{m*n}(p_j - c_i)^2 \qquad \ldots\textbf{(20)}$$

Subjected to constraint

$$p_j \in c_i$$

Where

k = number of clusters

m*n = dimensions of the image

p_j = current pixel under consideration

c_i = center of cluster i

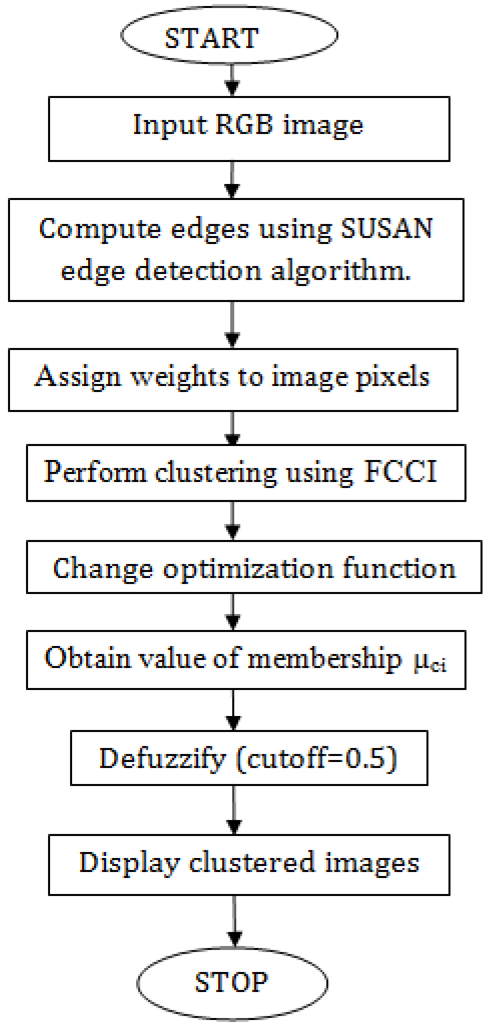


Figure 6.1: Flowchart for the proposed technique

Chapter 7

RESULT & COMPARATIVE ANALYSIS

The following system configuration has been used while conducting the experiments:

Processor: Intel Core i5

Clock Speed: 2.4 GHz

Main Memory: 4 GB

Hard Disk Capacity: 512 GB

Software Used: MATLAB 7.9.0 (2009b)

We have applied proposed scheme as explained in section 6 to segment color images on over 100 natural images taken from Berkeley's image segmentation database. 10 percent Salt and Pepper noise has been added to the original images to obtain noisy images. The 20 images from the database are illustrated in figure 7.1-7.20 with original image represented by (a) and noisy image by (b). Clustered images with four techniques represent clusters for FCM, FCM+EDGE, FCCI, and the proposed technique are shown in figure 7.1-7.20(c-f) respectively.

The results of the proposed approach are compared with the other techniques with the help of mean square error value. The quantitative analysis of the images is shown in table 7.1. As seen by the values in the table, out of the 20 images, 16 give the least MSE value for the proposed technique while in 3 images (2, 5 and 10) FCM+EDGE gives the least value while image number 14 gives the least value of MSE for FCCI closely followed by the proposed technique.

Table 7.1: MSE Values for FCM, FCCI, FCM+Edge And Proposed

S. No.	FCM	FCM+EDGE	FCCI	FCCI+EDGE
1	0.795974944	1.006997267	0.815539285	0.793294504
2	1.125172905	1.124151039	1.272412143	1.32142215
3	1.270802399	1.023901844	0.976675524	0.752616811
4	1.148822488	1.151098871	0.346599554	0.394821944
5	1.419929632	0.835916596	1.247435442	1.413199753
6	1.096326254	1.314029787	1.08028075	1.09776905
7	1.620028732	1.85216957	1.574918122	1.247543663
8	0.874048935	0.785885782	0.991835187	0.674796781
9	1.450333168	1.591103026	1.495061199	1.299547825
10	1.169716587	1.075546961	1.20562574	1.331979155
11	0.214790821	1.140137827	1.33428582	0.213694226
12	1.315615548	1.431824979	0.830601477	0.63808333
13	0.923552916	1.239727601	1.128185643	0.920553568
14	0.720055261	0.716496049	0.674680615	0.716761479
15	1.159703786	1.160637952	1.060265468	0.978238612
16	1.186725873	1.175236741	1.190185375	0.948495722
17	1.718217572	1.720483693	1.35128102	1.237847042
18	1.407452013	1.330408336	1.313986424	1.030823335
19	1.211248764	1.173037462	1.468544557	0.880679459
20	1.05569165	1.052007033	1.22662997	1.049770744

(a)

(b)

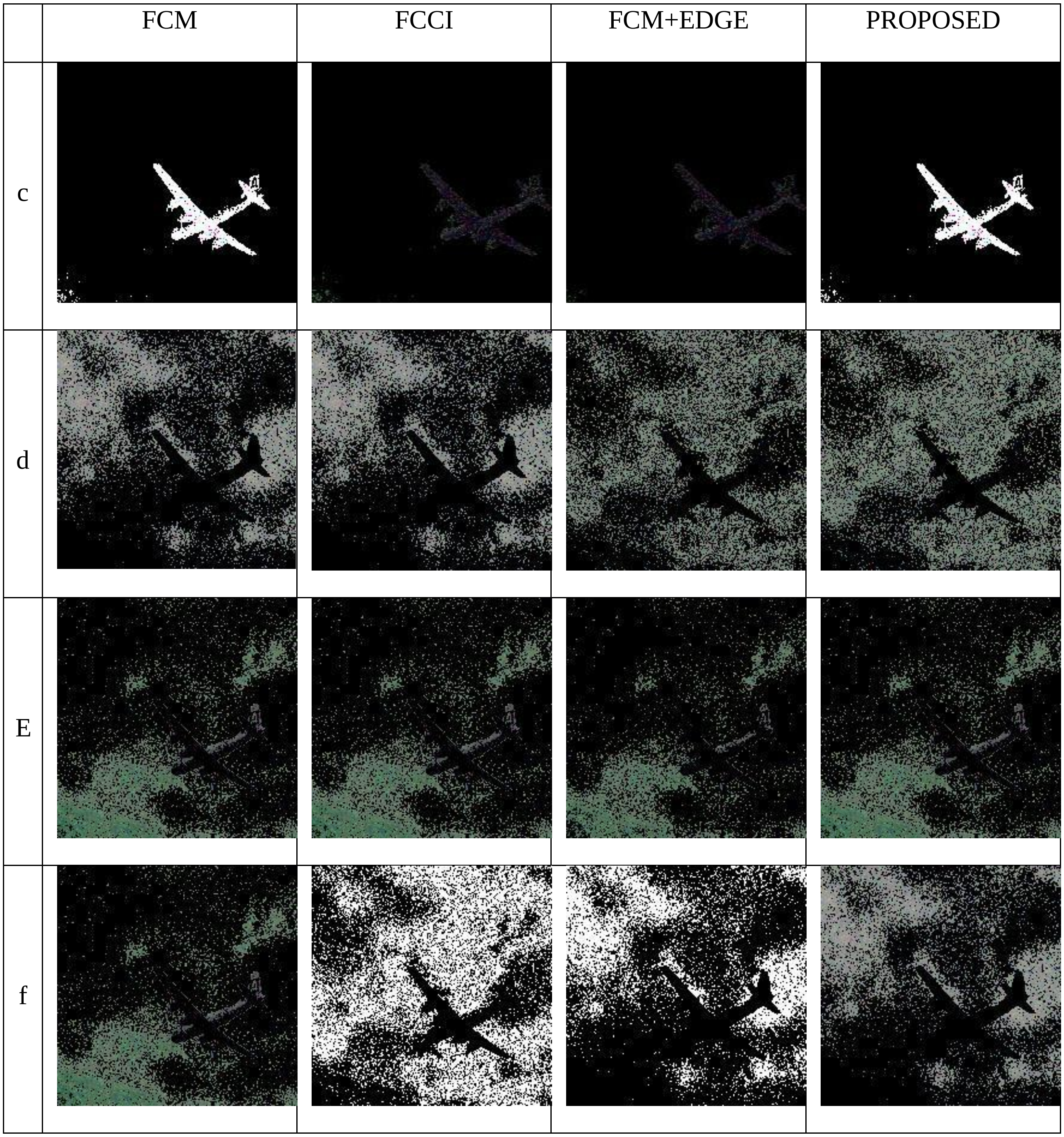


Figure 7.1 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

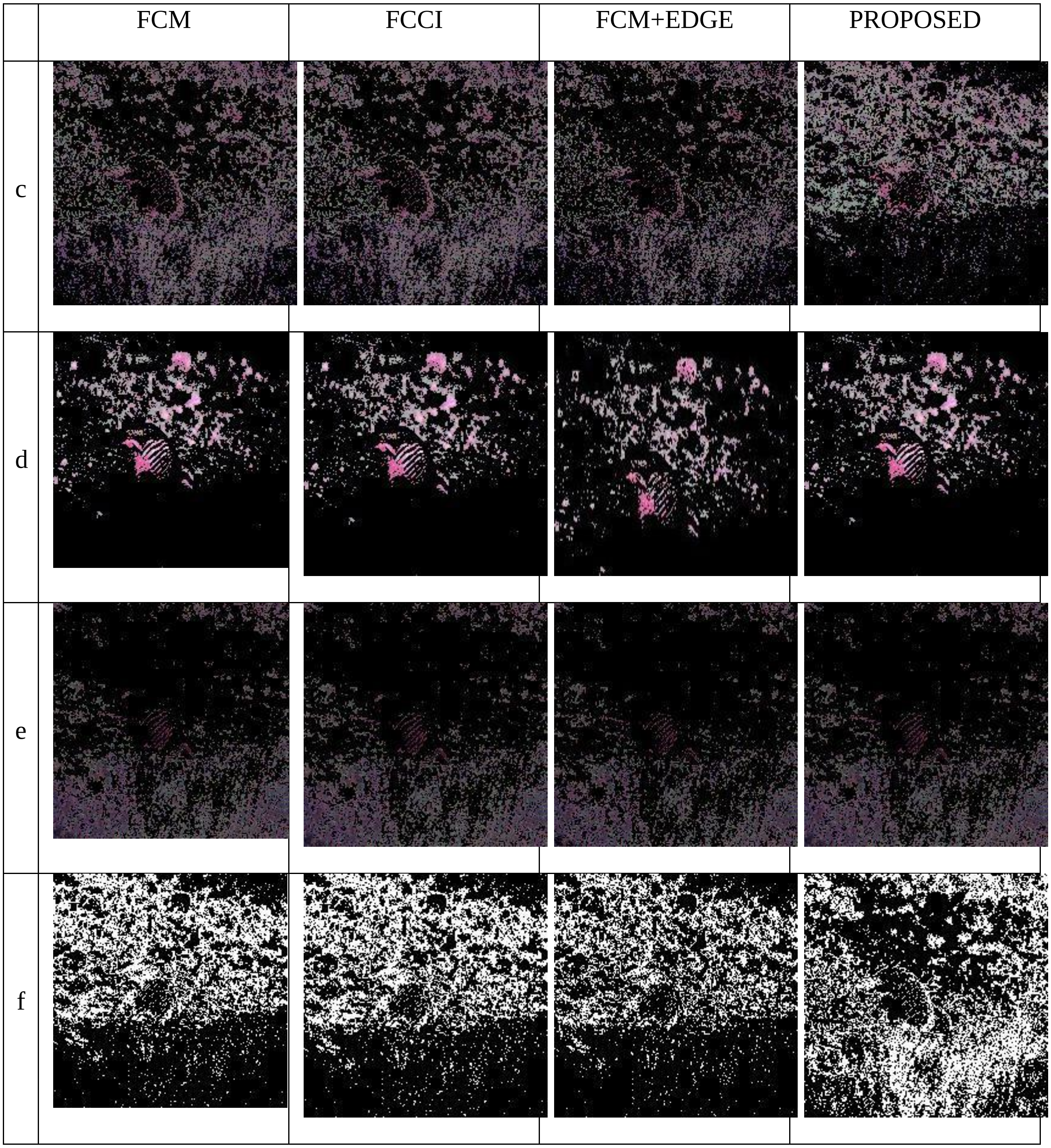
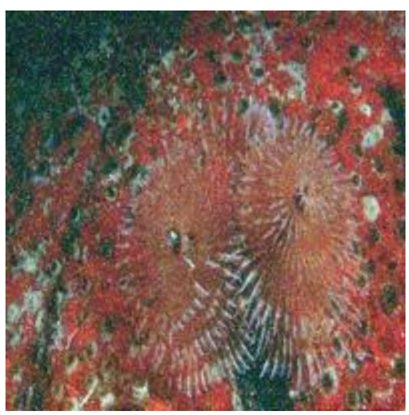


Figure 7.2 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)



(b)

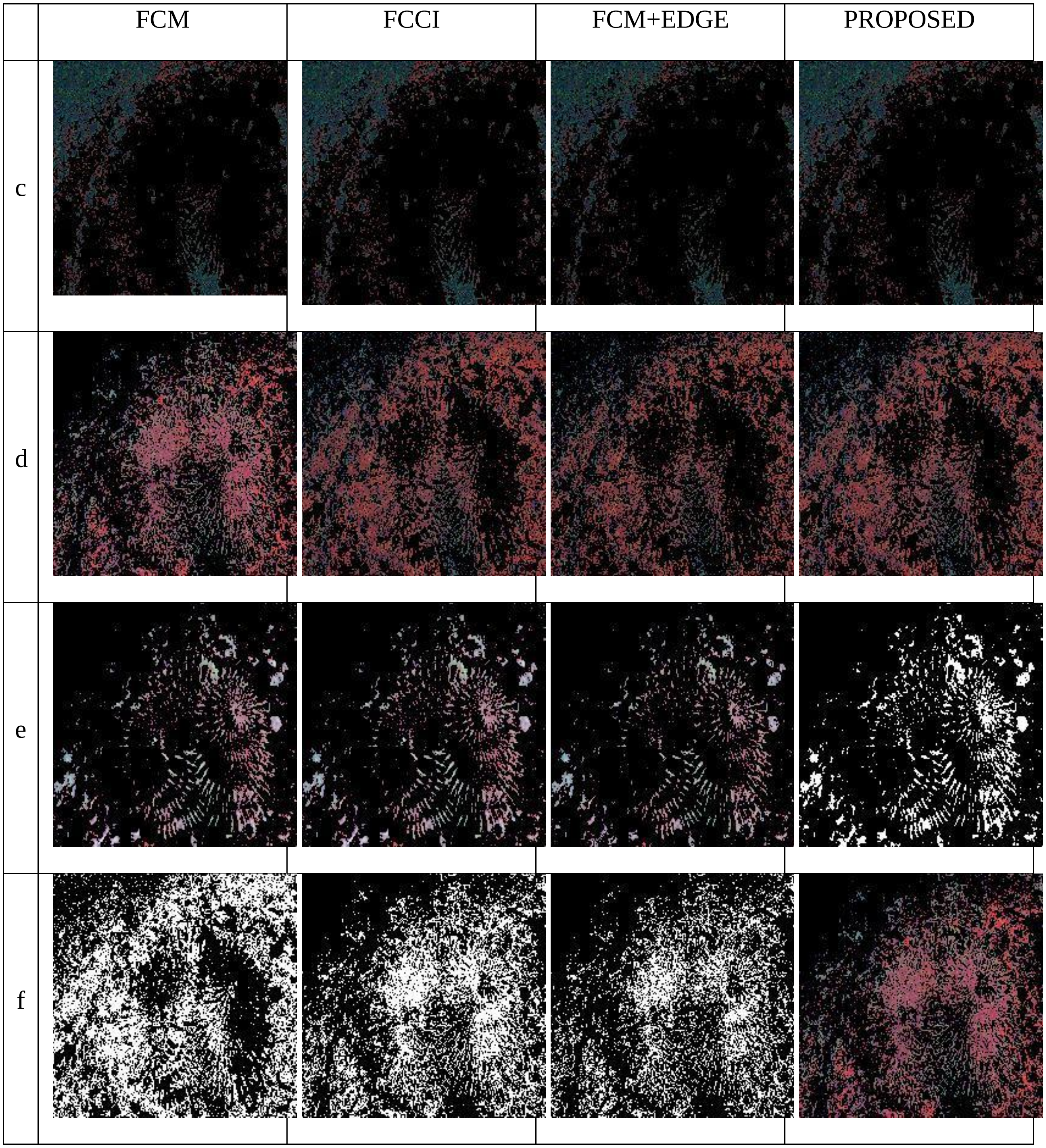


Figure 7.3 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

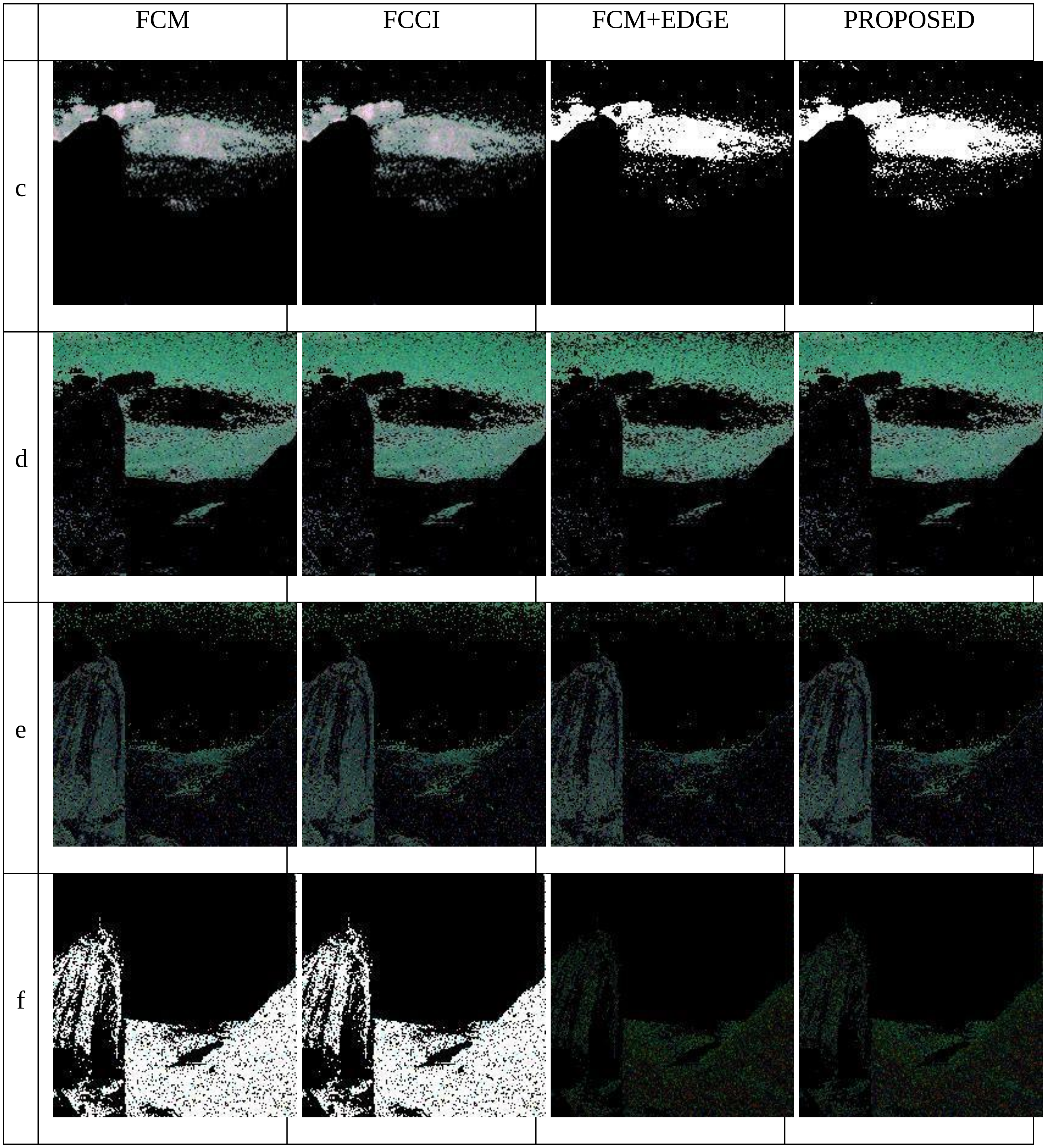


Figure 7.4 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

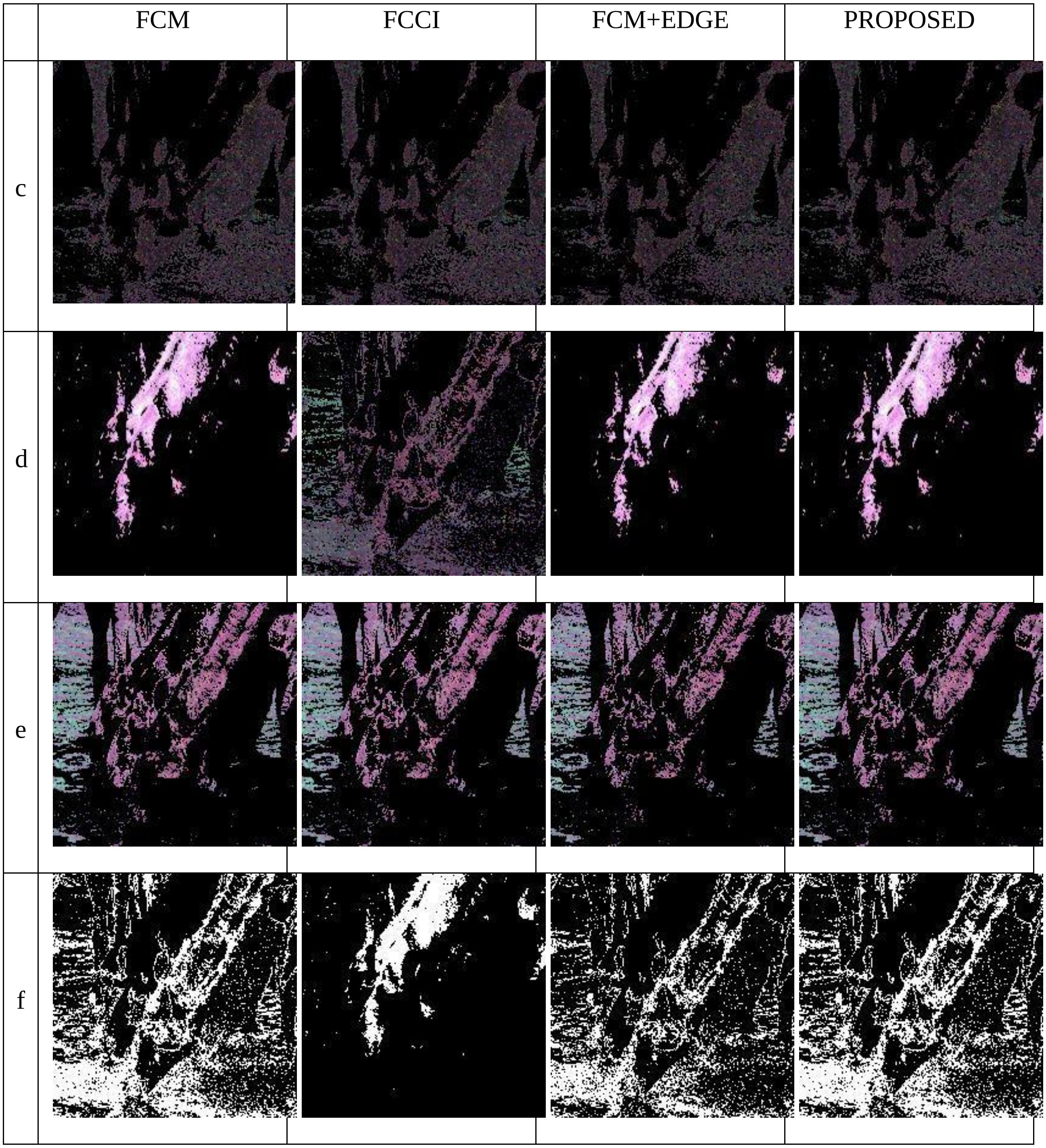


Figure 7.5 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

Figure 7.6 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

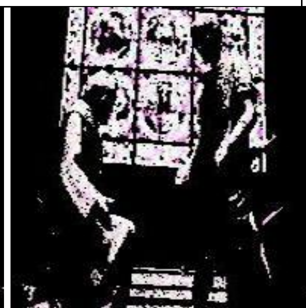
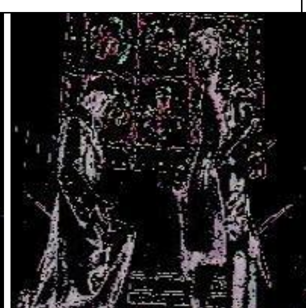
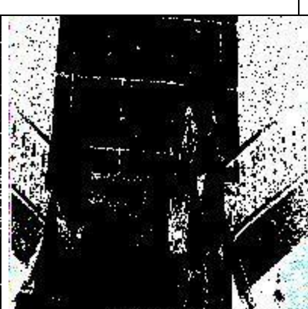
	FCM	FCCI	FCM+EDGE	PROPOSED
c				
d				
e				
f				

Figure 7.7 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

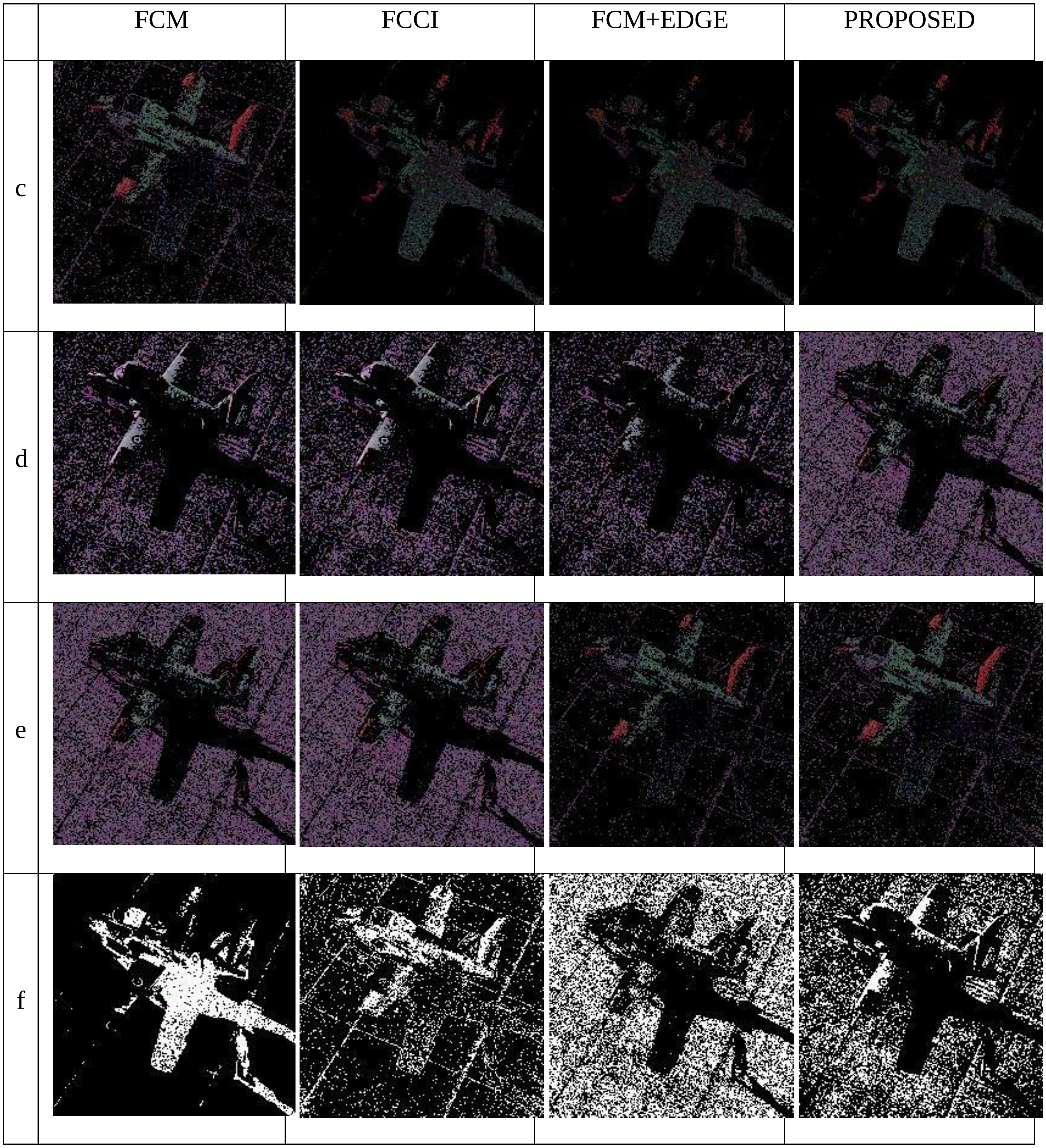


Figure 7.8 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

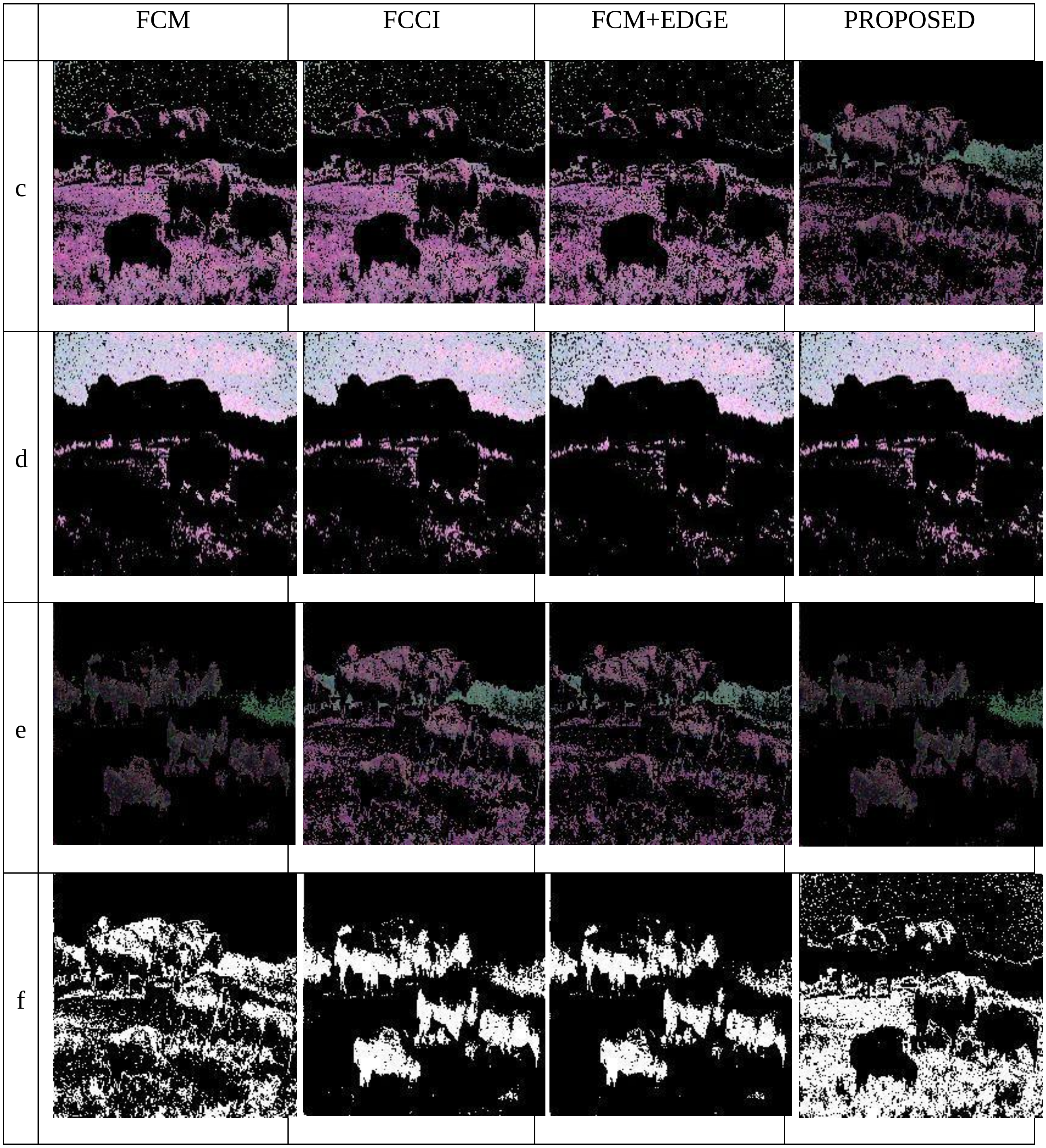


Figure 7.9 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

(a)

(b)

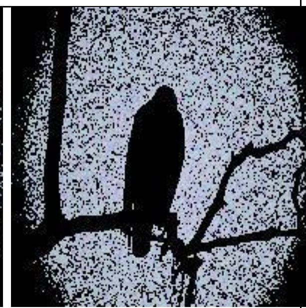
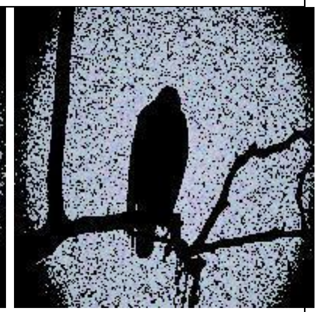
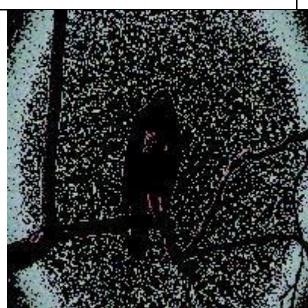
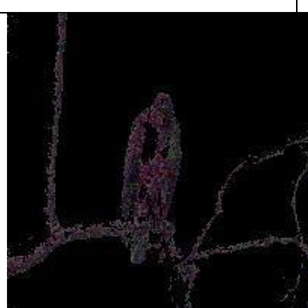
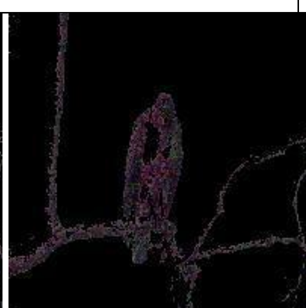
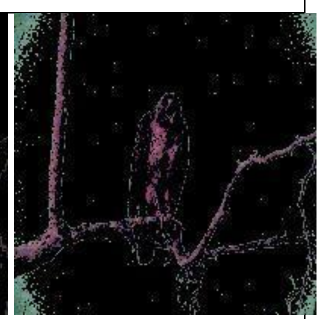
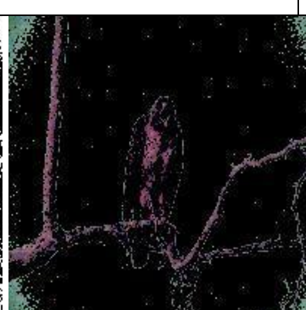
	FCM	FCCI	FCM+EDGE	PROPOSED
c				
d				
e				
f				

Figure 7.10 (a)Original image (b)Noisy image (c) Cluster 1 (d) Cluster 2 (e) Cluster 3 (f) Cluster 4

Chapter 8

CONCLUSION & FUTURE SCOPE

In this thesis, the Edge Based Fuzzy Co-clustering technique based on the assignment of weights to the edges to optimize the clustering used for the color segmentation of natural images. This thesis also discuss the comparative analysis of the proposed image segmentation techniques, with K-Means Clustering, Fuzzy C-Means(FCM) Clustering, Fuzzy Co-Clustering for Images(FCCI), and edge based FCM for segmentation of colored images. The performance evaluation of the above mentioned techniques is done on the basis of mean square error (MSE). The analysis is conducted on 100 natural images taken from Berkley image database and the proposed technique is found to outperform other techniques. In future, we can also optimize the parameters of the techniques used with the help of multi objective evolutionary algorithms to get better segmentation results. The proposed technique can also be applied for segmentation in medical images.

REFERENCES

[1] R.C.Gonzalez and R.E. Wood, *'Digital Image Processing'*, Prentice-Hall, India, Second Edition, 2007.

[2] R. M. Haralick and L. G. Shapiro, *'Image Segmentation Techniques'*, Computer Vision, Graphics, and lmage Processing, 29, 1, January 1985, 100–132.

[3] Y.C.Ohta, T.Kanade, T.Sakai, *'Color information for region segmentation'*, Comp. Graphics image processing , 133(1980) pp222-241.

[4] Jayarammamurthy , S.N and T.L.Huntsberger, *'Edge and region analysis using fuzzy sets'*, Procc. Of IEEE wrkshp. On Lang. aut. June 1985 , pp.71-75.

[5] P.Arbelaez, *'From contours to regions: an empirical evaluation'*, *CVPR*,2009.

[6] Tremeau, A., and Borel, N, *'A region growing and merging algorithm to colour segmentation'*, Pattern Recognition, 1997, 30, (7), pp. 1191–1203

[7]. S.C.Cheng, *'Region growing approach to color segmentation using 3-D clustering and relaxation labeling'*, IEEE Trans. On Vision & Image Signal processing, Aug. 2003, pp. 270-276.

[8] C.C.Kang, W.J.Wang, *'Fuzzy based seeded region growing for image segmentation'*, NAFIPS 2009.

[9]. .Mukherjee, *'MRF clustering for segmentation of color images'*, Pattern Recognition. Letters, 23(2002), pp 917-929.

[10]. C.Carson, S.Belongie, H.Greenspan, J.Malik, *'Blobworld: Image segmentation using Expectation maximization and its application to image querying'*, IEEE Trans. On Pattern Analysis and Machine Intelligence, vol.24, no.8, August 2002.

[11] D. Comaniciu, P.Meer, *'Mean-Shift: A robust approach towards feature space analysis'*, IEEE Trans. On Pattern Analysis and Machine Intell.,Vol. 24,no.5, pp 603-619, May 2002.

[12] J.Shi and J.Malik, *'Normalized Cuts and image segmentation'*, IEEE Trans. On Patern Analysis and Machine Intell.,Vol.22, no.8, pp 888-905, Aug. 2000.

[13] P.Felzenszwalb,.D.Huttenlocher, *'Efficient graph based image segmentation‖, International Journal of Computer Vision'*, Vol.59, no.2, pp. 167-181, Sept. 2004.

[14] R.Unnikrishnan, *'Toward objective evaluation of image segmentation algorithms'*, IEEE Trans. On Pattern Analysis and Machine Intell., vol. 29, no.6, June 2007.

[15] Uchiyama, Arbib, *'Color image segmentation using competitive learning'*, IEEE Trans. Pattern. Analysis and Mahcine Intell., Vol.16, no.12, pp. 1197-1206, 1994.

[16] Cheng,Jiang, Wang, *'Colour image segmentation :advances and prospects'*, Patt. Recog. , 34, pp.1277-1294, 2001.

[17] Vincent, Soille, *'Watersheds in digital spaces, an efficient algorithm based on immersion simulations'*, IEEE Trans. On Patt. Analysis and Machine Intell., vol.9, pp.735-744, 1991.

[18] Tilton, *'D-Dimensional formulation and implementation of recursive hierarchical segmentation'*, NASA, Case no. GSC 15199-1.

[19] Y.Deng, and B.S.Manjunath, *'Unsupervised segmentation of color texture regions in images and video'*, IEEE Trans. Patt. Anal. Mach. Intell., vol. 23(8), pp. 800-810, Aug. 2001.

[20] Wangenheim, *'Color image segmentation using an enhanced Gradient network method'*, Patt. Recog. Letters 30(2009), pp. 1404-1412.

[21] Q.Luo, Khoshgoftaar, *'Unsupervised multiscale color image segmentation based on MDL principle'*, IEEE Trans. On system, man, cyber., 15(9), pp.100-104, 2006.

[22] S.Ji, H.W.Park, *'Image segmentationof color image based on region coherency'*, Inter. Conf. on Image Procc., pp. 80-83, 1998.

[23] Lior Shamir, *'Human Perception based color segmentation using fuzzy logic'*, Intl. Conf. on Fuzzy sys.,2006, Vol. 2, pp. 496-505.

[24] Lim, Y.W., and Lee, S.U, *'On the colour image segmentation algorithm based on thresholding and the fuzzy c-means techniques'*, Pattern Recognit., 1990, 23, (3), pp. 935–952.

[25] H.D.Cheng, J.Li, *'Fuzzy homogeneity and scale space approach to color image segmentation'*, ICCV 2000.

[26] C.H.Oh, K.Honda, H.Ichihashi, *'Fuzzy clustering for categorical and multivariate data'*, Proc. of IFSA/NAFIPS, July 2001, Vol. 4, pp 2154-59.

[27] K.Kummamuru, A. Dhawale, R.Krishnapuram, *'Fuzzy co-clustering of Documents and Keywords'*, IEEE Conf. on Fuzzy Systems, May 2003, Vol.2, pp.772-777.

[28] W.C.Tjhi, L.Chen, *'Possibilistic fuzzy co-clustering of large document collections'*, Patt. Recog. 40(2007), pp. 3452-3466.

[29] W.C.Tjhi, L.Chen, *'Robust fuzzy co-clustering algorithm'*, IEEE Conf. ICICS 2007.

[30] M. Hanmandlua, O. P. Verma , Seba Susana, V.K. Madasu, *'Color segmentation by fuzzy co-clustering of chrominance color features'*, Neurocomputing 120 (2013) 235–249.

[31] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, *'Spectral–spatial classification of hyperspectral imagery based on partitional clustering techniques'*, IEEE Trans. Geosci. Remote Sens., vol. 47, no. 8, pp. 2973–2987, Aug. 2009.

[32] A. W. C. Liew, H. Yan, and N. F. Law, *'Image segmentation based on adaptive cluster prototype estimation'*, IEEE Trans. Fuzzy Syst., vol. 13, no. 4, pp. 444–453, Aug. 2005.

[33] P. Dulyakarn and Y. Rangsanseri, *'Fuzzy C-means clustering using spatial information with application to remote sensing'*, in Proc. 22nd Asian Conf. Remote Sens., Singapore, 2001.

[34] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag, and T. Moriarty, *'A modified fuzzy C-means algorithm for bias field estimation and segmentation of MRI data'*, IEEE Trans. Med. Imag., vol. 21, no. 3, pp. 193–199, Mar. 2002.

[35] S. Krinidis and V. Chatzis, *'A robust fuzzy local information C-means clustering algorithm'*, IEEE Trans. Image Process., vol. 19, no. 5, pp. 1328–1337, May 2010.

[36] Nan Li, Hong Huo, Yu-ming Zhao, Xi Chen, and Tao Fang, *'A Spatial Clustering Method With Edge Weighting for Image Segmentation',* IEEE Geoscience And Remote Sensing Letters, Vol. 10, No. 5, pp. 1124-1128 ,September 2013.

[37] Rastgarpour M., and Shanbehzadeh J., '*Application of AI Techniques in Medical Image Segmentation and Novel Categorization of Available Methods and Tools',* Proceedings of the

International MultiConference of Engineers and Computer Scientists 2011 Vol I, IMECS 2011, March 16-18, 2011, Hong Kong.

[38] Dinesh D. Patil, Sonal G. Deore, '*Medical Image Segmentation: A Review*',International Journal of Computer Science and Mobile Computing, IJCSMC, Vol. 2, Issue. 1, January 2013, pg.22 – 27.

[39] W. X. Kang, Q. Q. Yang, R. R. Liang, *'The Comparative Research on Image Segmentation Algorithms',* IEEE Conference on ETCS, pp. 703-707, 2009.

[40] L.Aurdal, *'Image Segmentation beyond thresholding',* Norsk Regnescentral, 2006.

[41] Wahba Marian, *'An Automated Modified Region Growing Technique for Prostate Segmentation in Trans-Rectal Ultrasound Images'*, Master's Thesis, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada, 2008.

[42] V. K. Dehariya, S. K. Shrivastava, R. C. Jain*, 'Clustering of Image Data Set Using K-Means and Fuzzy KMeans Algorithms'* , International conference on CICN, pp. 386- 391, 2010.

[43] Stephen M. Smith, J. Michael Brady, *'SUSAN—A New Approach to Low Level Image Processing'*, International Journal of Computer Vision, pp. 45–78, 1997.

[44] Ming-Ni Wu, Chia-Chen Lin,Chin-Chen Chang*, 'Brain Tumor Detection Using Color-Based K-Means Clustering Segmentation',* IEEE Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing(IIHMSP), vol.2, pp. 245-250, Nov.2007.

[45] Bezdek, J.C., '*Pattern Recognition with Fuzzy Objective Function Algorithms'*, New York: Plenum Press, 1981.