© 2016 IJSRSET | Volume 2 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Ant Colony Optimization Algorithm for Improving Efficiency of Canny Edge Detection Technique for Images

Prof. Divyanshu Rao, Sapna Rai

Shri Ram Institute of Technology, Jabalpur, Madhya Pradesh, India

ABSTRACT

Edge detection is one of the important parts of image processing. It is essentially involved in the pre-processing stage of image analysis and computer vision. It generally detects the contour of an image and thus provides important details about an image. So, it reduces the content to process for the high-level processing tasks like object recognition and image segmentation. The most important step in the edge detection based on Canny edge detection algorithm, on which the success of generation of true edge map depends, lies on the determination of threshold. In this work, purpose of edge detection, inspired from Ant Colonies, is fulfilled by Ant Colony Optimization (ACO). The success of the work done is tested visually with the help of test images and empirically tested on the basis of several statistical parameter of comparison. The process of extracting the important features present in an image, keeping the unnecessary or unimportant information present in the form of noise out as much as possible. There are many methods that have been developed in these field, but the most trustworthy and used among them is canny algorithm with ACO method with thresholding. The proposed novel method presented in this thesis is tested on the images better edge detection. The Canny Edge detected images obtained on the images are showing better results than the other conventional edge detectors.

Keywords : Ant Colony Optimization (ACO), Edge Detection, Canny Edge Detection, BER, Thresholding, Statistical evaluation.

I. INTRODUCTION

Ant colonies, besides the fact that they are simple and small by nature, are distributed system that is able to perform a highly structured social organisation. This happens as they are capable to perform many complex tasks which is far exceeding the individual capabilities of single ant.The ant algorithms take basic feature from real world ants and are helpful in the design of novel algorithms for the development of optimization and distributed systems [1]. The self-organising principles of the real world ants which are the basis of the highly coordinated behaviour can be researched further to develop some algorithms related to computational problems. Some of these features are foraging, division of labour, brood sorting and cooperative transport. The underlying nature behind all these activities is a form of indirect communication [2] known as stigmergy, which happens because of modification of the environment. What is happening here is that the foraging ants deposit some type of chemical on the ground and other ants because of this increasing probability follow the same path. Researchers have tried to implement this stigmergy [3] in the artificial ants to coordinate the societies of artificial ants [1].

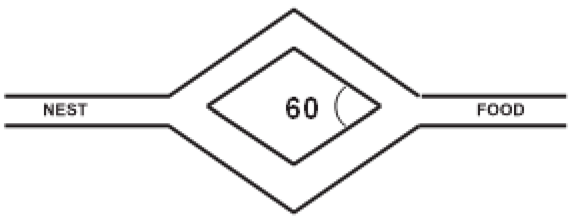
Foraging Behavior of Ants

Through some biologist's point of view, it is quite known that the visual sensory organs of the real world ants are rudimentary by nature and in some cases they are completely blind. A in depth research in the ants behaviour shows that the large part of communication individual or between individuals by ants is quite done by the use of chemicals produced by the ants, known as pheromone. Foraging behavior of ant species is also based on the indirect communication possibly done by pheromones. While having a walking from the food sources to the nest or vice-versa, the ants are depositing pheromone on the ground, forming in this way, a pheromone trail [3]. By sensing the path for any possible pheromone concentration, they choose paths probabilistically in the favor of any strong pheromone's concentration [1].

II. METHODS AND MATERIAL

1. Related Work

Edge detection refers to the process of extracting edges from the image where there are sudden changes or discontinuities. These extracted edge points from an image provides an insight into the important details in the field of image analysis and machine vision [10]. It acts as a pre-processing step for feature extraction and object recognition [9]. Various techniques are reported in the literature like Sobel [9], Prewitt [3], and Canny [5] detection techniques. However, most of the existing detection techniques use a huge search space for the image edge detection [7]. Therefore, without optimization the edge detection task is memory and time consuming. What is happening in this equal length. Here it is observed that in the initial phase, both paths are having same number of ants and after that one path gets more number of ants than the other. The reason behind this phenomenon is that ants at the very outset select both paths equally. But after some time due to random nature one path gets more preference than the other. As ants are leaving pheromone trails behind, so the path selected by the more number of ants gets more amount of pheromone which further reinforces the selection of that path. This nature of natural phenomenon is in another terms can be described as auto-catalytic or feedback process [1]. This is also explaining the stigmergy i.e., the indirect mode of communication happened due to the modifications in the environment.



Second experiment leads to the possibility that the one path is double the longer than the other path [8]. At the very start, selection is equal for both paths but after some time what is happening is that ants are selecting the path shorter among the two. The main reason behind is that ants choosing the shorter path are coming back from to their nest from food sources quickly. What is affecting their decision is that the shorter path is containing more amount of pheromone, so lead to the selection of that path more due to the auto-catalytic process, as described earlier. The effect of randomness is greatly reduced here and stigmergy, auto-catalysis, and differential path length are coming into action. Despite the fact that the shorter path is present, still some ants choose the longer path due to the path exploration [1].

A third case is also studied where a shorter path is added after a long time , and what happened here is that the ants are still attached with longer path due to the auto-catalytic nature and slow evaporation of pheromone trails.

III.PROPOSED WORK AND RESULTS

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's intentions were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "*A Computational Approach to Edge Detection*". In his paper, he followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be NO responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge. This was implemented because the first 2 were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (no maximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a nonedge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2.

Step1

In order to implement the canny edge detector algorithm, a series of steps must be followed. The first step is to filter out any noise in the original image before trying to locate and detect any edges. And because the Gaussian filter can be computed using a simple mask, it is used exclusively in the Canny algorithm. Once a suitable mask has been calculated, the Gaussian smoothing can be performed using standard convolution methods. A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. **The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise**. The localization error in the detected edges also increases slightly as the Gaussian width is increased. The Gaussian mask used in my implementation is shown below.

$\frac{1}{115}$

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Figure 3 Discrete approximation to Gaussian function with σ=1.4

Step 2

After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image. The Sobel operator performs a 2-D spatial gradient measurement on an image. Then, the approximate absolute grad-******ient magnitude (edge strength) at each point can be found. The Sobel operator uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown below:

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

The magnitude, or edge strength, of the gradient is then approximated using the formula:

$$|G| = |Gx| + |Gy|$$

Step 3

The direction of the edge is computed using the gradient in the x and y directions. However, an error will be generated when sumX is equal to zero. So in the code there has to be a restriction set whenever this takes place. Whenever the gradient in the x direction is equal to zero, the edge direction has to be equal to 90 degrees or 0 degrees, depending on what the value of the gradient in the y-direction is equal to. If GY has a value of zero, the edge direction will equal 0 degrees. Otherwise the edge direction will equal 90 degrees. The formula for finding the edge direction is just:

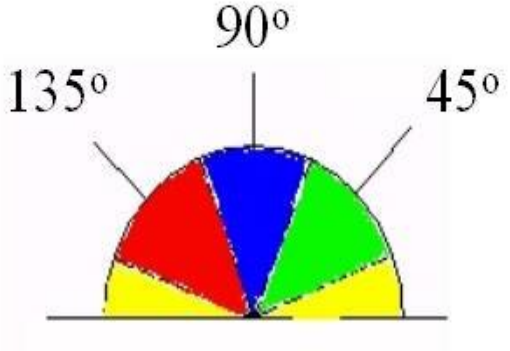
Theta = invtan (Gy / Gx)

Step 4

Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image. So if the pixels of a 5x5 image are aligned as follows:

x x x x x
x x x x x
x x a x x
x x x x x
x x x x x

Then, it can be seen by looking at pixel "**a**", there are only four possible directions when describing the surrounding pixels - **0 degrees** (in the horizontal direction), **45 degrees** (along the positive diagonal), **90 degrees** (in the vertical direction), or **135 degrees** (along the negative diagonal). So now the edge orientation has to be resolved into one of these four directions depending on which direction it is closest to (e.g. if the orientation angle is found to be 3 degrees, make it zero degrees). Think of this as taking a semicircle and dividing it into 5 regions.



Therefore, any edge direction falling within the **yellow range** (0 to 22.5 & 157.5 to 180 degrees) is set to 0 degrees. Any edge direction falling in the **green range** (22.5 to 67.5 degrees) is set to 45 degrees. Any edge direction falling in the **blue range** (67.5 to 112.5 degrees) is set to 90 degrees. And finally, any edge direction falling within the **red range** (112.5 to 157.5 degrees) is set to 135 degrees.

Step 5

After the edge directions are known, nonmaximum suppression now has to be applied. Nonmaximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.

Step 6

Finally, hysteresis is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold. If a single threshold, T1 is applied to an image, and an edge has an average strength equal to T1, then due to noise, there will be instances where the edge dips below the threshold. Equally it will also extend above the threshold making an edge look like a dashed line. To avoid this, hysteresis uses 2 thresholds, a high and a low. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T2 to start but you don't stop till you hit a gradient below T1.

and after that we will apply ACO algorithm ,As expected, using Canny or any other traditional edge detector as input to ACO will give better results and especially it can remove discontinuities in final result. Also due to the fact that result is dependent on input which itself can differ based on application, overall process can be considered adaptable.

After the completion of the above step what we have done is to implement the updated algorithm. In this step we will implement the improved algorithm using MATLAB and then comparing the present algorithm with the altered one.

At last we will do the result analysis and look after the differences in them with the help of graphs and charts.

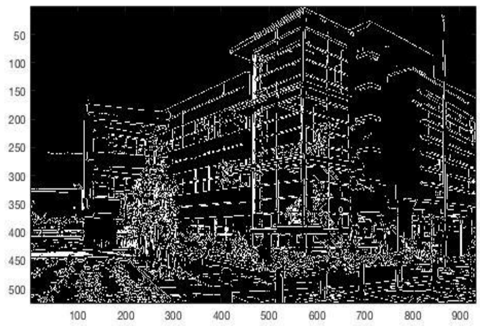
1. Study of the Existing Algorithms Identify the Different methods.
2. Identify the issue and Apply improved technique of Edge Detection using ACO.
3. ObtainEdged.
4. Performance measure of method by calculating
5. Performance parameters such as Mean, standard Deviation and BER.

III. RESULTS AND DISCUSSION

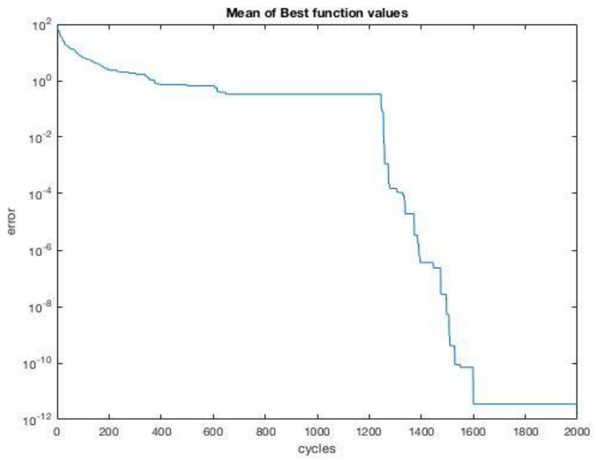
Simulation Results :

(a)this image is used as a input to Canny Edge detection algorithm for generating Edges in output.

(b) this image represents the output of Canny Edge detection algorithm in this edges of input image are shown with respect to pixels in X axis and Y axis.



(c) This graph shows the performance of Ant Colony Optimization Algorithm after taking output of Edge detection algorithm as a input and this graph represent mean and standard deviation for generated edges of image.



IV. CONCLUSION

Edge detection process is an important part of image processing. It is beneficial for many research areas of computer vision and image segmentation. Edge detection provides important details for the high-level processing tasks like feature detection etc. The success of edge detection depends on the optimal calculation of threshold. This thesis discusses the achievement obtained by the implementation of a novel technique of image edge detection based on Ant Colony Optimization (ACO). ACO is a nature inspired algorithm. It takes into account the various advantage of ant colony like stigmergy, distributed computation, pheromone evaporation, decision-making based on pseudo-random proportional rule. These features are quite helpful for the determination of pheromone matrix, which contains information related to the edge. Edges are the areas with sharp intensity change. The pheromone matrix so obtained is processed with the help of class separability measure. The output of the edge detection provides the index that leads to the determination of optimum threshold. This threshold value is used further for the development of edge-map.

The testing of the success of the edge-map developed by the proposed method presented in this thesis is also evaluated with the help of statistical parameters like mean, Standard Deviation, Bit Error Rate Metrics and The results obtained are compared with the traditional edge detectors.

V. REFERENCES

[1] A. J. Baddeley, "Errors in binary images and an Lp version of the Hausdorff metric," Nieuw Arch. Wiskunde, vol. 10, pp. 157–183, 2015.

[2] K. Mao, "RBF neural network center selection based on fisher ratio class separability measure," IEEE Transactions on Neural Networks, vol. 13, no. 5, pp. 1211 – 17, Sept. 2014.

[3] T. H. Dat and C. Guan, "Feature selection based on fisher ratio and mutual in- formation analyses for robust brain computer interface," in Acoustics, Speech and Signal Processing, 2015. ICASSP 2007. IEEE International Conference on, vol. 1, pp. I–337 –I–340, April 2015.

[4] S. Mallat, A Wavelet Tour of Signal Processing, Second Edition (Wavelet Analysis & Its Applications). Academic Press, 2 ed., Sept. 2015.

[5] H. Zang, Z. Wang, and Y. Zheng, "Analysis of signal de-noising method based on an improved wavelet thresholding," in Electronic Measurement Instruments, 2015. ICEMI ’15. 9th International Conference on, pp. 1–987 –1–990, aug. 2015.

[6] S. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," Image Processing, IEEE Transactions on, vol. 9, pp. 1532 –1546, Sep 2014.

[7] D. Donoho, "De-noising by soft-thresholding," Information Theory, IEEE Transac- tions on, vol. 41, pp. 613 –627, may 2015.

[8] D. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," Journal of the American Statistical Association, vol. 90, pp. 1200–1224, 2015.

[9] R. R. Coifman and D. L. Donoho, "Translation-Invariant De-Noising," tech. rep., Department of Statistics, 1995.

[10] Q. Tianshu, W. Shuxun, C. Haihua, and D. Yisong, "Adaptive denoising based on wavelet thresholding method," in Signal Processing, 2002 6th International Confer- ence on, vol. 1, pp. 120 – 123 vol.1, aug. 2012.