4.1 Edge Based Segmentation

In this section two different types of edge based segmentation is discussed, which are Marr-Hildreth edge detection and Canny based edge detection.

4.1.1 Laplacian of Gaussian

Edge-based segmentation relies on discontinuities in the image data to locate the boundaries of the segments before assessing the enclosed region. The problem with this approach is that in practice the edge-profile is usually not known. Furthermore, the profile often varies heavily along the edge caused by, for example, shading and texture. Due to these difficulties usually a symmetrical simple step-edge is assumed and the edge detection is performed based on a maximum intensity gradient [68]. However, the resulting boundary is seldom complete and so edge linking is usually necessary to fill gaps. Region edges which are complete may also be defined by the zero crossing of the Laplacian operator which provides a 2-D isotropic measure of the second spatial derivative of an image. The Laplacian of an image has largest magnitudes at peaks of intensity and has zero crossings at the points of inflection on an edge. Two common $3 \times 3$ convolution kernels used to calculate the digital Laplacian are as follows:

\[
\begin{bmatrix}
0 & 1 & 0 \\
1 & 4 & 1 \\
0 & 1 & 0 \\
\end{bmatrix} \quad \begin{bmatrix}
1 & 1 & 1 \\
1 & 8 & 1 \\
1 & 1 & 1 \\
\end{bmatrix}
\]

Fig 4.1 The Convolution kernels of Laplacian
Because it is sensitive to noise, the Laplacian is often applied to an image that has first been smoothed with an approximation of a Gaussian smoothing filter and since convolution is a commutative operation, the two techniques are often combined as the LOG or Marr-Hildreth operator. Marr-Hildreth edge detection [68] is based on locating the zero-crossings of the LOG operator applied to the image using various values for the standard deviation of the Gaussian. LOG methods incorporate noise reduction and have potential for rugged performance [69].

However, the computation of the zero crossings is complicated in general and although zero crossing positions are correct for ideal edges, errors as large as the standard deviation of the Gaussian can occur in other cases. In practice, the zero crossing detected edges often include many small closed loops because a threshold, if applied, is very small and the weak edges at the closed boundaries of many small minor regions are detected.

Fig 4.2 Marr/Hildreth edge detection of Solar Images
A Laplacian based approach called the inflexion Point Method (IPM) was proposed [70] for measuring the umbral and penumbral areas of sunspots. High-resolution sunspot images obtained with the 50 cm Swedish Vacuum Solar Telescope at the Observatorio del Roque de los Muchachos at La Palma were used. The images were captured using a Kodak Megaplus 8-bit CCD camera and filters centred at the wavelengths of 525.7 nm and 542.5 nm. Starting with a rectangular region (possibly smoothed) which contains a sunspot umbra or whole sunspot, a map of contours is obtained by applying a 7 x 7 Laplacian operator and then searching for its zero crossings. A mean intensity, $I_u$, along the contour positions is computed and the isoline at this intensity level is taken as the umbra-penumbra boundary. The same method is used to obtain the penumbra-photosphere boundary. The steps involved in the implementation of the algorithm are as follows:

(i) To define the area of a sunspot umbra, a rectangular region containing only the umbra and a portion as small as possible of a surrounding penumbra is manually chosen for extraction from the whole frame. If necessary the selected regions is smoothed to reduce the effect of perturbing features prior to computation of the derivatives.

(ii) The inflexion points in the spatial distribution of intensity in the box are located by computing the second derivative Laplacian operator and by searching for its zero crossing points, thus producing a map of contours.

(iii) To compute the mean intensity along a given contour, their coded algorithm requires that the contour be closed and have no spurious internal contours. If necessary, a contour is closed and spurious internal contours removed manually.
by adding or deleting pixels in an interactive way directly on the displayed contour image, using as a guide both the smoothed image and the original image.

(iv) The mean intensity in the image over the path defined by the zero crossing contour is computed. The isoline at this intensity level is taken to be the definitive boundary of the umbra.

(v) The isoline at the mean intensity level may include small contours inside the umbra that nevertheless belong physically to the umbra. To include such small regions into the umbral area, an operation is necessary to interactively remove their contours from inside the umbral boundary. The value of the umbral area is calculated by summing up all pixels enclosed by the umbral boundary.

Following a similar scheme for the whole sunspot, both the umbra-penumbra boundary and the total spot area can be determined using the IPM. The authors assert that methods for measuring sunspot areas based on fixed intensity thresholds cannot be justified because of the different characteristics of individual sunspots. In comparison with the difference and cumulative histogram based methods for measuring sunspot areas, the IPM is much less affected by image blurring due to the seeing conditions and more in accord with the physical structure of sunspots. However, the method, as described in reference [70] is not fully autonomous.

4.1.2 Canny Method

The Canny edge-detection method [37] also aims to find the edges of objects where the intensity changes rapidly and is optimal in a precise mathematical sense but more complex to implement than methods based on Prewitt or Sobel operators. Combined with hysterisis tracking, which requires two thresholds, the Canny detector
attempts to follow boundaries between poorly defined objects as well as hard edges and ignores weak edges which are not connected to stronger edges. The Canny based edge detection procedure involves the following steps:

- Smooth the image with a Gaussian filter
- Compute the gradient magnitude and orientation using finite-difference approximations for the partial derivatives,
- Apply non-maxima suppression, with the aid of the orientation image, to thin the gradient-magnitude edge image,
- Track along edges starting from any point exceeding a higher threshold as long as the edge point exceeds the lower threshold.
- Apply edge linking to fill small gaps.

The design of edge detectors for arbitrary edge profiles. The design was based on the specification of detection and localization criteria in a mathematical form. It was necessary to augment the original two criteria with a multiple response measure in order to fully capture the intuition of good detection. A mathematical form for the criteria was presented, and numerical optimization was used to find optimal operators for roof and ridge edges. The analysis was then restricted to consideration of optimal operators for step edges. The result was a class of operators related by spatial scaling. There was a direct trade off in detection performance versus localization, and this was determined by the spatial width. The impulse response of the optimal step edge operator was shown to approximate the first derivative of a Gaussian. A detector was proposed which used adaptive thresholding with hysteresis to eliminate streaking of edge contours. The thresholds were set according to the
amount of noise in the image, as determined by a noise estimation scheme. This detector made use of several operator widths to cope with varying image signal-to-noise ratios, and operator outputs were combined using a method called feature synthesis, where the responses of the smaller operators were used to predict the large operator responses. If the actual large operator outputs differ significantly from the predicted values, new edge points are marked. It is therefore possible to describe edges that occur at different scales, even if they are spatially coincident.

Fig 4.3 Canny Edge Detected Solar Images

4.2 Nature Inspired Algorithms

The nature inspired algorithms work based on the principles of nature. If it contains biological factors then its characteristic contain the feature of particular organism. Nature-inspired algorithms often use multiple interacting agents. A subset of meta-heuristics are often referred to as Swarm Intelligence (SI) based algorithms, and these SI-based algorithms have been developed by mimicking the so-called swarm intelligence characteristics of biological agents such as birds, fish, humans and others. For example, particle swarm optimization was based on the swarming behavior of birds and fish, while the firefly algorithm was based on the flashing
pattern of tropical fireflies, and cuckoo search algorithm was inspired by the brood parasitism of some cuckoo species [75].

Nature has been evolving for several hundred million years, and she has found various ingenious solutions to problem-solving and adaption to ever-changing environments. From Darwinian evolution point of view, survival of the fittest will result in the variations and success of species, which can survive and optimally adapt to environments, and thus selection is a constant pressure that drives the system to improve and adapt for survival. Any evolutionary advantages over competitors may increase the possibility of reproduction and success of the individuals and the species over the long run. We can learn from nature by mimicking the successful characteristics of complex systems in nature. Nature-inspired algorithms are still at a very early stage with a relatively short history, comparing with many traditional, well-established methods; however, nature-inspire algorithms have already shown their great potential, flexibility and efficiency with ever-increasing diverse ranges of applications.

Despite the increasing popularity of meta-heuristics, many crucially important questions remain unanswered. There are two important issues: theoretical framework and the gap between theory and applications. At the moment, the practice of meta-heuristics is like heuristic itself, to some extent, by ‘trial and error’. Mathematical analysis lags far behind, apart from a few, limited, studies on convergence analysis and stability; there is no theoretical framework for analyzing meta-heuristic algorithms. I believe mathematical and statistical methods using Markov chains and dynamical systems can be very useful in the future work. There is no doubt that any theoretical progress will provide potentially huge insightful into meta-heuristic algorithms [76].
4.3 Ant Algorithms

Ants are eusocial insects in habit and they live together in organized colonies consist of millions of ants. Ants form colonies that range in size from a few dozen predatory individuals living in small natural cavities to highly organized colonies that may occupy large territories and consist of millions of individuals. Larger colonies consist mostly of sterile, wingless females forming castes of "workers", "soldiers", or other specialized groups. Nearly all ant colonies also have some fertile males called "drones" and one or more fertile females called "queens". The colonies sometimes are described as super organisms because the ants appear to operate as a unified entity, collectively working together to support the colony.

When foraging, a swarm of ants or mobile agents interact or communicate in their local environment. Ants follow scent trails laid by scout ants to gather food. By following pheromone trails created by other ants from the colony, foraging ants can gather and store food efficiently. A scout ant first leaves the nest in search of food, and wanders somewhat randomly until it discovers something edible. It will then consume some of the food and return to the nest in a straight, direct line. It seems these scout ants can observe and recall visual cues that enable them to navigate quickly back to the nest. Along the return route, the scout ant leaves a trail of pheromones, special scents that will guide her nest mates to the food. The foraging ants then follow her path, each one adding more scent to the trail to reinforce it for others. The workers will continue walking back and forth along the line until the food source is depleted. From the initial random foraging route with higher pheromone concentration varies and the ants follow the route with higher pheromone concentration and the pheromone is enhanced by the increasing number of ants. More
number of ants follow the same path, it become the favored path, which is also shortest path.

The ant colonies consist of simple interaction among individual ants. Individual ant act based on the local information to carry out their activities. There no master or overseeing the entire colony and broadcasting instructions to the individual ants, organized behavior still emerges automatically. The foraging pattern of army ants species can show extraordinary regularity. Army ants search for food along some regular routes with a specified angle. On the first day, they forage in a random direction for example north and travel a few hundred meters then branch to cover a large area. The next day they will choose a different direction, which are about 123 degree from the direction of pervious day and cover a larger area. On the third day they again choose a angle 123 degree from the direction of second day direction. Finally ants over the whole area in 2 weeks and they move out to a different location to build a bivonac and forage in the new region [78].

4.4 Ant Colony Optimization

Based on the features on ant behavior, different ant colony algorithm is developed. Marco Dorigo pioneered the ant colony research from 1992 and different variations also evolved. The issues in ant colony optimization discussed here are the probability of choosing a route and the evaporation rate of pheromone. ACO aims to iteratively find the optimal solution of the target problem through a guided search over the solution space, by constructing the pheromone information. To be more specific, suppose totally K ants are applied to find the optimal solution in a space S that consists of M × N nodes, the procedure of ACO can be defined as steps below

1. Initialize the positions of totally K ants, as well as the pheromone matrix $P^{(0)}$.  

81
2. For the construction-step index \( n = 1 : N \), – For the ant index \( k = 1 : K \),

3. Repeatedly move the \( k \)-th ant for \( L \) steps, according to a probabilistic transition matrix \( PT^{(n)} \).

4. Update the pheromone matrix \( P^{(n)} \).

5. Make the solution decision according to the final pheromone matrix \( P^{(N)} \).

ACO is a population based met heuristic approach to find approximate solutions to difficult optimization problems. The inspiring source of ACO is the pheromone trail laying behavior of real ants, which use pheromone as a communication medium [77]. In analogy to the biological example, ACO is modeled based on the indirect communication of a colony of simple agents, called artificial ants, mediated by artificial pheromone trails. These pheromone trail values are modified at runtime based on a problem-dependent heuristic function and the amount of pheromone deposited by the ants while they traverse between their colonies and a food source. The problem-dependent heuristic function, in the case of famous ACO algorithms for travelling salesman problem, is set to be the inverse of the distance between one city and another city [78]. In ACO, pheromone trail values serve as distributed, numerical information, which the ants use to construct solutions probabilistically. There is one solution per ant. The higher the pheromone value, the higher the probability of an ant choosing that particular trail will be. The pheromone values on lower quality trails which are not reinforced often enough will progressively evaporate. The pheromone based evaporation implements a useful form of forgetting: it avoids the algorithm from converging too rapidly toward a suboptimal region; therefore, as mentioned above, it is repeatedly applied until a termination condition is satisfied. In practice, a termination condition may be the maximum number of solutions generated, the
maximum CPU time elapsed, or the maximum number of iterations without improvement in solution of favoring the exploration of new areas in the search space [77].

There are two fundamental issues in the above ACO process; that is, the establishment of the probabilistic transition matrix $p(n)$ and the update of the pheromone matrix $\tau(n)$, each of which is presented in detail as follow, respectively.

For a network routing problem the probability of ants at a particular node $i$ to choose the route from node $i$ to node $j$ is given by

$$P_{ij} = \frac{\phi_{ij}^\alpha d_{ij}^\beta}{\sum_{n=1}^N \phi_{ij}^n d_{ij}^n} \quad [4.1]$$

$\alpha, \beta>0$ influence parameter $\approx 2$

$\phi_{ij}$ is the pheromone concentration, $d_{ij}$ desirability of the same route

$d_{ij} \propto \frac{1}{s_{ij}}$, shorter routes will be selected due to their shorter travelling time a priori knowledge about the route such as the distances $s_{ij}$. Due to shorter travelling time and thus the pheromone concentrations on these routes are higher.

The travelling time is shorter and thus the less amount of the pheromone has been evaporated during this period. This probability formula reflects the facts that ants would normally follow the paths with higher pheromone concentrations.

$\alpha = \beta = 1$, the probability of choosing a path by ants is proportional to the pheromone concentration on the path. Local optimal solution problem can be reduced due to evaporation of pheromone. For a constant rate $\gamma$ of pheromone decay or evaporation, the pheromone concentration usually varies with time exponentially.

$$\phi(t) = \phi(0)e^{-\gamma t} \quad [4.2]$$

$\phi(0)$ Initial concentration of pheromone

t time
The increment amount of pheromone deposited at time $t$ along route $i$ to $j$, when an ant travels a distance $L$. The ants with the current global best solutions are allowed to deposit pheromone.

![Original solar images and segmented sunspots using ACO](image)

**Fig 4.4 (a)–(c) Original solar images and (d)–(h) Represent segmented sunspots using ACO**

### 4.5 Region growing methods

It is a simple region based image segmentation method. The main goal of segmentation is to partition an image into regions. Region based segmentation is a technique for determining the region directly. Seeded region growing method takes a set of seeds as input along with the image. The seeds mark each of the objects to be segmented. The regions are iteratively grown by comparing all unallocated neighbouring pixels to the regions. The difference between a pixel’s intensity value and the region mean is used as a measure of similarity. The pixel with the smallest
difference measured this way is allocated to the respective region. This process continues until all pixels are allocated to a region [53].

Seeded region growing requires seeds as additional input. The segmentation results are dependent on the choice of seeds. Noise in the image can cause the seeds to be poorly placed. Unseeded region growing is a modified algorithm that does not require explicit seeds. It starts off with a single region, the pixel chosen here does not significantly influence final segmentation. At each iteration considers the neighbouring pixels in the same way as seeded region growing [54].

It differs from seeded region growing in that if the minimum region means is less than a predefined threshold T then it is added to the respective region. If not, then the pixel is considered significantly different from all current regions and new region is created with this pixel. The selection of seed points is based on preferences and image. The connectivity or pixel adjacent information is helpful for us to determine the threshold and seed points. The major drawback of region growing is the power and time consuming. The aim of region based segmentation techniques is to extract the homogeneous zones from the image. Region growing techniques is generally better in noisy images, where borders are extremely difficult to detect. For region growing method homogeneity is an important property, which can be based on gray-level, shape, model etc.

In this method seeded region growing approach is used which segment the image into the homogeneous regions with respect to a set of seed points. The selection of the similarity criteria for region growing

\[
\alpha (i,j) - \beta (i,j) \propto (m,n) \propto \alpha (i,j) + \beta (i,j)
\]

[4.3]

where \( \beta (i,j) \) is the statistical similarity bound.

\[
\beta (i,j) = a + be^{-c\alpha(i,j)}
\]

[4.4]
and a, b, c are the coefficients with values that are estimated empirically. The local statistics \( \alpha(i, j) \) are used as the quantitative measure to obtain a homogeneous region for each image pixel.

### 4.5.1 Seeded region growing method (SRG)

The input of seeds, either individual pixels or regions which will control the formation of regions into which the image will be segmented. Seeded region growing performs a segmentation of image with respect to set of points, known as seeds [71]. We start with a number of seeds which have been grouped into n sets, say A1, A2, ...... An

Sometimes, individual sets will consist of single points. It is in the choice of seeds that the decision of what is a feature of interest and what is irrelevant or noise is embedded. The algorithm finds a division of image into regions with property that each connected component of a region meets. Exactly one of the A, and the regions are chosen to be as homogeneous as possible.

T set of all as yet unallocated pixels which border at least one of the regions x pixels.

\( \delta(x) \) A measure of how different x is from the region it adjoins.

Sequentially stored list (SSL) is a linked list of objects (pixel addresses) which are ordered according to some attribute.

**Algorithm**

1. Label seed points according their initial grouping
2. Put neighbors of seed points in the SSL
   
   While the SSL is not empty:
   
   Remove first point y from ssl. Test the neighbours of this point
If all neighbors of y which are already labeled (other than with the boundary label) have the same label- set y to this label.

Update running mean of corresponding region.

Add neighbors of y which are neither already

Set nor already in the SSL to the SSL according to their values of $\delta$

otherwise Flag y with the boundary label.

Region growing approaches exploit the important fact that pixels which are close together have similar gray values. Region growing starts with a single pixel (seed) add new pixels slowly or multiple seed

• Choose the seed pixel
• Check the neighbouring pixels and add them to the region if they are similar to the seed.
• Repeat step 2 for each of the newly added pixels; stop if no more pixels can be added.
• The algorithm uses 8 neighbors predicate

4.5.1.1 Seed Selection

• It depends on the nature of the problem
• If targets need to be detected using infrared images for example, choose the brightest pixels
• without a priori knowledge, compute the histogram and choose the gray-level values corresponding to the strongest peaks
• Similarity criteria (Predicates)
• The homogeneity predicate can be based on any characteristic of the regions in the image such as average intensity, variance, colour, texture, motion, shape, size
1. A simple approach to image segmentation is to start from the seeds (pixels) representing distinct image regions and to grow them, until they cover the entire image.

2. For region growing we need a rule describing a growth mechanism and a rule checking the homogeneity of the regions after each growth step.

3. The growth mechanism at each stage $K$ and for each region $R_i(k)$, $i = 1, 2, \ldots, N$. We check if there are unclassified pixels in the 8-neighbourhood of each pixel of the region border.

4. Before assigning each a pixel $x$ to a region $R_i(k)$, we check if the region homogeneity $P (R_i(k) \cup \{x\}) = TRUE$, is valid.

5. The arithmetic mean $m$ and standard deviation $sd$ of a class $R_i$ having $n$

$$M = \left( \frac{1}{n} \right) \sum_{(r, c) \in R_i} I(r, c)$$

Homogeneity test

$$|I(r, c) - M| \leq T_i$$  \[4.5\]

Disadvantage is that all pixels must be assigned to region not only ROI

---

Fig 4.5(a) – (c) Original solar images and (d) – (i) Represent segmented sunspots using RGM
4.5.2 Unseeded region growing method (URGM)

The Unseeded region growing is preferred due to the known issues of seeded region growing algorithms. A good segmentation result depends on a set “correct” choice for the seeds. When the input images are noisy, the seeds may fall on typical pixels that are not representative of the region statistics. This can lead to erroneous segmentation results. Unseeded region based segmentation is similar to seeded except that no explicit seed selection is done [72].

Formally the segmentation process initializes with region $A$, containing a single image pixel, and the running state of the segmentation process consist of a set of all unallocated pixels which borders at least one of these regions.

$$T = \{ x \in \bigcup_{i=1}^{n} A_i \land \exists k : N(x) \cap A_k \neq \emptyset \}$$

[4.6]

$N(x)$ are immediate neighbouring pixels of point $x$

$g(x)$ denotes the image value at point $x$

Growing process involves selecting a point $Z \in T$ and region $A_{i=1}^{n}$, $Z$ is added to $A_j$ only if it is less than the predefined threshold $t$. Otherwise a new region is formed.

The most important issue for consideration is the threshold, which is virtual to the success of the segmentation process. Threshold is closely correlated with contrast in the image, it is better to analyze contrast.

4.6 Swarm based region growing model (SBRG)

The region based segmentation is a method for shaping the region directly. The first step in region growing is to select a set of seed points. Seed point selection is based on some user criterion for example, pixels in a certain gray-level range, pixels evenly spaced on a grid, etc. The initial region begins as the exact location of these seeds. The regions are then grown from these seed points to adjacent points depending
on a region membership criterion. The criterion could be, for example, pixel intensity, gray level texture, or color. Since the regions are grown on the basis of the criterion, the image information itself is important. For example, if the criterion were a pixel intensity threshold value, knowledge of the histogram of the image would be of use, as one could use it to determine a suitable threshold value for the region membership criterion. Different parameters should be defined for proper working of method like suitable selection of seed points and similarity threshold. The similarity threshold can be identified by the knowing content information about the image. Here the sunspot images captured earlier will provide the information regarding similarity threshold value. The sunspot will occur randomly on the solar image. The seed points can be considered as points nearby the sunspot or within sunspot. Due to the random nature of sunspot automatic selection of seed is required which further require knowledge about the solar image. To retrieve basic information from solar image preprocessing is done.

The Ant colony optimization (ACO) method is used to detect the boundary of sunspot as a standalone technique for segmentation. The ACO output can be considered as an initial map, which helps us to initiate seed within the sunspot image. ACO is a population based swarm approach to find approximate solutions to difficult optimization problems. The inspiring source of ACO is the pheromone trail laying behavior of real ants, which use pheromone as a communication medium. In analogy to the biological example, ACO is modeled based on the indirect communication of a colony of simple agents, called artificial ants, mediated by artificial pheromone trails. These pheromone trail values are modified at runtime based on a problem-dependent heuristic function and the amount of pheromone deposited by the ants while they traverse between their colony and a food source. The problem-dependent heuristic
function, in the case of famous ACO algorithms for travelling salesman problem, is set to be the inverse of the distance between one city and another city. In ACO, pheromone trail values serve as distributed, numerical information, which the ants use to construct solutions probabilistically. There is one solution per ant. The higher the pheromone value (initial edge), the higher the probability of an ant choosing that particular trail will be. The pheromone values on lower quality trails which are not reinforced often enough will progressively evaporate. The Pheromone based evaporation implements a useful form of forgetting: it avoids the algorithm from converging too rapidly toward a suboptimal region, therefore mentioned above are repeatedly applied until a termination condition is satisfied. In practice, a termination condition may be the maximum number of solutions generated, the maximum CPU time elapsed, or the maximum number of iterations without improvement in solution re favoring the exploration of new areas in the search space. The Swarm based region growing (SRG) method uses ACO for seed selection for region growing method to identify the sunspot. The region growing methods can correctly separate the regions that have the same properties defined by us. The initial preprocessing will help us to decide the numbers of seed point to be initiated in an image.

In swarm based region growing the seeded method is used in region growing model. The criterion for seed selection can be considered as pixel intensity, gray level, texture, colour and histogram for finding threshold. The swarm based region growing segmentation is a method for shaping and extracting the region of interest accurately. The seeded region growing method is used here to detect sunspot from a solar image. Usually seed point selection is based on some user criterion, but here we used the swarm based method ant colony optimization edge map is created. The ant colony
optimization edge is considered as the seed point information. The similarity threshold can be identified by the knowing content information about the image. The sunspot images captured earlier will provide the information regarding similarity threshold value. The sunspot will occur randomly on the solar image on different time sequences. The seed points can be considered as points nearby the sunspot or within sunspot. The ant colony optimization methods are used to detect the boundary of sunspot as standalone techniques for segmentation. The ACO output can be considered as an initial map, which helps us to initiate seed within the sunspot image.

Ant colony optimization is a population based swarm approach to find approximate solutions to difficult optimization problems. A termination condition may be used to stop the initial edge map. In practice, a termination condition may be the maximum number of solutions generated the maximum number of iterations without improvement in solution. The initial pre-processing will help us to decide the number of seed points to be initiated in a solar image.

Fig 4.6 (a) – (c) Original solar images and (d) – (g) Represent segmented sunspots using SRGM
4.7 Web weaving model (WWM)

All In the ideal case, the result of applying an edge detector to a solar image may lead to a set of closed loops that indicate the boundaries of sunspots. Using an edge sensor to a solar image may significantly cut the quantity of information to be processed and may therefore filter out information that may be viewed as less relevant, while conserving the important structural properties of a solar image. If the edge detector detects the basic structure in a successful manner, the subsequent task of optimizing the edges from an edge based image may therefore be substantially simplified. Edges obtained from solar images often vulnerable by noises and edges are not in a closed loop structure, missing edge segments. The false edges in the solar image are mainly caused by the noise in the image making difficult the subsequent task of classifying the solar image. Noise in solar images is mostly shot noise that comes from the quantum nature of light and an uncertainty in the number of electrons emitted by a photo detector. Shot noise is a type of electronic noise which originates from the discrete nature of electric charge. The term also applies to photon counting in optical devices, where shot noise is associated with the particle nature of light [23].

Different segmentation methods are analyzed for detecting the sunspot in the solar image. The proposed method describes an initial segmentation by the classical methods like modified canny [60]. Because the image data is already processed with the median filter to remove noise and other artifacts, the Gaussian filtering process is not done as an initial step. An edge in an image may point in a variety of directions, so the canny algorithm uses four filters to detect horizontal, vertical and diagonal edges in the blurred image. The Canny algorithm basically finds edges where the grayscale intensity of the image changes the most. These areas are found by determining gradients of the image. Gradients at each pixel in the image are
determined by applying operator. First step is to approximate the gradient in the x- and y-direction respectively by applying the two kernels by convolution operation.

\[
\begin{pmatrix}
-2 & 0 & 2 \\
-3 & 0 & 3 \\
-2 & 0 & 2 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
2 & 3 & 2 \\
0 & 0 & 0 \\
-2 & -3 & -2 \\
\end{pmatrix}
\]

Then the edge strengths or gradient magnitude

\[
G = \sqrt{G_x^2 + G_y^2}
\]

and the angle of the gradient \( \theta = \arctan \left( \frac{|G_y|}{|G_x|} \right) \) [4.7]

Compute \( \theta \) by rounding the angle to one of eight directions \( 0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ \) and obviously for edges, \( 180^\circ = 0^\circ, 202.5^\circ = 22.5^\circ, 225^\circ = 45^\circ \) etc. The non-maximal suppression step keeps only those pixels on an edge with the highest gradient magnitude. These maximal magnitudes should occur right at the edge boundary, and the gradient magnitude should fall off with distance from the edge. So, three pixels in a \( 9 \times 9 \) around pixel \((x, y)\) are examined:

- If \( \theta \'(x, y) = 0^\circ \), then the pixels \((x + 4, y), (x, y), \) and \((x - 4, y)\) are examined.
- If \( \theta \'(x, y) = 22.5^\circ \), then the pixels \((x - 4, y - 2), (x, y), \) and \((x + 4, y + 2)\) are examined.
- If \( \theta \'(x, y) = 90^\circ \), then the pixels \((x, y + 4), (x, y), \) and \((x, y - 4)\) are examined.
- If \( \theta \'(x, y) = 45^\circ \), then the pixels \((x - 4, y - 4), (x, y), \) and \((x + 4, y + 4)\) are examined.
- If \( \theta \'(x, y) = 67.5^\circ \), then the pixels \((x - 2, y - 4), (x, y), \) and \((x + 2, y + 4)\) are examined.
- If \( \theta \'(x, y) = 112.5^\circ \), then the pixels \((x - 2, y + 4), (x, y), \) and \((x - 2, y - 2)\) are examined.
- If \( \theta \'(x, y) = 135^\circ \), then the pixels \((x - 4, y + 4), (x, y), \) and \((x + 4, y - 4)\) are examined.
If pixel (x, y) has the highest gradient magnitude of the three pixels examined, it is kept as an edge. If one of the other two pixels has a higher gradient magnitude, then pixel (x, y) is not on the “center” of the edge and should not be classified as an edge pixel. Some of the edges detected will not actually be valid, but will just be any artifacts, which requires filtering. A single threshold may actually remove valid parts of a connected edge, leaving a disconnected final edge image. This happens in regions where the edge’s gradient magnitude fluctuates between just above and just below the threshold in the case of group of sunspot dispersed over an area in solar image. In hysteresis thresholding two different thresholds are used and one threshold will be used as low and another one will be high threshold. The pixels above the high threshold are classified as sunspot and below the low threshold as solar image background. The pixels value between the low and high threshold are classified as sunspot only if they are adjacent to other sunspot pixels.

- If pixel (x, y) has gradient magnitude less than $t_{\text{low}}$ discard the edge.
- If pixel (x, y) has gradient magnitude greater than $t_{\text{high}}$ keep the edge.
- If pixel (x, y) has gradient magnitude between $t_{\text{low}}$ and $t_{\text{high}}$ and any of its neighbors in a $9 \times 9$ region around it have gradient magnitudes greater than $t_{\text{high}}$, keep the edge.

Now we got an initial segmented sunspot image map from the modified classical approach. The initial result may lack the connectedness in the edges, to overcome this issue we discuss how disconnected edges are connected using filling algorithms from bio-inspired solutions like spider web weaving algorithm [76]. The natural life is a source of divine guidance for the conception of novel image segmentation algorithms.
If these algorithms are independently used the performance lacks, because these algorithms require an initial knowledge about the environment it is used. The modified classical approach provides the initial knowledge here to further enhance the segmentation quality. The spider construction of web using silk weaving is considered and initial segmentation map is considered as the attraction for silk. The spider is always situated on top of the web. The web weaving activity needs two behavior, one is the movement of spider in the different direction and the second one is silk fixing. These activities can be done simultaneously by the spider. Here the program will imitate the spider and the edge map will create silk attraction. When disconnected edges will be weaved by the spider. The possibility to fix the silk and the movement is constant over time. More than one spider can be spread over an sunspot detected in the edge map and whenever two spiders meet the movement is terminated. The orientation and direction of movement by the spider is done based on the 9 X 9 region and edge gradient map. The movement of the spider can be expressed as a stigmergic process. Stigmergy is a mechanism of indirect coordination between agents or actions. The principle is that the trace left in the environment by an action stimulates the performance of a next action, by the same or a different agent. In that way, subsequent actions tend to reinforce and build on each other, leading to the spontaneous emergence of coherent, apparently systematic activity.

Spider web weaving algorithm

Step 1: Move forward the spider agents across the edge map

Step 2: If disconnected path appears the spider agents move forward 9 X 9 regions based on orientation and direction followed and check the condition for connectedness.

Step 3: If condition match go to step 4 otherwise stop the spider agent.
Step 4: Move backward and web weaving is done to connect the disconnected edges.

Step 5: If two spider agents collide in opposite direction the spider agents is stopped.

Fig 4.7 (a) – (c) Original solar images and (d) – (i) Represent segmented sunspots using WWM

4.8 Classification of Sunspots

Sunspot observation, analysis and classification form an important part in furthering knowledge about the sun, the solar weather and its effect on earth. Certain categories of sunspot groups are associated with solar flares. Observation around the world track all visible sunspots in an effort detect flares at an early stage of their formation. Sunspot recognition and classification are currently manual and labour intensive processes which could be automated if successfully learned by a machine. Some initial attempts at automated sunspot recognition and classification were presented [55].
Several learning algorithms were examined to investigate the ability of machine learning in dealing with the problem of sunspot classification. The experiment showed that it is very difficult to learn the classification scheme using only visual properties as attribute. Using different segmentation technique different characteristics of sunspots can be determined. To improve the classification accuracy experiments were performed with classification learning in combination with clustering and layered learning methods.

A sunspot consists of one or more dark cores, called umbrae often surrounded by a less dark area called penumbra. In the umbrae, very intense, longitudinally oriented magnetic fields cause the photosphere gases to become very cool, and thus dark compared to overall photosphere. Sunspots have a tendency to appear in magnetically bi-polar groups. In each group there are normally two major spots, oriented approximately east-west, called the leading, preceding or western, and the following or eastern spot.

The leading spot is usually larger in size and has stronger magnetic field strength. It is first to form, first to develop penumbra, and last to dissipate. Also the leading spot is often located slightly closer to the equator than the following spot. The sunspot exhibit “proper motion” due to the growth and expansion of the magnetic regions, and differential solar rotation. The polar region of the sun rotates slower than the equatorial regions. Once a sunspot has reached its maximum longitudinal extent, it usually stabilizes or starts to decay as the magnetic field weakens. Sunspot within a region will sometimes move relative to each other or the major spot may rotate about an axis. The number of spots in a sunspot group is the number of umbrae visible. Two umbrae surrounded by the same penumbral area count as two spots.
4.9 Classification Scheme

The sunspot appear on the solar disk as individual spots or as a group of spots. Sunspot groups can have an infinite variety of formations and sizes, ranging from small solo spots to giant groups of spots with complex structures [56].

Using the McIntosh sunspot classification scheme spots are classified according to three descriptive codes.

First code which consist of seven broad categories:

A: Unipolar group with no penumbra, at start or end of spot group’s life

B: Bipolar group with penumbras on any spots

C: Bipolar group with penumbras on one end of group, usually surrounding largest of leader umbrae.

D: Bipolar group with penumbras on spots at both ends of group, and with longitudinal extent less than 10 arc seconds (120000 km)

E: Bipolar group with penumbras on spots at both ends of group, and with longitudinal extent between 10 and 15 arc seconds

F: Bipolar group with penumbras on spots at both ends of group, and length more than 15 arc seconds (above 180000 km)

H: Unipolar group with penumbra

Principal spot is usually the remnant leader spot of pre-existing bipolar groups

Second code describes
The penumbra of the largest spot of the group.

Third code describes

The compactness of the spots in the intermediate part of the group. Solar flares are usually associated with large groups.

4.10 Data Preparation

The features extracted by the previous method were shape descriptors, describing the shape of single sunspots and information about spot’s neighbours. The following sunspot features were extracted x and y coordinates of a spot center, area of a spot, perimeter length around a spot, spot’s angle to the main axis, spot’s aspect ratio, compactness and form factor, spot’s feret’s diameter, spot’s circularity, the count of how many neighbouring sunspots are within a specified radii. To clarify individual sunspots using layered learning concept is used. To improve the classification accuracy the domain knowledge is added into the learning process.

Given hierarchical concept decomposition, the main idea is to synthesize a target concept gradually from simpler ones. A learning process is performed through the hierarchy, from leaves to the root, layer by layer. The solution to the sunspot classification problem using layered learning consists of four main steps:

1. Recognize single sunspots using image processing techniques and create decision table describing their classification made by experts.

2. Group daily sunspots into clusters and create decision table for those clusters

3. Create a hierarchical learning method based on rough set theory to learn the zurich sunspot classification scheme
Above classification can be described by simpler concepts using

- Magnetic type of groups (unipolar, bipolar)

- Group span is a heliographical distance of two farthest spots in a group (i.e., three spanning degrees)

  Null

  Small (less than 10 h.degs/ 120000 km)

  Large (more than 15 h.degs/ 180000 km)

  Middle (between 10 h.degs & 15 h.degs)

- Penumbra type of the leading spot and different types are

  No Penumbra, Rudimentary, Asymmetric, Symmetric

- Penumbra size of the leading spot and it consist of two possible values

  Small (less than 2.5 h.degs or 30000 km)

  Large (more than 2.5 h.degs)

- Distribution of spots inside a group and four possible values are

  Single, Open, Intermediate, Compact

If we consider all situations described by those concepts, there will be many different scenarios. Every possibility is characterized by those concepts and can be labelled by one of seven letters {A, B, C, D, E, F, H} according to the Zurich classification scheme. Based on the decision tree observations a new classification is defined.
- Classes D, E and F are similar on almost all attributes except attribute group span.

- Classes A, H have similar magnetic type (both are unipolar) but they are discerned by the attribute penumbra type.

- Classes B, C have similar magnetic type (both are bipolar) but they are discerned by the attribute penumbra size.

The classification study can be done by the help of the neural or fuzzy classifier for better understanding of the sunspots [65]. It can be done as the further extension of this work and weather predication software can be created. The proposed method segmented images are analyzed using performance measures in the next chapter.

**4.11 Summary**

In this chapter initially different edge based segmentation algorithm and nature based algorithms is analyzed. In nature based algorithm ant colony optimized algorithm is used for segmentation. Then region growing method with seeded and unseeded is discussed. From the analysis two different segmentation algorithms is evolved. First algorithm is swarm based region growing algorithm which uses the ant colony optimization for the identifying the optimized seeds and initiate the region growing algorithm. Second algorithm uses modified directions in the canny edge detection and creates an initial map. Using the initial map a web weaving algorithm creates segmentation. Finally the scheme and data preparation in the classification of sunspot groups defined by different standards is discussed.