CHAPTER 4

FRACTAL TECHNIQUES IN IMAGE ANALYSIS
4.1 Introduction

Image analysis has an important role in many applications ranging from medical imaging to astronomy. The purpose of image analysis is to extract symbolic information from an image. Many of the image processing techniques which are valuable as visualization tools are useful at preprocessing stages in the image analysis task. In the simplest terms, image analysis involves mapping of a concrete image into an abstract symbolic representation. The ultimate goal of image analysis is the identification of a scene and all objects in the image. Image analysis can be described as a set of techniques required to extract symbolic information from the image data. It is the process of identifying and understanding patterns that are relevant to the performance of an image based task. It is different from other image processing applications like restoration, enhancement and coding where the output is another image. The image is processed in such a way that it removes all unwanted information retaining only that part of the image which contributes significantly to the analysis task. The analysis task creates a mapping from a digital image to the description of image's content. The image description can either be a number which would represent the number of objects in the image, or it can be a degree of anomaly which would define the shape variation of an object or it can be labelling of pixels to classify different regions of the image. Until early 90s, image processing was performed with specially designed and constructed areas of memory called framestores and the performance and speed were rather slow. But, now it is possible to download image analysis software from a number of websites and perform the task of analysis with less effort.

There are three levels in image analysis. In the low level processing, functions that may be viewed as automatic reactions are dealt where as intermediate level processing deals with the task of extracting regions in an image that results from a low level process. High level processing deals with recognition and interpretation tasks. Image analysis is performed using either bottom up approach or top down strategy. In bottom up approach, low level features are extracted from the raw image data and later, this is processed in higher levels. In top down approach, the image characteristics are
hypothesized at the highest level and is proceeded towards the lower level until the raw image has been reached (Gonzalez and Woods, 2000; Mantas, 1987). Image analysis involves the study of feature extraction, segmentation and classification (Jain, 1995).

In this chapter, some of the existing segmentation techniques like iterative thresholding, region growing by double thresholding, histogram technique based on fractals are applied on a set of solar images and test images (Jahne and Haubecker, 2000). A modified variation algorithm for computing fractal dimension is suggested and implemented on these images. The new algorithm when applied on a sequence of multiwavelength solar images gives a new method of analyzing the plages as well as their evolution. Also, an important feature namely, area of the plage region could be studied using this approach. This method of analysis could serve as a new way of studying the dynamics of solar images and its characteristics.

4.2 Applications of image analysis

Image analysis in the areas of medicine, biology or in movie production usually requires the processing of thousands of images. Therefore automatization is crucial in these applications. Much of the development work in image analysis has been done in the medical field since medical images contain a wide spectrum of information. Examples of medical images are those obtained using X-rays, magnetic resonance, ultrasound, cine-angiograms etc. In recent years, the design of computer-based display mechanisms (large video screens, head-mounted displays, high definition screens) which can perform better compared to the conventional display methods such as X-ray images on film or Computerised Tomography images attracted the attention of a number of researchers.

Another potential field of application where image analysis is performed is astronomical image processing. Vision models have been developed for the automated identification of the astrophysical sources and their relevant measurements depending on the image content. Several image analysis tools are available in astronomy, of which the most commonly used is the Graphical Astronomy and Image Analysis Tool (GAIA).
GAIA provides the features like highly interactive environment for controlling the positions, sizes and orientations of circular and elliptical apertures, ability to automatically detect and parameterise all the objects on an image, identification of extended objects (galaxies) and profile measurements using ellipse fitting, ability to select arbitrary shaped regions on an image and replace them with a surface fit to other regions, image statistics, contouring of the displayed image, ability to display vector maps produced by the POLPACK package etc (Draper et al., 2001). Developments are still progressing in the area of astronomical image analysis since most of the software available are not sufficient in computing large variety of digital sky surveys.

4.3 Segmentation Techniques

Segmentation is the process of identifying regions of pixels in an image. The main goal is to divide an image into parts that have a strong correlation with objects or areas in the real world. There are three classes of segmentation procedures; global knowledge, edge based and region based. Global knowledge methods rely on knowing something about the image such as the expected pixel intensity. Edge based methods find the borders between regions. Region based methods use different properties of the image to define regions such as grey level, color, texture, shape etc.

Over the past thirty years or so, classification and segmentation has been studied using human and vision perspectives. The three decades have witnessed a slow but steady evolution of techniques ranging from co-occurrence matrices to Markov random field. The main aim is to recognise homogeneous regions within an image as distinct and belonging to different objects. The segmentation process can be based on finding the maximum homogeneity in grey levels within the regions identified. There are several issues related to image segmentation. One of the common problems encountered in image segmentation is the choosing of a suitable approach for isolating different objects from the background. The segmentation doesn't perform well if the grey levels of different objects are quite similar. Image enhancement techniques seek to improve the visual appearance of an image. They emphasize the salient features of the original image and
simplify the task of image segmentation. The type of operator chosen has a direct impact on the quality of the resultant image. It is expected that an ideal operator will enhance the boundary differences between the objects and their background making the image segmentation task easier. Issues related to segmentation involve choosing good segmentation algorithms, measuring their performance, and understanding their impact on the scene analysis system.

A segmented image consists of two regions namely homogeneous region and transition region. The most difficult task in a vision system is to identify the sub images that represent objects. Region detection though appears simple for human is not an easy task for computers. The partitioning of images into sub images is what is done in segmentation. In other words, segmentation is grouping of pixels into regions \((R_i)\) such that

\[
\begin{align*}
(i) & \quad \bigcup_{i=1}^{k} R_i = \text{Entire image} \\
(ii) & \quad R_i \cap R_j = \emptyset, \quad i \neq j \\
(iii) & \quad \text{The pixels belonging to region } R_i \text{ possess some common characteristics.} \\
(iv) & \quad \text{Pixels belonging to adjacent regions possess different characteristics.}
\end{align*}
\]

There are different techniques for finding object regions in grey-level images. They are histogram thresholding, edge detection, tree/graph based approach, region growing, clustering.

4.3.1 Histogram Thresholding

In histogram thresholding, the picture is thresholded at its most clearly separated peak. The process iterates for each segmented part of the image until no separate peaks are found in any of the histograms. The criteria to separate peaks was based on the ratio of peak maximum to peak minimum to be greater than or equal to two.
4.3.2 Edge Detection

Edge detection provides an automatic way of finding boundaries of one or more objects in an image. From an image containing many objects edge detection allows us to single out a particular object of interest. Edge detection is used in many applications. In edge based segmentation, pixel neighbourhood elements are used for image segmentation. For each pixel, its neighbours are first identified in a window of fixed size. A vector of these neighbours as individual grey values or vector of average grey levels in windows of size $1 \times 1$, $3 \times 3$ and $5 \times 5$ is determined. Then a weight matrix is defined which when multiplied with these vectors will yield a discriminant value that allows the classification of pixel in one of the several classes (Cheriet et al., 1998).

4.3.3 Tree/Graph based approach

In Tree/graph based approaches segmentation is derived from the consensus of a set of different segmentation outputs on one input image. Instead of statistics characterising the spatial structure of the local neighbourhood of a pixel, for every pair of adjacent pixels their collected statistics are used for determining local homogeneity. Several initial segmentations are derived from the same input image by changing the probabilistic component of the hierarchical Region Adjacency Graph (RAG) pyramid based technique. From the ensemble of these initial segmentations, for every adjacent pixel pair, a co-occurrence probability is derived, which captures global information (about the image) at the local level (pixel level). The final segmentation of the input image is obtained by processing the co-occurrence probability field with the same RAG pyramid technique. The pixel pairs with high co-occurrence probability are then grouped together, based on the consensus about local homogeneity.
4.3.4 Region Growing

Another segmentation approach is based on region growing. Region growing algorithms take one or more pixels, called seeds, and grow the regions around them based upon a certain homogeneity criteria. If the adjoining pixels are similar to the seed, they are merged with them within a single region. The process continues until all the pixels in the image are assigned to one or more regions.

4.3.5 Clustering

Image segmentation can be performed effectively by clustering image pixels. Cluster analysis allows the partitioning of data into meaningful subgroups and it can be applied for image segmentation or classification purposes. Clustering analysis either requires the user to provide the seeds for the regions to be segmented or uses non-parametric methods for finding the salient regions without the need for seed points. Clustering is commonly used in a range of applications such as image segmentation and unsupervised learning (Jain and Dubes, 1988). A number of issues related to clustering are worth studying including how many clusters are the best and how to determine the validity of clusters. In a number of segmentation techniques, such as fuzzy c-means clustering, the number of clusters present in the image have to be specified in advance.

4.3.6 Neural Networks Segmentation

Campbell et al. (1997) proposed an automatic segmentation and classification method for outdoor images using neural networks. In their work, the images are segmented using Self-Organising Feature Maps (SOFM) based on texture and colour information of the objects. SOFMs used consisted of 64 x 64 nodes for best segmentation. A set of 28 features is then extracted from each region. The features include: average colour, position, size, rotation, texture (Gabor filters) and shape (using principal components). Classification is then performed using a Multi Layer Perceptron with 28 input nodes and 11 output nodes. The training is performed on 7000 regions and testing is done on an independent set of 3000 samples. Over 80% regions were classified
correctly using Learning Vector Quantisation and 91.9% regions were classified correctly using the Multi Layer Perceptron. (Sharma, 2001)

4.4 Fractal Methods

Fractal techniques have been extensively used to characterize data in the fields of medicine, atmospheric physics, geophysics, astrophysics etc. Kasparis et al. (2001) describe a new approach to the segmentation of textured gray-scale images using fractal features. The technique used is the variation method followed by filtering of the image to generate the feature set. Dimensionality reduction is applied to reduce the feature set and clustering is performed to obtain the segmented regions. Chaudhuri and Sarkar (1995) have defined a technique to compute fractal dimension of images known as differential box counting method. This technique has proved to be successful in classifying textures by means of unsupervised k-means clustering. Osman et al. (1998) have conducted fractal based image analysis of human trabecular bone. The relationship of BCD to standard measures of trabecular bone is also analyzed. Klonowski (2000) has demonstrated the usefulness of fractal analysis as a tool to understand the structural information from digitized images of medical science and engineering. He also describes how fractal models may be used for image segmentation, texture classification, estimation of 3D roughness from image data. In their work, Barbara and Chen (2000) designed a fractal clustering algorithm that places points incrementally in the cluster for which the change in fractal dimension after adding the point is the least. Vuduc (1997) introduced a local fractal operator with his algorithm, known as Beavor’s method which is a modified form of Sarkar’s method to find the fractal dimension of a single pixel. The technique is used for image segmentation. However, the quantisation step (histogram partitioning) is not automated and this is a major drawback of the work. Marchette et al. (1997) have employed fractal based techniques in digital mammography to detect tumorous tissues. The tumor regions were better identified when segmentation boundaries were incorporated into the calculation. Zwiggelaar and Bull (1994) have conducted fractal analysis to study recognition of plant images. Fractal techniques are widely used in astronomy, particularly solar physics to identify fractal clusters in sunspot penumbras and solar active regions. Fractal techniques can be applied to identify
temporal changes of the Hurst coefficient by applying Hurst's modified "Range over Standard Deviation" analysis to a finite set of data taken at discrete intervals. The evolution of Hurst coefficients from active regions are compared to determine the correlation with solar flares. Chapman et al. (2001) has studied the ratio of facular area to sunspot area. The identification of faculae on continuum images and their maximum contrasts has been studied by Watson and Preminger (2000). New features which would help in the modeling of the total solar irradiance were introduced by Oritz et al. (2002).

4.4.1 Computation of pixelwise fractal dimension

Images are segmented using fractal approach by computing the local fractal operator followed by pixelwise classification. The different methods of computing pixelwise fractal dimension are Simple thresholding, Sarkar's method, Beavor's method, Variation method and the New method.

4.4.1.1 Simple thresholding

This method is a direct implementation of the box counting method applied locally to each pixel. For a pixel \((i, j)\), a small window, \(W\) of size \(m \times m\) surrounding it is considered. The fractal dimension corresponding to the pixel is the slope obtained by fitting a line of \(\log N(r)\) versus \(\log r\) where \(N(r)\) is the number of nonempty boxes and \(r\) is the size of the box.

4.4.1.2 Sarkar's method

In Sarkar's method the fractal dimension corresponding to the pixel is the slope obtained by fitting a line of \(\log N(r)\) versus \(\log (r)\). \(N(r)\) is the summation of \(n(r)\) for various grid sizes, \(r\) that surrounds the pixel and \(n(r)\) is the difference between the bin numbers corresponding to the maximum and minimum intensity values present in grid size, \(r\).
4.4.1.3 Beavor's method

Beavor's method is essentially the same as that of Sarkar's method except that the intensities of the grid are rescaled locally before proceeding with the box counting method. The fractal dimension corresponding to the pixel is the slope obtained by fitting a line of \( \log N(r) \) versus \( \log r \). \( N(r) \) is the summation of \( n(r) \) for various grid sizes, \( r \) that surrounds the pixel. \( n(r) \) is the difference between the bin numbers corresponding to the maximum and minimum intensity values present in grid size, \( r \).

4.4.1.4 Variation method

In variation method, the fractal dimension corresponding to the pixel is the slope obtained by fitting a line of \( \log (R\epsilon) \) versus \( \log (R\epsilon)^3 E_\epsilon \) for \( \epsilon = 1,2,3.. \epsilon_{\text{max}} \) and \( R \) is the size of the window. \( E_\epsilon \) is the average of \( V_\epsilon \) over the window \( R \) and \( V_\epsilon \) is the difference between the maximum and minimum intensity values computed over a window of size \( T \) that surrounds the pixel, where

\[
T = 2\epsilon + 1
\]

for \( \epsilon = 1,2,3.. \epsilon_{\text{max}} \). ………..(4.1)

4.4.1.5 New algorithm proposed

This method is a modified form of variation method. In the new method, the fractal dimension corresponding to the pixel is the slope obtained by fitting a line of \( \log (R\epsilon) \) versus \( \log (R\epsilon)^3 E_\epsilon \) for \( \epsilon = 1,2,3.. \epsilon_{\text{max}} \) and \( R \) is the size of the window. \( E_\epsilon \) is the average of \( V_\epsilon \) over the window \( R \) and \( V_\epsilon \) is the average of the intensity values computed over a window of size \( T \) that surrounds the pixel, where \( T \) is same as in equation 4.1, for \( \epsilon = 1,2,3.. \epsilon_{\text{max}} \). This method proved to be better in segmenting certain images.
4.5 Algorithms

The algorithms used have two divisions, one consisting of the computation of pixel-wise fractal dimension and the other, for segmentation.

4.5.1 Algorithms used for the computation of pixelwise fractal dimension

4.5.1.1 Sarkar’s method

The fractal dimension of a pixel \( x(i, j) \) is computed by considering window \( W \) and following the steps given in section 4.4.1.2

1. For various \( r \in (0,1] \) do
   (i) Divide \( W \) into \((1/r)^{2}\) squares.
   (ii) Divide the intensities from 0-225 into \( 1/r \) levels or bins numbered 1\ldots 1/r
   (iii) For each grid cell
   a. Let \( b_1 \) denote the bin number corresponding to the maximum intensity and \( b_2 \) denote the bin number corresponding to the minimum intensity.
   b. Let \( n_{p,q}(r) = b_1 - b_2 + 1 \)
   (iv) \( N(r) = \sum n_{p,q}(r) \)

2. Do a line fit of \( \log(N(r)) \alpha \log(r) \).

4.5.1.2 Variation method

Following the description given in section 4.4.1.4, the algorithm may be written as

1. for each pixel in image \( I(i,j) \)
   for \( e = 1, 2, 3, \ldots, e_{\text{max}} \)
   construct a window \( T = 2e + 1 \)
compute $M_e = \text{maximum}(T)$
compute $N_e = \text{minimum}(T)$
compute $D_e = M_e - N_e$
end for

2. for each pixel in image I (i,j)

construct a window of suitable size, $R$.
for $e = 1, 2, 3, \ldots, e_{\text{max}}$

$E_e = \text{average}(D_e)$
end for
Do a line fit of $(\log(R/e), \log((R/e)^3E_e))$, where $R$ is a window of suitable size.
end for

4.5.1.3 New method

This algorithm follows the method given in section 4.4.1.5

1. for each pixel in image I(i,j)

for $e = 1, 2, 3, \ldots, e_{\text{max}}$

construct a window $T = 2e + 1$
compute $D_e = \text{average}(T)$
end for
end for

2. for each pixel in image I (i,j)

construct a window of suitable size, $R$.
for $e = 1, 2, 3, \ldots, e_{\text{max}}$

$E_e = \text{average}(D_e)$
end for
Do a line fit of $(\log(R/e), \log((R/e)^3E_e))$, where $R$ is a window of suitable size.
end for
4.5.2 Algorithms used for segmentation

The different segmentation algorithms implemented (Jain et al., 1995) in the present study are iterative thresholding, region growing and histogram based thresholding.

4.5.2.1 Iterative thresholding

1. Select an initial estimate of the threshold, T.
2. Partition the image into two groups, \( R_1 \) and \( R_2 \), using the threshold T.
3. Calculate the mean gray level \( \mu_1 \) and \( \mu_2 \) of the partitions \( R_1 \) and \( R_2 \).
4. Select a new threshold
   \[ T = \frac{1}{2} (\mu_1 + \mu_2) \]
5. Repeat steps 2-4 until the mean values \( \mu_1 \) and \( \mu_2 \) in successive iterations do not change.

Here the average of the pixelwise fractal dimensions is taken as the initial estimate for T.

4.5.2.2 Region Growing

1. Select two thresholds \( T_1 \) and \( T_2 \).
2. Partition the image into three regions \( R_1, R_2 \) and \( R_3 \). \( R_1 \) contains all pixels with grey values below \( T_1 \). \( R_2 \) contains pixels with grey values between \( T_1 \) and \( T_2 \). \( R_3 \) contains pixels with gray values above \( T_2 \).
3. Each pixel in \( R_2 \) is examined. If it has a neighbor in region in \( R_1 \), then reassign the pixel to region \( R_1 \).
4. Repeat step 3 until no pixels are reassigned.
5. Reassign any pixels left in region \( R_2 \) to region \( R_3 \).
4.5.2.3 Histogram based segmentation

1. Partition the image into different blocks.
2. Find \( fd \) for each block
3. Plot histogram for the fractal dimension obtained above
4. Assign the peak value as the threshold to segment the image.

Altogether, nine different algorithms are implemented on a set of images that comprise of solar images and test images. The solar images were those maintained by Bear Solar Observatory and SOHO and the test images were prepared by Jahne and Haubecker (2000).

4.6 Data Set

The above methods were applied to a set of test images and a sequence of solar images. The automatic identification of solar features, such as faculae and plages, is becoming increasingly important as the resolution and the size of solar data sets increases. The introduction of space borne solar telescopes, in addition to the ground based observations have increased the solar image data set many fold. The identification of solar features are required for the quantitative study of the solar activity, which includes locations, lifetimes, contrasts, and other characteristics of sunspots and faculae etc and the modeling of the total solar irradiance and variations of sunspot and facular properties with latitude and/or solar cycle phase.

A portion of the chromospheric image from Bear Solar Observatory (http://www.bbso.njit.edu) obtained on April 1, 2001 is chosen for study (figure 4.2 (a)). The images are recorded on photographic film and later digitized. The detection of plage regions from chromospheric image is done and is suggested as a means of understanding the shape of plages and their evolution.

The solar images selected are gray scale images of size 226 x 251 with intensity variation from 0 to 255. Solar and heliospheric observatory (SOHO) continuously monitors solar atmosphere using Extreme ultra violet Imaging Telescope (EIT) at four wavelengths 171 A\(^0\), 195 A\(^0\), 284 A\(^0\) and 304 A\(^0\), shown as blue, green, yellow and red.
respectively. The solar images taken at different wavelengths provide information about the features at different altitude regions of the solar atmosphere and their time evolution (http://sohowww.nascom.nasa.gov). In this work, we have tried to identify the bright regions in the solar atmosphere from the solar images taken at different wavelengths. By analyzing the bright regions, we can get an idea about the geometry of the surface magnetic field at different altitude region of the Sun and also its time evolution.

At present, to determine the plage areas, one has to either apply a threshold or manually surround the plages with polygons. The thresholding method ignores the spatial information contained in the image. The second method uses a large amount of information and is highly subjective (Turmon and Mukhtar, 1997). In this work, we propose a fractal based approach for detecting plage regions. We chose this approach since it could be observed that the plage and non-plage regions possess some kind of self similarity which appeared to be highly homogeneous. Applying our new technique, the fractal dimension and plage areas at different wavelengths have been evaluated for the SOHO satellite images (Figures 4.5, 4.7, 4.9).

### 4.7 Application of edge detectors

There are different approaches for the segmentation of images. The most common and simple technique is by means of edge detection. Different edge detection techniques like, Canny, Prewitts were applied to the images of the plage region. The results are given in Figure 4.1 b and 4.1 c. In all cases, the edge detectors could not identify the plage region satisfactorily.

![Figure 4.1: (a) is the original image (b) edge detected by Prewitts (c) edge detected by Canny edge detector](image-url)
It is known that the plage region and the non-plage regions possess a texture-like structure which motivated to apply local fractal operator to the segmentation problem. Hence, a fractal-based approach was followed which could segment the image into two regions, thus detecting plage from the rest of the component.

4.8 Results and Discussions

The existing methods, namely Sarkar's method and variation method as well as the modified algorithm for evaluating pixelwise fractal dimension have been applied on the image given in Figure 4.2a and on a set of test images (Figure 4.12a – Figure 4.12c). The results are shown in figure 4.2 – 4.4.

Using Sarkar's algorithm to compute fractal dimension corresponding to each pixel, the segmentation proved to be satisfactory. However, it could not segment the plage regions into solid blocks. The segmentation results are shown in the following figures 4.2b – 4.2d.

Using variation algorithm to compute fractal dimension corresponding to each pixel, the segmentation proved to be better than the above technique. It could differentiate the plage region from the network region to a certain extent. The segmentation results are shown in the following figures 4.3a – 4.3c.

Using new algorithm to compute fractal dimension corresponding to each pixel, the segmentation holds good for this application. It could differentiate the plage region from the network region quite well. The segmentation results are shown in the following figures 4.4a – 4.4c.

The modified algorithm proposed by us for obtaining the fractal dimension proved to give better results for test images and solar images and the plage regions could be identified more elegantly. The results are depicted in figures 4.4, 4.13, 4.14, 4.15. Using the proposed method, after detecting the plage region, one can study the property
of the region. One of the important properties of plage regions, namely plage area is evaluated for a sequence of images (Figures 4.5 - 4.10). This has important implications on the evolution of many characteristics of multi wavelength solar images. The variation of fractal dimension with time for different images is given in Figure 4.11. Also, it could be observed that the plage areas vary in an identical manner with fractal dimension (Tables 4.1-4.3). The method of image analysis implemented in this chapter could serve as an alternative to the existing methods in the study of dynamics of solar images and their features.

Figure 4.2 a: Original image

![Original Image](image)

Figure 4.2 b: Iterative thresholding (Sarkar’s method)

![Iterative Thresholding](image)

Figure 4.2 c: Region growing (Sarkar’s method)

![Region Growing](image)

Figure 4.2 d: Histogram based thresholding (Sarkar’s method)

![Histogram Based Thresholding](image)
It could be observed that Sarkar’s algorithm could detect the plage region to a little extent. With iterative thresholding and histogram based approach, the background region overlaps with the plage region. With region growing, the segmentation result is improved.

Fig 4.3 a: Iterative thresholding (Variation method)

Fig 4.3 b: Region growing (Variation method)

Figure 4.3 c: Histogram based thresholding (Variation method)

Figure 4.4 a: Iterative thresholding (New method)

The variation algorithm gave better results compared to Sarkar’s algorithm. However, it failed in detecting the whole plage region.
With the new algorithm, the plage regions were completely detected from the background. With iterative thresholding, the result is similar as that of the other algorithms but with region growing and histogram approach, the results hold good.
Table 4.1: Area of plage region detected for SOHO images taken at 304 Å

<table>
<thead>
<tr>
<th>Time</th>
<th>Fractal dimension</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>01:09 (28.11.02)</td>
<td>1.33</td>
<td>717</td>
</tr>
<tr>
<td>07:09</td>
<td>1.31</td>
<td>680</td>
</tr>
<tr>
<td>13:19</td>
<td>1.28</td>
<td>658</td>
</tr>
<tr>
<td>19:19</td>
<td>1.34</td>
<td>799</td>
</tr>
<tr>
<td>01:19 (29.11.02)</td>
<td>1.31</td>
<td>582</td>
</tr>
<tr>
<td>07:19</td>
<td>1.36</td>
<td>870</td>
</tr>
<tr>
<td>13:20</td>
<td>1.32</td>
<td>556</td>
</tr>
<tr>
<td>19:19</td>
<td>1.35</td>
<td>618</td>
</tr>
</tbody>
</table>

Table 4.1: Area of plage region detected for SOHO images taken at 304 Å
Table 4.2: Area of plage region detected for SOHO images taken at 171 Å

<table>
<thead>
<tr>
<th>Time</th>
<th>Fractal dimension</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>01:00 (28.11.02)</td>
<td>1.17</td>
<td>540</td>
</tr>
<tr>
<td>07:00</td>
<td>1.14</td>
<td>494</td>
</tr>
<tr>
<td>13:00</td>
<td>1.13</td>
<td>471</td>
</tr>
<tr>
<td>19:00</td>
<td>1.13</td>
<td>464</td>
</tr>
<tr>
<td>01:00 (29.11.02)</td>
<td>1.16</td>
<td>453</td>
</tr>
<tr>
<td>07:00</td>
<td>1.11</td>
<td>359</td>
</tr>
<tr>
<td>13:00</td>
<td>1.13</td>
<td>343</td>
</tr>
<tr>
<td>19:00</td>
<td>1.10</td>
<td>320</td>
</tr>
</tbody>
</table>

Table 4.3: Area of plage region detected for SOHO images taken at 284 Å

<table>
<thead>
<tr>
<th>Time</th>
<th>Fractal dimension</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>01:06 (28.11.02)</td>
<td>1.50</td>
<td>599</td>
</tr>
<tr>
<td>07:06</td>
<td>1.52</td>
<td>540</td>
</tr>
<tr>
<td>13:06</td>
<td>1.525</td>
<td>511</td>
</tr>
<tr>
<td>19:06</td>
<td>1.525</td>
<td>476</td>
</tr>
<tr>
<td>01:06 (29.11.02)</td>
<td>1.552</td>
<td>468</td>
</tr>
<tr>
<td>07:06</td>
<td>1.558</td>
<td>485</td>
</tr>
<tr>
<td>13:06</td>
<td>1.555</td>
<td>475</td>
</tr>
<tr>
<td>19:06</td>
<td>1.558</td>
<td>456</td>
</tr>
</tbody>
</table>

Figure 4.11: Variation of fractal dimension with time for different set of images
Figure 4.12: Original images of (a) face (b) glasgow (c) bonsai

Figure 4.13: Segmentation obtained for ‘face’ by (a) histogram technique (b) region growing (c) iterative thresholding

Figure 4.14: Segmentation obtained for ‘glasgow’ by (a) histogram technique (b) region growing (c) iterative thresholding
Figure 4.15: Segmentation obtained for 'bonsai' by (a) histogram technique (b) region growing (c) iterative thresholding

For comparison, Sarkar's algorithm was applied to the above images and the results are as follows. It could be observed that the edges are broken in all cases. However, the image 'bonsai' was found to give better results.

Figure 4.16: Segmentation results for face by (a) histogram technique (b) region growing (c) iterative thresholding

Figure 4.17: Segmentation results for glasgow by (a) histogram technique (b) region growing (c) iterative thresholding
The same images were tested using variation algorithm and the results are as follows. The variation method detected edges better than Sarkars algorithm. The image 'bonsai' was segmented to give the mesh properly, but could not extract tree from its background.

Figure 4.18: Segmentation results for bonsai by (a) histogram technique (b) region growing (c) iterative thresholding

Figure 4.19: Segmentation results for face by (a) histogram technique (b) region growing (c) iterative thresholding

Figure 4.20: Segmentation results for glasgow by (a) histogram technique (b) region growing (c) iterative thresholding
In this chapter, segmentation of images using local fractal operator is discussed. The local fractal operator is computed using different algorithms like Sarkar's algorithm and variation algorithm. A new algorithm, which is a modification of variation algorithm is proposed. The complexity of this algorithm is $O(mn)$, where $n$ is the number of pixels in the image and $m$ is the number of windows considered. Since $m$ is very small compared to $n$, the complexity can be taken as $O(n)$. Algorithms used for segmentation purpose are iterative thresholding, region growing and histogram based thresholding. The techniques could successfully detect active regions from solar images.