Hybridization of Moth Flame Optimization and Gravitational Search Algorithm and its application on detection of Food Quality

Undergraduate Thesis

submitted in partial fulfillment of the requirement of BITS F422T

by

Anirudh Bhutani

2013A7TS114P

Under the Supervision of

Dr. Lavika Goel

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

Acknowledgment

I acknowledge my thesis advisor, Dr. Lavika Goel for being a constant guidance during this research project. I would also like to thank Dr. Sundaresan Raman, who supported me with the necessary technical knowledge when required.

I thank them for giving me the opportunity to a project which applies computer science to real world problems, and goes beyond just a simulation. This project enabled me to realize the design problems facing the prototyping of a real-world challenge and has help me gain immense knowledge.

I would also like to thank Dr. Satyam Srivastava and Dr. Shashikant Sadistap at Central Electronics Engineering Research Institute, Pilani, for lending me the state of the art equipment, so that we could proceed with the research.

Certificate

This is to certify that the thesis entitled '**Hybridization of Moth Flame Optimization and Gravitational Search Algorithm and its application on detection of Food Quality'** is submitted by Anirudh Bhutani bearing the ID No. 2013A7TS114P in partial fulfillment of the requirement of the course BITS F422T. This thesis embodies the work done by him under my supervision.

Dr. Lavika Goel
Assistant Professor
Department of Computer Science and Information Science
Birla Institute of Technology and Science, Pilani

Contents

1 Abstract

Gravitational Search Algorithm is an optimization algorithm inspired by the Newton law of Gravitation and the Newton's laws of motion. Moth Flame Optimization is another optimization algorithm, motivated by the locomotion of moths around a light source. Both these algorithms, have tried to model the search agents, and altered its properties like mass, Gravitational constant, fitness, location etc. in order to find the most optimal value. The optimization algorithms usually solve only a class of problems, and therefore the search for a faster, and more comprehensive algorithm is always on. By combining the Moth Flame Optimization and Gravitational Search Algorithm, the performance is expected to improve across various measures.

Project Food Sense aims to use this improvement in order to find the degree of rottenness of various food items. This will help decrease the losses in food storage, and early detection of spoilage of food, in order to minimize monetary losses due to food and storage. This hybrid method is used in improving the results of segmentation. We have successfully implemented it over K-means and Multi-level thresholding.

On application of our optimization algorithm to K-means clustering, we aim to reduce the mean squared error from each data point to the centroid of the cluster. Since the initialization of the cluster centers is random in the original algorithm, the mean squared error varies with each application, even on the same dataset. Through the application of optimization algorithm, we aim to bring down the mean squared error and bring uniformity to the application of K-Means clustering.

For Multi-level Thresholding, the brute force approach to finding the thresholds is a very expensive process. Finding the threshold, for multiple clusters, where each threshold could be a value anywhere between 0 and 255, the task becomes very expensive. Also, this is mostly a manual process. However, the optimization algorithm aims to bring down this complexity by using multiple search agents, each trying to find the most optimal threshold value, while

communicating to each other the most optimal value. This reduces the algorithm complexity, and in turn automates the whole process of finding the right values. This has multiple applications as the problem complexity increases as the dimensions or color space changes. However, this method can be extended to multiple color spaces, finding the best threshold values for each color space parallelly.

Analysis is done through these improved segmentation techniques and then texture analysis techniques like Grey Level Co-Occurrence Matrix, Local Binary Pattern etc. are explored and applied to detect the food quality. Each segment detected through the segmentation techniques mentioned above are then labelled according to its disease, and a model is created. These are then tested for in the test images.

In addition to the vision data, odor sensors also find their place in the midst. Intuitively, the food items are checked for quality by looking and smelling them, when done manually. Our aim to automate the process of detecting deterioration is therefore supported by the use of QCM gas sensors. These return the value of presence of particular gasses in the carrier gas, and each reading in PPM becomes another feature for training the model. We have used Alpha Fox 2000 foe collecting the gas data for the infected fruits.

The long-term aim of the project remains real time application of the proof of concepts we have tested. The features extracted, including the texture analysis on the vision data, and the gas sensors' readings, show great promise our aim of reducing losses through spoilage in food storage facilities and improving food safety in messes.

2 Problem Statement

Hybridize Moth Flame Optimization and Gravitational Search Algorithm to improve its results and apply this technique to better the segmentation techniques.

3 Literature Review

3.1.1 Image processing Techniques:

Artificial Neural Networks: Inspired from the biological neural system, this machine learning technique tries to simulate the human intelligence. Here, the physical properties extracted from the images can act as quality factors. It results in near human level performance in areas of color, content, shape, and texture inspection. This has been applied in many food categories like fishery, fruits, vegetables, grain and meat and has resulted in accuracy of detection nearly 90%.

Optimization Techniques: Classification can also be done using optimization techniques. Appropriate fitness function needs to be chosen and optimized accordingly. Rule sets can be generated in order to do classification of test data.

Color Space:

Color spaces are representations of different colors using different models. There are wide variety of color spaces to choose from each having its own benefits.

RGB: Widely used color space, since it works in a way similar to human vision. Views each color as a mixture of red, green and blue in different proportions.

HSV: Unlike RGB it separates luma (illumination information) from chroma (color information). Mainly used when dealing with variable illumination. HSV images can be easily segmented.

LAB: The components of this color space are Luminance, a and b(color opponent dimensions). This color space is efficient in digital image manipulations i.e. handles shadows, noise etc. well.

3.1.2 Nature Inspired Algorithms

There has been research going on in this area for a long time now and there have been several algorithms that are inspired from nature like the human mind, ants, bees, Geo sciences etc. A subset of nature inspired techniques in Swarm Intelligence. Swarm Intelligence refers to the collective behavior of several individual agents communicating with each other. It is a totally decentralized mechanism. Agents share knowledge among themselves by communicating. This is one of the most used techniques for optimization. There are algorithms like Ant Colony Optimization, Swarm Particle Optimization, Bee Clustering Algorithm, Gravitational Search and Moth Flame Optimization each algorithm mimicking the swarm behavior found in nature.

3.1.3 Previous Research in Food Safety

With the advancements in computer vision, physical properties such as shape, size, position etc can be efficiently and easily measured. Combined with that, the rapid growth in computer hardware has been exploited to use of better calibrated cameras for usage in collecting data. Computer vision has definitely played a very important role in automated food safety systems.

Mechanisms for food safety already exist in almost all countries but most of them involve manual inspections which in turn take time and resources. Below are examples of food products that have been studied extensively.

Apples: Research related to the quality of bicolored food has been done extensively. Multispectral imaging has been employed to evaluate the quality of bicolored apples. Geometrical and textural properties are invaluable in identifying defects in food. Kang and Sabarez have proposed a segmentation algorithm to segment apples and then use a classifier that will classify apple slices into multiple classes. There has also been research to predict the color changing process of a freshly cut apple.

Bananas & Oranges: During ripening process bananas exhibit a steady alteration in shade and surface. At a swift rapidity, banana skin deteriorates to form yellowish-green to dark shades. Fourier analysis was a potential and promising method for evaluating spots in banana peels.

There has been research going with respect to various other food products such as Potatoes, Meat, Berries, Dates Mangoes, Pears etc.

Recently nature inspired algorithms which comprise of algorithms modelling the human mind, artificial immune system, swarm based algorithms and geo science based computing have emerged as efficient techniques to handle a diverse set of problems. Fuzzy Set Theory, Genetic Algorithms, Swarm Intelligence based Algorithms, etc. are examples of Nature Inspired Techniques. Nature Inspired Algorithms have been applied to a wide variety of fields ranging from Computer Vision, Clustering, Learning, General Optimization Problems to name a few.

Nature Inspired Metaheuristic Algorithms have been used extensively for Partitional Clustering. Genetic Algorithms and Swarm Intelligence algorithms have been proven to give good results in partitional clustering. An in depth survey of nature inspired metaheuristic algorithms used for partitional clustering can be found[1].

There have also been approaches [2] where the Gravitational Search Algorithm has been used for data clustering. The algorithm proposed uses concepts of the Gravitational Search Optimization Algorithm and applies them to clustering. Candidate solutions for clustering and generated randomly and then the candidates interact using the Newton's second law of gravity to search the sample space. A comparison of the proposed algorithm has been provided, where comparisons have been made with standard clustering algorithms including, K-means etc.

There have also been hybrid approaches combining nature inspired algorithms with the classical algorithms like in [3] where a combined hybrid approach of the famous K-means algorithm along with Gravitational Search Algorithm is presented. Gravitational Search Algorithm has been used to improve the cluster quality as well as to speed up the convergence of K-means. Once a random initial population is created. The initial population plays an important role in the algorithm. The cluster centers that are updated in each iteration are then updated by the Gravitational Search Algorithm in order to make the Mean Square Error as low as possible. It is an obvious fact that a good clustering tries to minimize the Mean Square Error.

Nature Inspired algorithms also have found applications in computer vision and related areas, Multi Threshold Segmentation for example. A comparison of nature inspired algorithms for multi threshold image segmentation has been provided in [4] . Algorithms like PSO (Particle Swarm Optimization), ABCO (Artificial Bee Colony Optimization) and DE (Differential Evolution). The results have shown that Differential Evolution is a superior algorithm among the three. It has the minimum Helligner distance and is also the most efficient.

3.1.4 Previous Research in Food Safety using Nature Inspired Algorithms

With the increased awareness among consumers about nutrion and food safety aspects, the food industry is coming up with efficient and highly accurate food quality and food safety systems. Computer vision being a viable, non destructive approach can be used to estimate food quality by using specific characteristics. Such as, color, texture, surface defects, shape etc.

Advances in nano technology can also be exploited for the same.

3.1.4.1 Nanotechnology

The paper discusses the use of low-cost portable Nanoparticle -based technology for rapid assessment of food safety [5]. The use of Gold, Silver, Cerium Oxide Nanoparticles as well as Low cost platforms for the detection of biological and chemical contaminants Methods for detecting Microbial Contamination, Pesticides, Metal Contaminants and Mycotoxins has been explained. The challenges for practical implementation as well as the direction of future research have also been discussed.

It also discusses about the uses of Nanotechnology in Food safety. Several applications of nanomaterials for food packaging and food safety are reviewed [6]. Silver nanoparticles in particular, have been used as potential antimicrobial agents. Nano material based arrays for detection of gases like methane, ethane and other food borne particles have been used.

Some advanced techniques involving surface based Raman scattering for identification and detection of organic molecules have also been discussed.

The paper speaks about recent progress in food safety analysis using nano bio sensing. The paper talks about the various roles nanomaterials can play in food safety analysis. Toxins, Pathogens, Pesticides and Antibiotics are some of the many contaminants detected by nano bio sensing [7].

3.1.5 Computer Vision

With advances in hardware, high resolution hyperspectral and multispectral imaging is being used extensively. Widely different data is acquired from hyperspectral and multispectral sensors is then analyzed on a case by case basis in majority of the cases.

Vibrational Spectroscopy, Hyperspectral Imaging and Multispectral Imaging, Biomimetic Sensors are inspired array of sensors designed to mimic the olfactory and gustatory systems of humans called E-nose and E-tongue respectively. This variation in data results in the formation of multivariate data sets

Chemometric, Machine Learning and Evolutionary Algorithms are used to extract information from the data extracted. Supervised as well as Unsupervised learning is applied to generate models which are then validated later on. The paper then goes on to talk about the sensor quality, external features, bruise detection and other quality parameters.

The journal talks about the kind of hardware used for a standard computer vision setup and goes on to talk about the computer vision techniques used for food safety of Bakery products, Vegetables, Fruit, Grain, Prepared Consumer foods [10].

3.1.6 Bakery Products

In regard to bakery and related products, appearance plays a very important role. Appearance of the bakery product often speaks about its freshness, flavor etc. Thus, consumers pay particular attention to the visual appearance of bakery products. That is why, industries have come up with a variety of approaches to analyze the quality of bakery and baked products. Approaches varying from measuring the defects in baked loaves, to analyzing the height and slope of the bread loaf have been proposed.

Vision related techniques have been extensively developed to study the internal structure of bread crumbs, and have been applied on inspecting the quality of muffins. Systems that assess the quality of chocolate chip cookies have been developed, they use features like the shape, density of chocolate chips in the biscuits, color etc. to determine whether the chocolate chip cookie meets the industry standards.

3.1.7 Meat and Fish

Computer vision is yet again used in the quality estimation of Meat and fish related products. With respect to meat especially, meat marbling has been analyzed using vision techniques. McDonald and Chen thoroughly investigated and proposed an automated computer vision based beef quality assessment system. Their work was one of the earliest in this respect. Discrimination between fat and lean muscles was identified and used in the quality grading of beef. An accuracy of 0.86 was reported with a data of around 60 steak images. Beef texture was realized to be an important characteristic for measuring beef tenderness . Statistical regression as well as Artificial neural network based methods have also been proposed in this regard.

Color, marbling and textural features were extracted from beef images and analyzed using statistical regression and neural networks.

3.1.8 Fruit

Computer vision has been used for such tasks as shape classification, defect detection, quality grading and variety classification. Defect segmentation on Golden Pleasant apples was performed by CMV. A color model developed was used as a standard for comparison with sample images. The transform converts a round object image to a planar object image allowing fast feature extraction, giving the system an examination capacity of 3000 apples/min from the three cameras, each covering 24 apples in the field of view.

3.1.9 Vegetables

Computer vision techniques have been used to analyze potatoes based on the HSI color system. Similar approaches have also been applied for the quality assessment of mushrooms, nuts, etc. Computer vision has proven to be an efficient method for vegetables, especially in congruence with the HSI color system. Size, texture, shape and black spots (flaws) are some of the most commonly used features for vegetables in general.

Similarly a paper presents a review of various computer vision approaches applied for food safety of meat, fish, vegetables, fruits etc.[13]. This paper especially talks about the kind of learning techniques applied in food safety algorithms. They range from Artificial Neural Networks used for classification of cereal grains, fruits especially apples and fish and meat and vegetables, Segmentation is discussed as a preprocessing step where in ANN is applied on the segmented image. Statistical Learning, Decision Trees and their applications to various food products has been discussed [14].

4 Optimization Algorithms

Optimization problems refer to the process of finding the best possible solution for a given problem. This can be finding the minimum value, if the problem is a minimization problem, or the maximum value if the problem is a maximization problem.

Given that the nature of arriving to the optimal value differs greatly for each optimization algorithm, each is able to solve only a subset of optimization problems well. Thus the wide range of the optimization problems, and with the increasing complexity of them, calls for development of new optimization problems. They might solve a different class of problems more efficiently than others, or may improve upon the time taken to converge to the final optimal value. Both of these measures are marked as improvement over the existing algorithms.

Optimization problems also often encounter the problem of local minima. This is when the optimal value returned by the optimization algorithm is some local minima instead of the global minima. Thus the quest to find algorithms which successfully evade this problem is still a challenge to the scientific community.

A proposed heuristic to curb the problem of local minimum is to use swarm intelligence, where the search agents are spread across the domain, and they communicate some information in each iteration, in order to reach the global minimum. Such algorithms are usually divided into 2 phases:

Exploratory phase

The first phase of the multi-agent search, where the search agents are spread across the solution domain. This is often done randomly, in order to increase the chances of search agents being spread across the whole domain, ensuring better chances of avoiding the local minima. The search agents then communicate their information, usually through their fitness values, to the others.

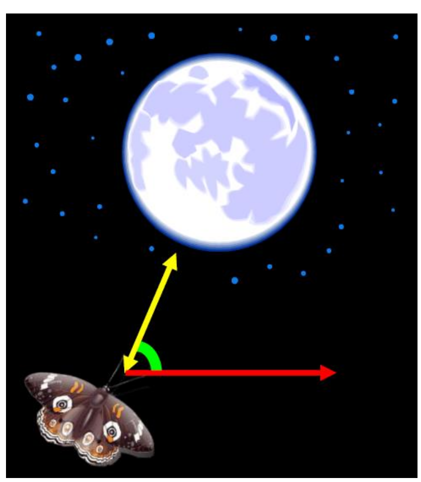
Exploit phase

This is the later phase of the swarm based optimization algorithm. This happens after some iterations of the exploratory phase, when the search agents have certain amount of information about the fitness of the other agents. This is when the search agents start to converge around the fittest search agents. The search agents then assume to role of finding the optimal value around the fitter moths.

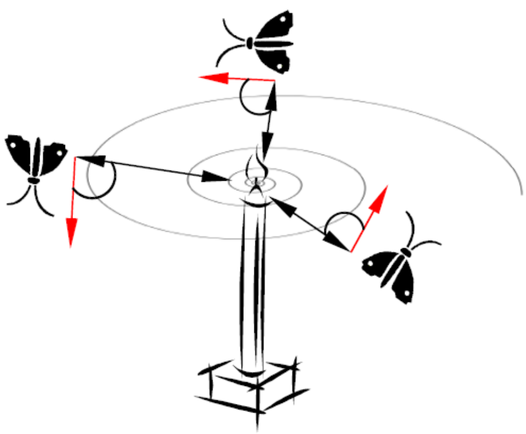
There is no clear distinction between the two phases, and may overlap. Additionally, there is no heuristic to mark the transition between the two, and is dependent on the algorithm to switch form the exploratory phase to the exploitation phase. If the algorithm takes more time to explore, the chances of returning some local minima reduces. However, it may lead more time, i.e. number of iterations, to converge to the optimal value. It is therefore a trade-off between the accuracy of the result and the time taken to reach an optimal value.

4.1 Moth Flame Optimization

Moth Flame Optimization (MFO) [16] is a nature inspired algorithm which is motivated by the navigation of moths in night. Moths use the moon as reference to travel through the night. This mechanism of navigation is called as Transvers Orientation. In this method, the moths move at a fixed angle with respect to the moon. Since the moon is at a very large distance, the motion is effectively linear.



However, in an experiment, it was observed that in presence of a light source extremely close to the moth, leads to the moth confusing the light source with the moon, and navigates similarly with respect to the light source, as it does with respect to the moon. However, since the light source is too close, the moth falls into a deadly spiral, converging into the flame.



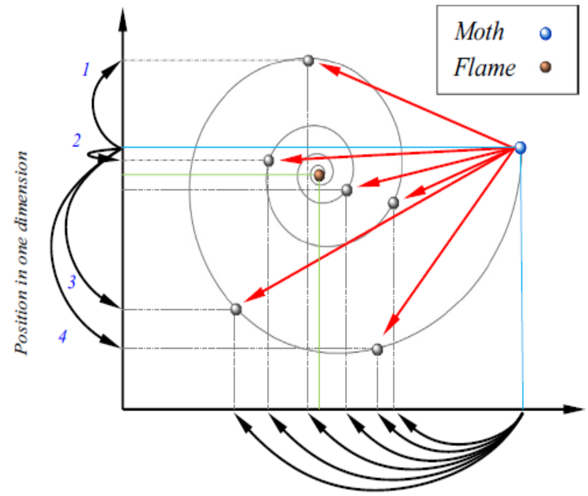


Fig. 2. Spiral flying path around close light sources.

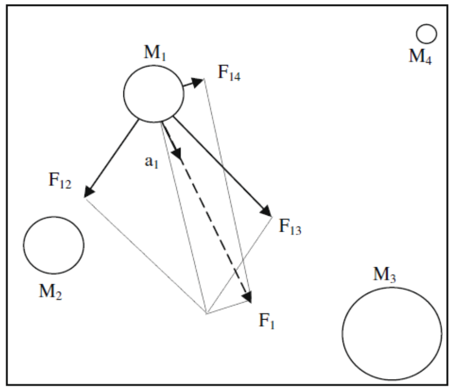
In this algorithm the candidate solutions (search agents) are the moths. The problem's variables are the positions of the moths in the hyperspace.

4.2 Gravitational Search Algorithm

Gravitational Search Algorithm (GSA)[17] is another nature inspired algorithm which is motivated by the Newton's law of Gravitation, and the laws of motion. It is based on the concept that two masses in space attract each other with a force.

$$F_{ij} = G\frac{M_{aj} \times M_{pi}}{R^2},$$
$$a_i = \frac{F_{ij}}{M_{ii}},$$

In this algorithm, the candidate solutions are the masses, spread across the hyperspace and each has an associated mass, which dictates the motion of all the search agents, which eventually converge to the most optimal value.



4.3 Hybrid Algorithm

The algorithms are themselves very powerful. By using multiple search agents, and adding some degree of randomness in the locomotion, avoid local minima to a large extent. However, the hybrid algorithm seeks for faster convergence, while maintaining the exploration of both the algorithms. This is expected to be achieved through the combined locomotion in every iteration.

MFO, due to its spiral locomotion avoids the local minima, as the agents are not directly attracted towards the optimal flame in a linear trajectory. On the other hand, the locomotion is in a logarithmic spiral, which lets other moths explore the domain, for better optimal value.

GSA, on the other hand, uses a linear force on the search agents while it seeks to converge to the most optimal value. This typically introduces a dominance of the exploitation over exploration.

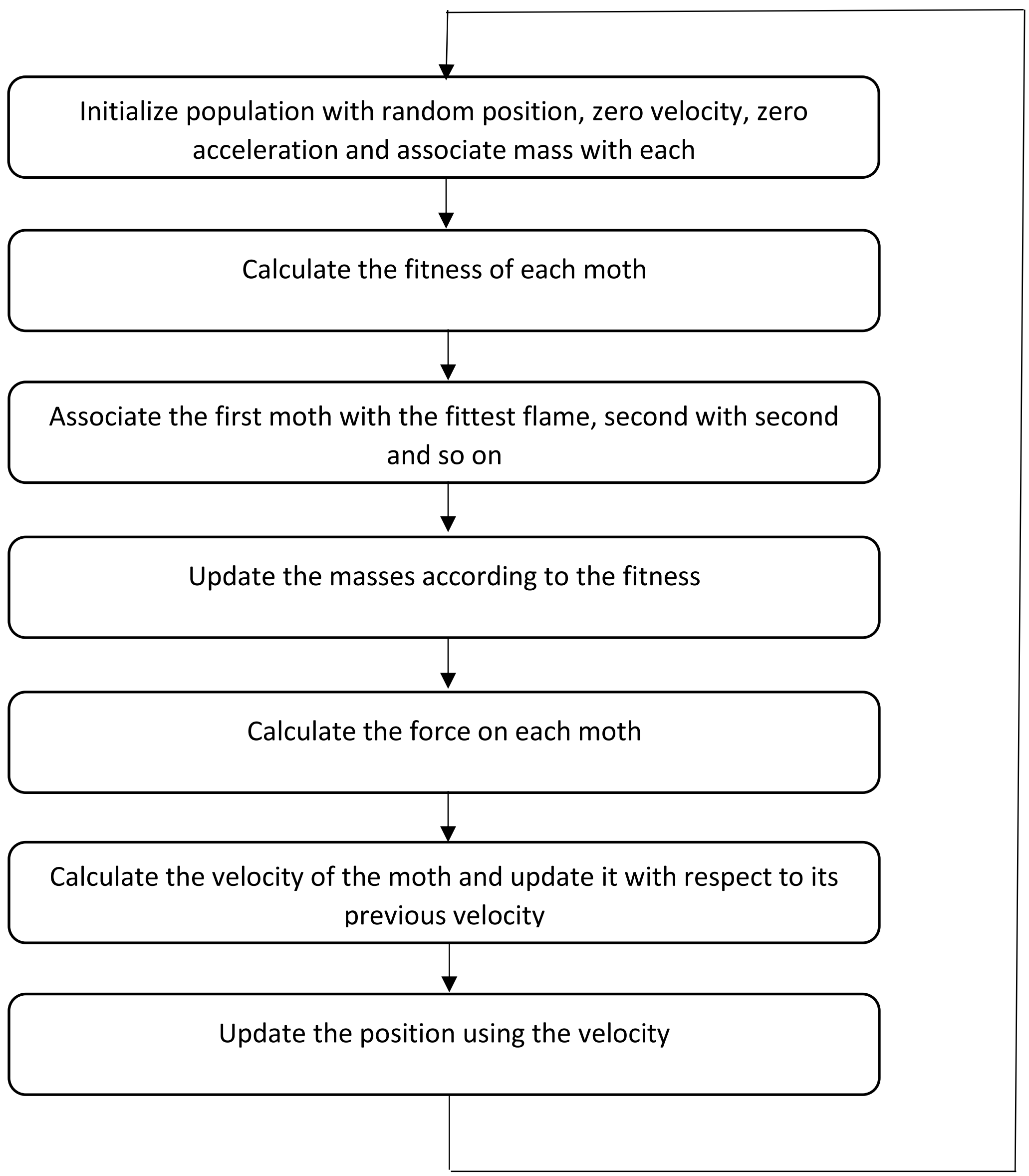
Thus, in the hybrid algorithm, the exploration of MFO and the exploitation of GSA is exploited, in order to achieve better results, while maintaining the integrity of the optimization algorithms individually, and evading the convergence to any local minima.

4.3.1 Justification for hybridization

The algorithm exploits the nature of the moth flame optimization, by using its exploratory in nature, and utilizes the exploitation of the gravitational search algorithm. The intuition behind this approach to hybridize these two algorithms is that the MFO utilizes the logarithmic spiral for exploration, while the GSA uses the linear motion for locomotion. This enables the hybrid algorithm to exploit both the properties of the individual algorithm and therefore fit together, covering what the other fails to cover.

The fitness measure is mass for GSA while the fitness measure for MFO is distance to the fittest flame. This adds the variability to exploit both the behavior in a single algorithm, thus rendering the hybrid as a better fit. MFO makes the search agent go around the fittest flame, while GSA ensures faster convergence by bringing all the search agents closer to the fittest search agent. Thus, these algorithms are compatible, and produce the desired effect.

Flowchart for the algorithm:



4.3.2 Pseudo Code

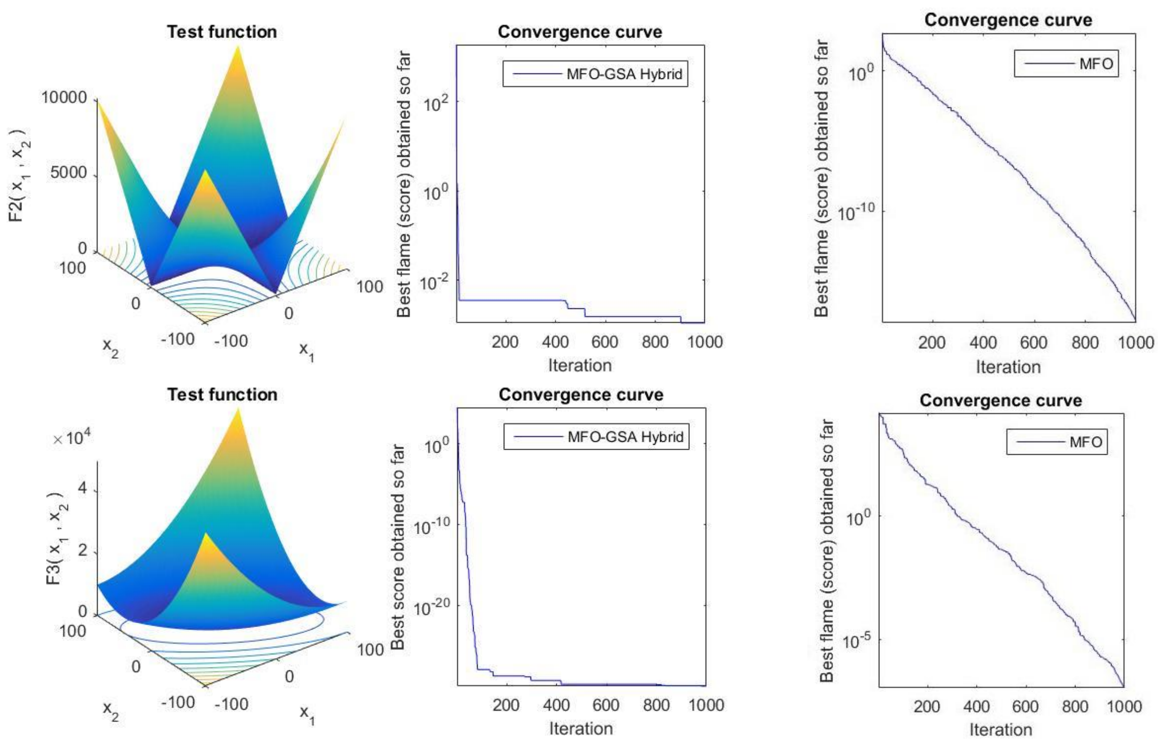
```
initialize moth_position, moth_velocity, moth_acceleration
calculate initial fitness for each moth
while(iteration < max_iteration)
        update flame_number
        if (iteration == 1)
                F = sort(M)
                F' = sort(M')
        else
                F = sort(Mt-1, Mt)
                F'=sort(Mt−1, Mt);
        end
        for i=1: n
                for j=1: d
                        Update r and t
                        D(i) = |F(i) – M(i)|
                        M(i) = D(i) . e^bt . cos(2∏t) + F(j)
                end
        end
        G = G0(1/iteration)^b
        for i=1:n
                mass(i) = 1/fitness(i)
                for j=1:n
                        r = moth_position(i) – moth_position(j)
                        F = G*mass(j)/r
                end
                moth_velocity(i) = moth_velocity(i)*rand(0,1) + F
                moth_position(i) = moth_position(i)*rand(0,1) + moth_velocity(i)
        end
end
```

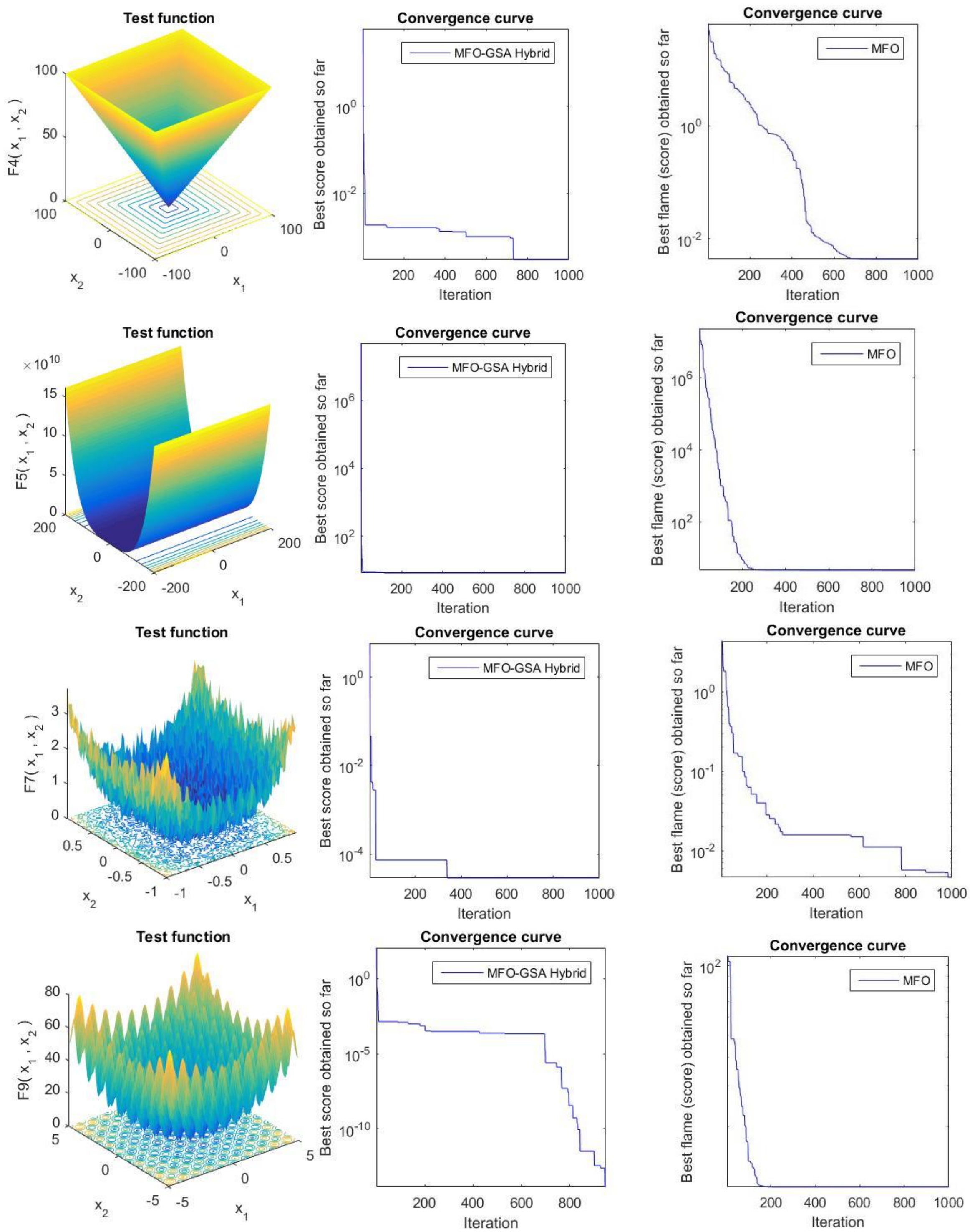
The flow of the algorithm is as follows:

- First, the population is initialized to random positions in the search domain.
- Next, the fitness is calculated for each search agent.
- Then the moths are associated with their serial order to the fittest flame. The first moth in the list is associated with the fittest flame, second in serial with second best flame and so on.
- Now the search agents rotate around their associated flames in the spiral trajectory. After that, the moths attract each other and come closer to the fitter flames.
- This process is repeated until the maximum number of iterations.

5 Testing Hybrid Optimization on Benchmarks and Results

FUNCTION	MFO	HYBRID	GSA	GA
F1	6.6092e-21	5.9991e-17	1.321152	21886.03
F2	2.6592e-15	0.0010437	7.715564	56.51757
F3	7.2411e-09	2.2439e-17	736.5616	37010.29
F4	0.0043443	0.00031035	12.97988	59.14331
F5	15.7733	8.0849	77360.4184	313.21418
F6	0.12108	0.32572	2.86418	52.64496
F7	0.0046347	2.8854e-05	1.03951	20964.83
F8	-2879.4219	-2471.4722	-102.5649	-13.37504
F9	10.9445	0	25.46556	2.14891
F10	4.4409e-15	0.00011668	5.32221e-10	5.4981e-5
F11	0.22639	2.9199e-14	6.1948e-5	7.4198
F12	8.4387e-32	0.024027	1.28685e-25	0.02959
F13	1.992	2.9821	12.4944	4.5195




Test function
F4(x1 , x2)
x2
x1
Convergence curve
MFO-GSA Hybrid
Best score obtained so far
Iteration
Convergence curve
MFO
Best flame (score) obtained so far
Iteration
Test function
F5(x1 , x2)
×10^10
x2
x1
Convergence curve
MFO-GSA Hybrid
Best score obtained so far
Iteration
Convergence curve
MFO
Best flame (score) obtained so far
Iteration
Test function
F7(x1 , x2)
x2
x1
Convergence curve
MFO-GSA Hybrid
Best score obtained so far
Iteration
Convergence curve
MFO
Best flame (score) obtained so far
Iteration
Test function
F9(x1 , x2)
x2
x1
Convergence curve
MFO-GSA Hybrid
Best score obtained so far
Iteration
Convergence curve
MFO
Best flame (score) obtained so far
Iteration

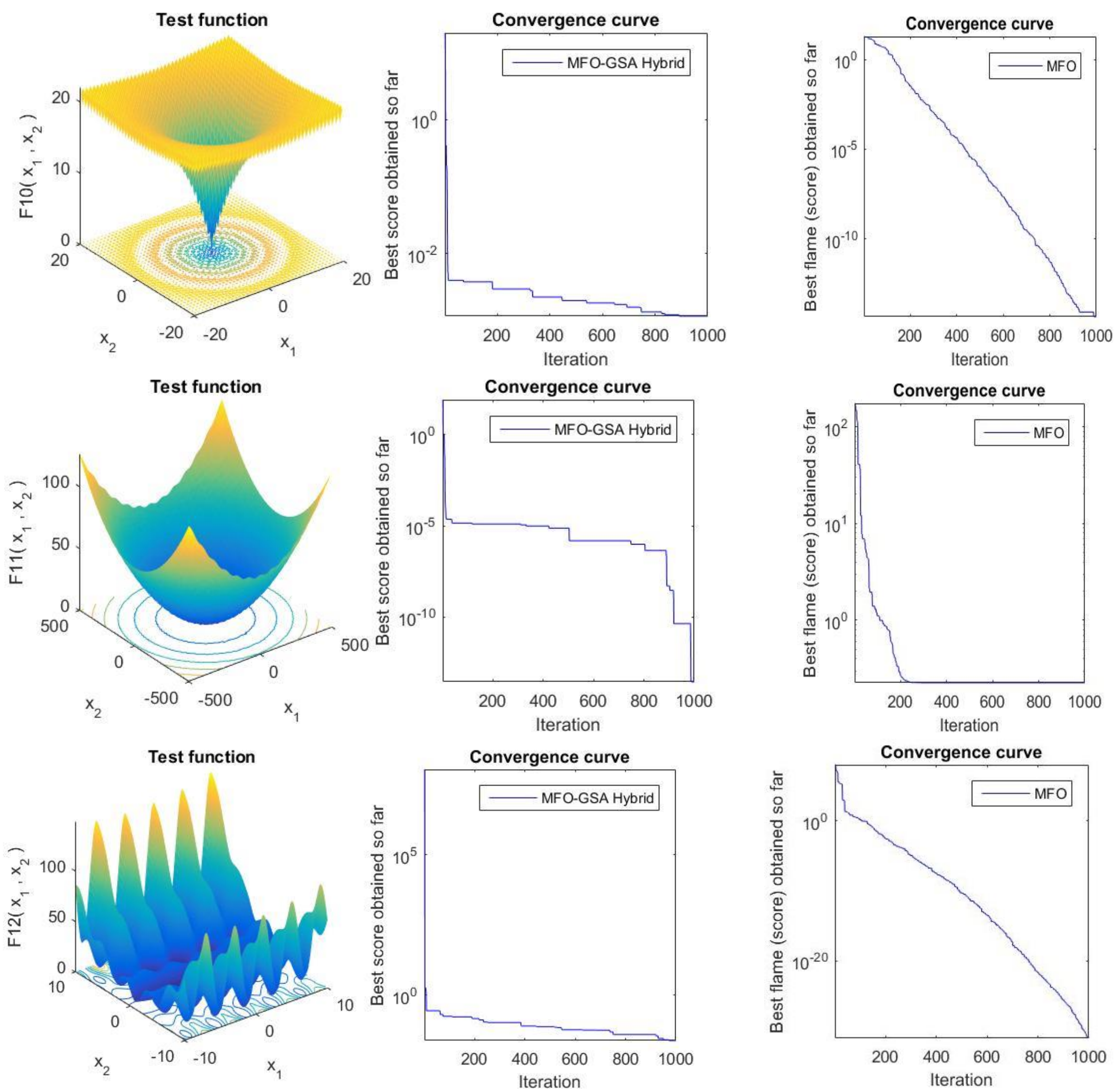


Figure: Comparison between Hybrid and MFO

Benchmark functions:

Unimodal Functions:

Function	Dim	Range	Shift position	f_{min}
$f_1(x)=\sum_{i=1}^{n}x_i^2$	100	$[-100, 100]$	$[-30, -30, \ldots, -30]$	0
$f_2(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	100	$[-10, 10]$	$[-3, -3, \ldots, -3]$	0
$f_3(x)=\sum_{i=1}^{n}\left(\sum_{j-1}^{i}x_j\right)^2$	100	$[-100, 100]$	$[-30, -30, \ldots, -30]$	0
$f_4(x)=\max_i\{\lvert x_i\rvert, 1\leqslant i\leqslant n\}$	100	$[-100, 100]$	$[-30, -30, \ldots, -30]$	0
$f_5(x)=\sum_{i=1}^{n-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	100	$[-30, 30]$	$[-15, -15, \ldots, -15]$	0
$f_6(x)=\sum_{i=1}^{n}([x_i+0.5])^2$	100	$[-100, 100]$	$[-750, \ldots, -750]$	0
$f_7(x)=\sum_{i=1}^{n}ix_i^4+random[0, 1)$	100	$[-1.28, 1.28]$	$[-0.25, \ldots, -0.25]$	0

Multimodal Functions:

Function	Dim	Range	Shift position	f_{min}
$F_8(x)=\sum_{i=1}^{n}-x_i\sin\left(\sqrt{\lvert x_i\rvert}\right)$	100	$[-500, 500]$	$[-300, \ldots, -300]$	-418.9829×5
$F_9(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	100	$[-5.12, 5.12]$	$[-2, -2, \ldots, -2]$	0
$F_{10}(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e$	100	$[-32, 32]$		0
$F_{11}(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	100	$[-600, 600]$	$[-400, \ldots, -400]$	0
$F_{12}(x)=\frac{\pi}{n}\left\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]+(y_n-1)^2\right\}+\sum_{i=1}^{n}u(x_i, 10, 100, 4)$	100	$[-50, 50]$	$[-30, -30, \ldots, -30]$	0
$y_i=1+\frac{x_i+1}{4}$				
$u(x_i, a, k, m)=\begin{cases}k(x_i-a)^m & x_i>a\\ 0 & -a<x_i<a\\ k(-x_i-a)^m & x_i<-a\end{cases}$				
$F_{13}(x)=0.1\left\{\sin^2(3\pi x_1)+\sum_{i=1}^{n}(x_i-1)^2[1+\sin^2(3\pi x_i+1)]+(x_n-1)^2[1+\sin^2(2\pi x_n)]\right\}+\sum_{i=1}^{n}u(x_i, 5, 100, 4)$	100	$[-50, 50]$	$[-100, \ldots, -100]$	0

6 Image Segmentation – Computer Vision

In the broad field of Computer Vision, Image Segmentation falls under the subcategory of Image Recognition and Understanding. Segmentation is a fundamental technique in Computer Vision and plays a vital role in Object Recognition, Image Retrieval and Medical Image Analysis. The main role of segmentation is to group similar set of pixels into the same groups. It's importance just cannot be stated, so much so that Segmentation is often the first step applied for understanding images automatically. It acts to bridge the gap between image processing and computer vision.

Image segmentation involves grouping sets of pixels in an image in appropriate segments. The resultant segments formed after segmentation provide valuable information that help in understanding the image and make processing easier. Formally speaking, Image Segmentation is the process of assigning groups of pixels that share some common characteristics to same segments. There is still a lot of work that needs to be done, since a reliable and automatic segmentation of objects is difficult just by using automatic segmentation algorithms.

Segmentation algorithms can in general be classified into the following types: Segmentation based on active contours, split and merge segmentation algorithms, mean shift and node finding clustering and finally segmentation using normalized cuts. The techniques that have gotten wide spread recognition because of their effectiveness are the following:

Watershed algorithm, Dynamic snakes and condensation, Divisive clustering(example K-means), Agglomerative clustering, Graph based segmentation, K-means and mixture of Gaussians, Mean shift clustering, Normalized cuts etc.

6.1 Image Segmentation using K-means clustering algorithm:

Clustering: Clustering is a data mining technique that involves grouping of given data points into clusters. The goal is to make sure that groups of data points in the same cluster have similar properties and groups of clusters in different clusters do not share many characteristics. It is a form of unsupervised learning technique. There are several algorithms for clustering groups of data points. K-means, Optics, DBSCAN, Clique, Enclus are only a few examples of clustering algorithms. Among all the clustering algorithms, K-means is one of the most intuitive and simple algorithms, nevertheless powerful. It is due to this simplicity and efficiency that K-means is used in a wide variety of domains.

K-means Clustering: This algorithm requires as an input parameter, the number of clusters that are finally needed. The number is represented by K; hence the name K-means. From the set of data points, K random points are initialized as cluster centers and points are assigned to the cluster centers using the minimum distance criterion. The goal is to make sure that points in the same cluster are as similar as possible and points in one cluster are as different as possible from the points in other clusters. The similarity function can be chosen to be anything, in the spatial data case it is Euclidean / Murkowski distance. If documents are being clustered, the document matrix can be used. For images, the pixel values, intensity values, color values, hue values etc can be used to define similarity

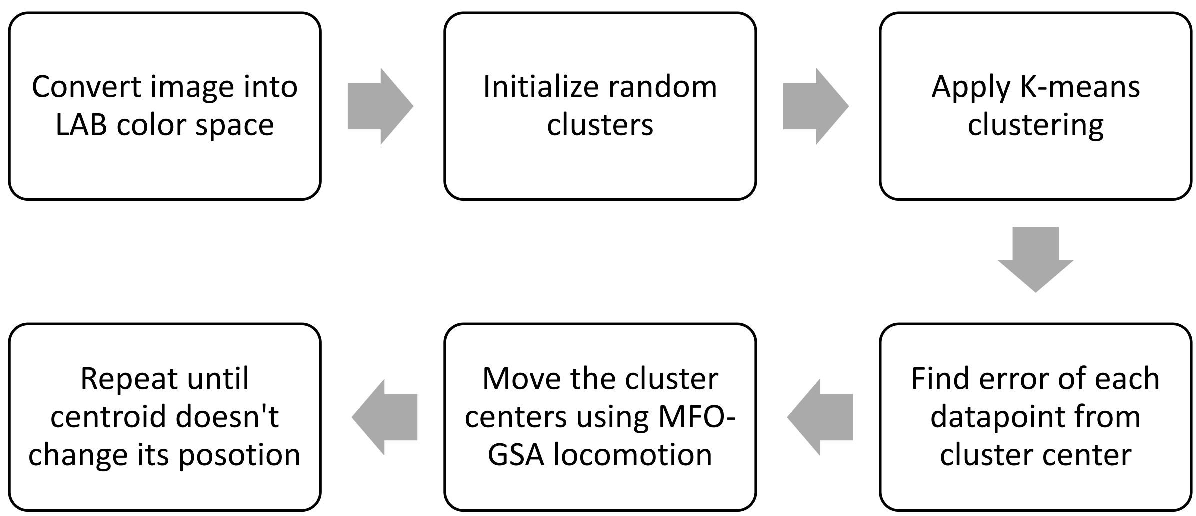
K means in Image Segmentation: The K-means clustering algorithm when applied to images will produce K clusters, also called segments. The number K is a user provided parameter. Clusters provide a grouping of the pixels that is dependent on their values in the image, but not necessarily on their locations in the image unless location is a specified property. If $X = \{x_1, x_2, ... x_N\}$ represent N pixels. The K-means segmentation algorithm on a high level has the following steps:

1. Parameter initialization: The initial K-mean points are initialized randomly from the set of pixels provided. The value of each pixel in each dimension is chosen randomly from the set of existing points, for this reason the initial K – mean points may not represent actual points in the image. But their components would be definitely being available in the image.

2. Hard Assignment of Pixel to Clusters: Given that each of the K clusters has a mean point Mu_k, every point in the set of data points is assigned to the cluster to which it is most similar. Similarity is calculated based on Euclidean distance between pixels. After this, every point is assigned to exactly one cluster Ck.

3. Parameter Re computation: The new means of the clusters are computed based on the points assigned to the cluster in the first iteration.

Steps 2 and 3 are repeated till convergence, which typically means that no pixel shifts from one cluster to another between iterations.



Pseudo code for optimized K-means segmentation:

```
initialize each particle to contain k randomly selected cluster centroids
for t = 1 to t_max do
        for particle i do
                for each data vector d,
                        dist(d, m_ij) to all cluster centroids C_ij
                        assign d to cluster C_ij: dist(d, m_ij) = min {dist(d, m_ij}
                        F = [F dist(d, m_ij)]
                endfor
                C = best(F[])
                C_ij = MFO_GSA(C)
        endfor
return C_ij

endfunction
```

The K-means segmentation technique was applied on the apple dataset. The images below show the results for various types of diseased apples.

BLOTCH

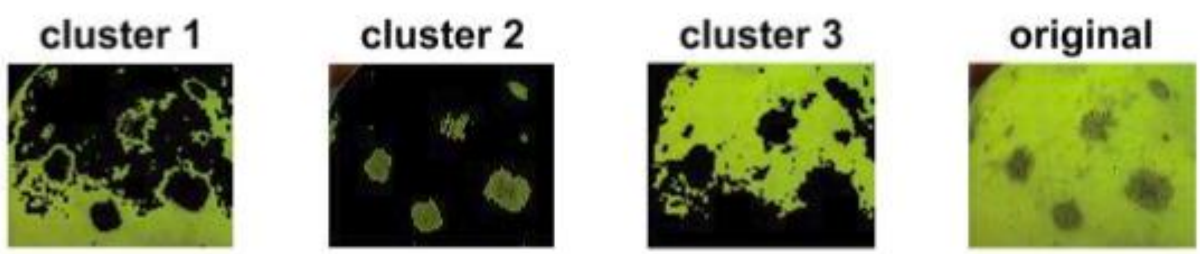


Fig1. Cluster output for apple blotch

ROT

Fig2. Cluster output for apple rot

SCAB

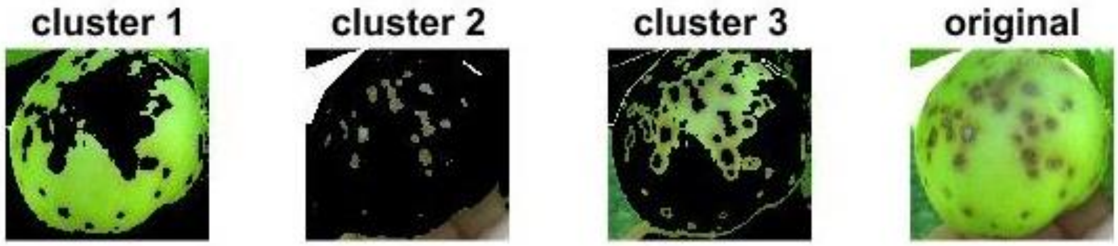


Fig3. Cluster output for apple scab

Results:

Mean Squared Error is calculated for standard datasets. It is clearly evident that the optimized k-means is performing better than normal k-means, as the error is less than or equal to the normal k-means.

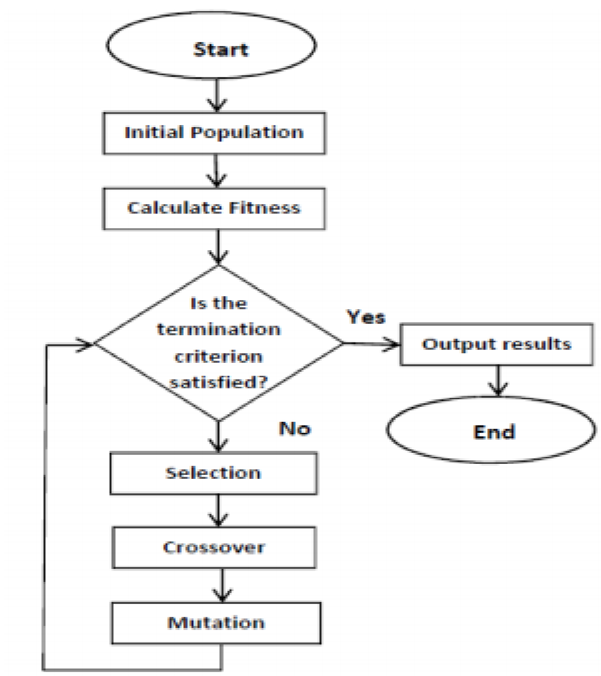
Dataset	K-Means	Optimized K-Means
Set 1	1.11	1.11
Set 2	41.59	39.54
Iris	0.54	0.52
Cancer	29.12	27.46

Analysis:

1. Optimized K-means works very well with our problem of identifying diseases in the apple dataset. It is clearly evident that optimized K-means in the LAB color space is able to segment out the diseased portion, even when it is scattered throughout the apple, in the form of small legions.
2. Optimized K-means shows an overall improvement over a normal K-means, and thus finds its application in various other domains, and not just segmentation.
3. Optimized K-means by optimizing the error is able to reduce the role of randomization to some extent as the error is corrected through the iterations of the optimization algorithm.

6.2 Image Segmentation using Thresholding:

Genetic Algorithm is a nature inspired metaheuristic algorithm that mimics genes. It is based on the evolution theory of "Survival of the fittest", literally. Genetic algorithms like other algorithms that we've discussed so far are used as approximate solutions for seemingly complex problems, like NP-hard problems. It is a swarm based algorithm, so to say. Every potential solution is represented as a chromosome and chromosomes are built up using genes.



The fundamental difference between the two lies in the fact that in our hybrid model, instead of a crossover, hybrid moth locomotion is used. The fitness function used is same as that of Genetic algorithms.

How is moth locomotion similar to Crossover: In crossover different segments of different chromosomes are interchanged. That is basically a kind of communication done between chromosomes, information exchange. The information exchange can be easily realized in a swarm like scenario like our hybrid's using the moth's locomotion due to gravity as well as flames. The final output would be the fittest image which would be the result of segmentation.

There are several techniques to overcome the obvious drawbacks of normal thresholding. The drawbacks are these. Only intensity of pixels is considered. Other properties of pixels and their neighbors is not considered.

```
Read Image

Add noise to the image

Generate image histogram

Initialize k moths with random positions and masses

for i = 1 to max iterations:

        while size(population) > size (new population):

                MFO_GSA(population)

                F_i = S_inter_cluster /S_intra_cluster

                sort(F[])

        endwhile

endfor

endfunction
```

Image Segmentation using Hybrid Optimization Algorithm: In this method, the hybrid algorithm is adopted to the segmentation problem. In this method, the given image is first made noisy by adding uniformly distributed noise to the original image I.e. $F_{ij} = f_{ij} + n_{ij}$

Initial population generation: For any pixel (i, j) the value, choose pixel value of the initial population as the value of the original pixel at (i,j) or the value of noise at (i,j). Thus, the initial population is a mixture of images with a mixture of original image values / noise values. Such a selection of the initialization helps in the exploratory phase of the optimization algorithm

Fitness evaluation: Fitness function is defined based on the following principle. Inter cluster similarity should be as less as possible and intra cluster similarity should be as large as possible. Thus the fitness is defined as a ration of $S_{intercluster}$ / $S_{intracluster}$.

Locomotion operations: In each iterations, the moths which represent images / potential solutions of segmentation, move due to the MFO + GSA motion i.e. both linear as well as spiral motion. From these newly formed moth positions along with the old moth positions, the best k moths are chosen. The flames are reduced as the iterations go on.. and finally only one moth, i.e. the solution remains. This represents the segmented image.

Handling of larger images: Divide image into subimages, and final best subimages using optimization and fitness and merge them to create final segmented image.

The final result of this segmentation is a gray level image where different intensity ranges mean different clusters. There are overlaps sometimes which have to be dealt with accordingly. But the segmentation is overall very satisfactory.

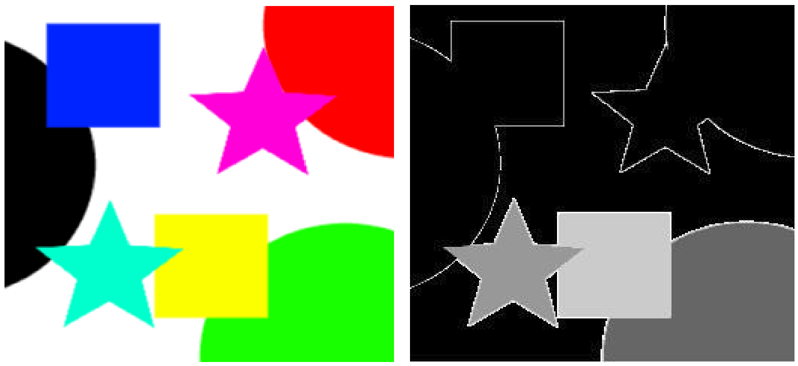


Fig: Segmentation results for standard color image

Fig: Segmentation results for rot

Fig: Segmentation results for scab

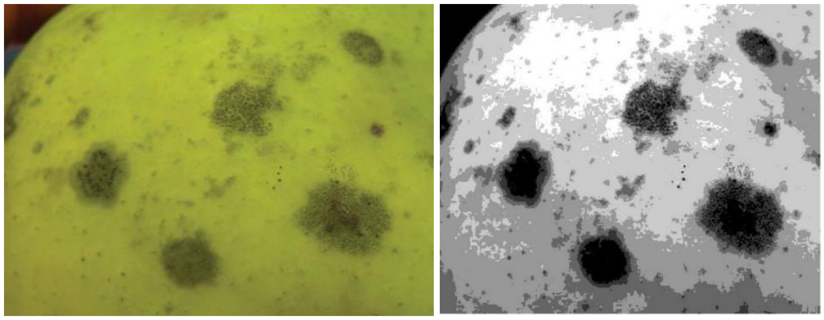


Fig: Segmentation results for blotch

Analysis:

Multi-Level Thresholding using optimization algorithm has the following advantages:

1. It performs better than the K-means algorithm used above as it does not involve the factor of randomization and searches the whole search domain for the best thresholds. This reduces the error rate, and makes the algorithm more robust.
2. For grayscale images, the pixel values range from 0-255. This search space being fairly limited makes this algorithm work very fast, producing the desired results in a very small amount of time.
3. It automates the whole process of thresholding. Thresholds are often calculated through the image and real time filters. However, that is a manual process. By automating this process, we remove the role of human, and thus improve upon the efficiency of the thresholding process.
4. As the role of human is removed, the human error associated is also completely removed from the process of thresholding.
5. This thresholding can be expanded to any of the known color spaces, RGB, LAB, HSV etc. Since thresholding is a very common computer vision technique, this automation results in an improvement over the current algorithms.

7 Feature Generation:

7.1 Local Binary Patterns:

Local Binary Pattern is a texture analysis technique. It runs on the basic concept of assigning binary values to each pixel. These values are generated through analysis of its neighbors. It is computationally inexpensive and yet effective that it finds its application in multiple computer vision problems like facial recognition. It is a robust algorithm for it works even in different illuminations and environments.

LBP describes two dimensional textures, based on the surface in the image. One is the local spatial patterns, and the other is the grey level contrast.

Implementing LBP:

1. Compare the grey level value of the center pixel with its 8 neighbors.
2. Generate the binary pattern (1 for greater pixel value for center pixel, 0 for less)
3. Convert the binary string into a decimal number.
4. Repeat this process for each pixel.
5. Store the frequency of each pixel's pattern value.
6. The number of distinct texture values becomes the dimension for the model.

Example:

157	178	220
219	**218**	255
215	219	255

218 < 157 ? …… **0**

218 < 178 ? …… **0**

218 < 220 ? …… **1**

218 < 255 ? …… **1**

218 < 255 ? …… **1**

218 < 219 ? …… **1**

218 < 215 ? …… **0**

218 < 219 ? …… **1**

The string generated for the center pixel is **00111101**. Convert the binary number to decimal, which in this case is **61**. Repeat this process for all the pixels, and make a histogram of the frequency of each LBP value.

Once we have the features, and the model, we can use any machine learning algorithms to work around with LBP features.

There are various extensions to this technique. Some of them are:

1. RGB-LBP: comparison is done for each pixel in all three spheres, Red, Green, and Blue, instead of just the grey level value, and then the results are clubbed together.

2. Multi-block LBP: the image is first divided into multiple blocks and then the LBP is calculated for each block and the results are clubbed together, as the histogram is built together for all the blocks.
3. Transitional LBP: the LBP is calculated for the transition in the clockwise direction, barring the central pixel.
4. Modified LBP: the LBP comparison are made with the average of the intensities of the 3x3 block in consideration at a time.

7.2 Grey Level Co Occurrence Matrix:

The grey level co occurrence matrix is a distribution matrix which represents the co-occurring values in the vicinity. It signifies the relationship of a sub-image of a fixed size to its surroundings. It calculates the frequency of a pixel value occurring in its vicinity either horizontally or vertically or diagonally.

Once we have the GLCMs, one can extract information out through various statistical measures like the following:

- Contrast: measures the contrast between the pixel and its surroundings.
- Correlation: measures how much a pixel is correlated to its surroundings.
- Energy: calculates sum of elements raised to the power of 2.
- Homogeneity: calculates the closeness of the GLCM to its neighbors.

7.3 Haralick Features:

There are fourteen different statistics which Haralick proposed for analysis of co-occurrence matrix. These form the fourteen textural features. Since they are derived from the co-occurrence matrix, the features are based on the relation of the pixel's grey scale values to its surroundings.

First order statistics include mean and standard deviation. These are thus focussed only on the individual pixel values. These therefore do not form good measures for texture analysis. Second order statistics are obtained by using GLCMs, and an inter pixel relationship is formed and inter dependency established. These co-occurrence matrices are calculated for each pixel in the following directions: vertically above and below, horizontally left and right, and diagonally up and below. This becomes an exhaustive neighbor search, and all information can be extracted through this exhaustive search. The fourteen Haralick features are:

- Cluster Tendency
- Measure of Correlation
- Difference Moment
- Sum Average
- Variance

- Angular Second Moment
- Inverse Difference Moment
- Contrast
- Correlation
- Entropy
- Difference Entropy
- Sum of Entropy
- Sum Variance
- Information Measure of Correlation

These features are characteristic features which can be extracted for any image, and thus can be useful in a wide application in the domain of computer vision.

7.4 Color Based Features

Color based features can also be used to conduct analysis. The analysis is done in different color spaces like Red-Green-Blue (RGB), Hue-Saturation-Value (HSV), LAB etc. Statistical measures like mean, variance, standard deviation, etc. can be used to do the analysis. A histogram is usually formed by the pixel values and then appropriate measures are chosen, once the person gets visual cues from the histogram and distribution.

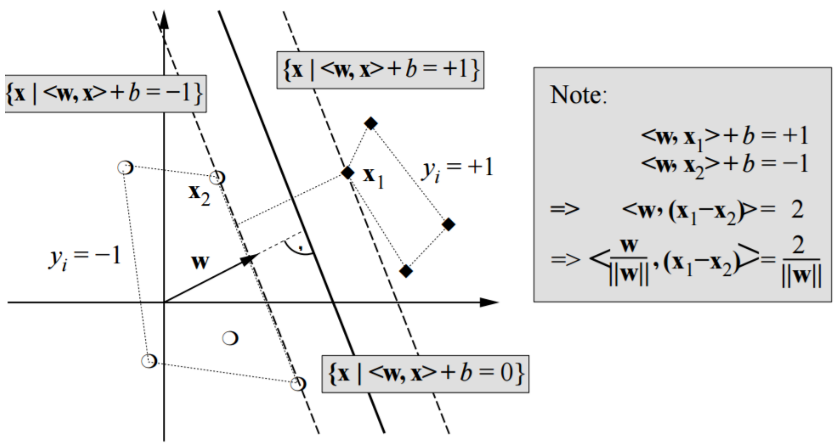
8 Classification

The next phase of the problem is to train and build the classification model for the dataset.

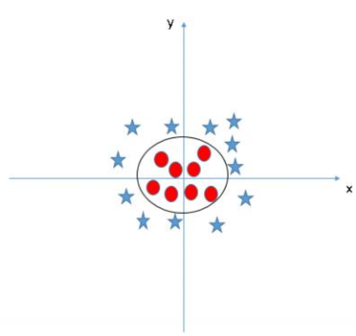
8.1 Support Vector Machine

This is a very popular algorithm for data classification. The basic concept behind the working of this algorithm is that we try to divide the space into two/more classes. This algorithm works for multiple dimension, and not just 2-D or 3-D. For higher dimensions, the plane is a hyper plane, of appropriate dimensions. The aim is to maximize the distance between the classes in the dataset. This classification algorithm is a robust and stable, and does not fail on high dimensional data.

$$\frac{1}{m}\sum_{i=1}^{m}\ell(f(x_i,\alpha),y_i) + \text{Complexity term}$$

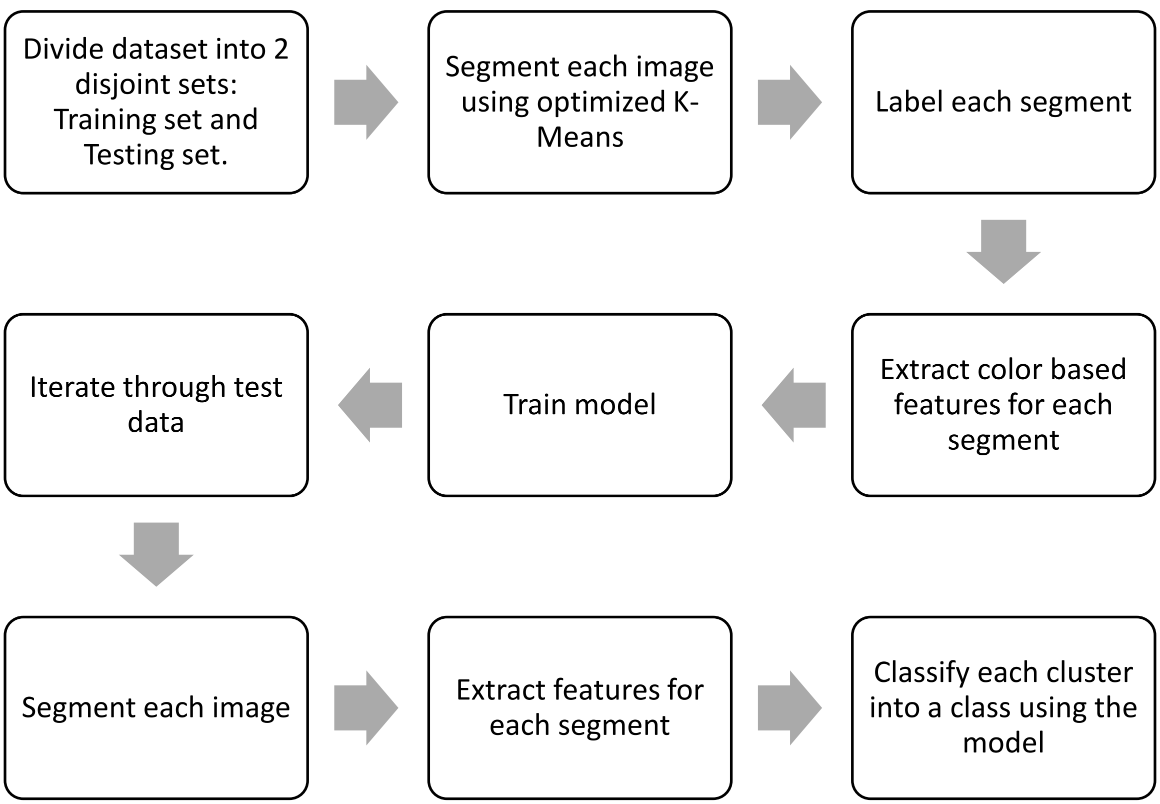


SVM offers flexibility in terms of choosing the kernel function. It can be used with a linear kernel function, which will create a hyperplane separating the data points. However, it can also be used with non-linear kernel functions, which create a hyperbola separator.



The process of training the dataset (in our problem statement) is as follows:

1. Separate data into 2 parts, training set and testing set.
2. Iterate through the images in the training set.
3. For each image, segment using optimized K-Means on LAB.
4. Then label each segment as diseased or non-diseased. For a multi class problem, where one wishes to build a model for identifying the type of disease, label each segment accordingly.
5. Now for each segment, extract the color based features (Mean of HSV).
6. Now train your model with the features generated and the labels on the test data only. While training, choose the hyperparameters and the kernel function as appropriate.
7. Next, iterate through the testing set.
8. For each image in the testing set, segment it using the same technique as used before.
9. Now for each segment, extract the same set of features, as done for the training set.
10. Now use the model built on the training set, and classify the data points into different classes.



Results:

The confusion matrix is as follows:

Actual\Predicted	Normal	Rot	Scab
Normal	54	4	1
Rot	12	57	0
Scab	7	4	14

The blotch class does not have good enough features and overlaps with rot and scab. Thus, it has been excluded from the final classification.

The accuracy for this multiclass model is **81.69%**.

Analysis:

If we analyze the multiclass model, with scab, rot and normal classes, the accuracy is 81.69%. There are a couple of reasons why the model is not as accurate: there are a few overlaps in the diseases, and not a clear distinction sometimes. On doing some literature survey, it was found that the diseases can occur together, and the visual signs are often legions. For these legions to clearly depict the correct disease, it sometimes takes some time to mature.

9 Future Work

We have studied the diseases in apples. However, there is huge scope to extend this study to other fruits, vegetables and grains. There are several other texture analysis tools, which may give better results for other products. Gas sensors can also be added as features, which will increase the quality of information that can be extracted and used to build a much more robust model.

10 Conclusion

As evident from the results, the hybrid optimization algorithm performs better in F3, F4, F5, F7, F8, F9 and F11 by reaching a more optimal value. However, the values are comparable in other functions except F1, F2 and F12. However, as seen in the graphs, the convergence is much faster in those cases, where the values are comparable. Therefore, our algorithm shows considerable improvement over MFO alone.

The newly proposed segmentation algorithms, optimized K-means and multi level thresholding using the developed hybrid algorithm has performs well. The segmentation techniques are not problem specific, and can find its application in any field of computer vision. The optimization algorithm improves upon the results of the segmentation for K-Means, and automates the threshold selection in Multi-Level Thresholding.

11 References

[1] - Nanda, Satyasai Jagannath, and Ganapati Panda. "A Survey on Nature Inspired Metaheuristic Algorithms for Partitional Clustering." Swarm and Evolutionary Computation 16 (2014): 1-18. Web.

[2] - Hatamlou, Abdolreza, Salwani Abdullah, and Hossein Nezamabadi-Pour. "Application of Gravitational Search Algorithm on Data Clustering." Rough Sets and Knowledge Technology Lecture Notes in Computer Science (2011): 337-46. Web.

[3] - Hatamlou, Abdolreza, Salwani Abdullah, and Hossein Nezamabadi-Pour. "A Combined Approach for Clustering Based on K-means and Gravitational Search Algorithms." Swarm and Evolutionary Computation 6 (2012): 47-52. Web.

[4] - Osuna-Enciso, Valentı´n, Erik Cuevas, and Humberto Sossa. "A Comparison of Nature Inspired Algorithms for Multi-threshold Image Segmentation." Expert Systems with Applications 40.4 (2013): 1213-219. Web.

[5] - Bülbül, Gonca, Akhtar Hayat, and Silvana Andreescu. "Portable Nanoparticle-Based Sensors for Food Safety Assessment." Sensors 15.12 (2015): 30736-0758. Web.

[6] - Duncan, Timothy V. "Applications of Nanotechnology in Food Packaging and Food Safety: Barrier Materials, Antimicrobials and Sensors." Journal of Colloid and Interface Science 363.1 (2011): 1-24. Web.

[7] - Yang, Tao, Huifen Huang, Fang Zhu, Qinlu Lin, Lin Zhang, and Junwen Liu. "Recent Progresses in Nanobiosensing for Food Safety Analysis." Sensors 16.7 (2016): 1118. Web.

[8] - Mutlu, Mehmet. Biosensors in Food Processing, Safety, and Quality Control. Boca Raton: CRC, 2011. Print.

[9] - Ropodi, A.i., E.z. Panagou, and G.-J.e. Nychas. "Data Mining Derived from Food Analyses Using Non-invasive/non-destructive Analytical Techniques; Determination of Food Authenticity, Quality & Safety in Tandem with Computer Science Disciplines." Trends in Food Science & Technology 50 (2016): 11-25. Web.

[10] - Brosnan, Tadhg, and Da-Wen Sun. "Improving Quality Inspection of Food Products by Computer Vision—a Review." Journal of Food Engineering 61.1 (2004): 3-16. Web.

[11] - Abdullah, Mohd Zaid, Sabina Abdul Aziz, and Abdul Manan Mohamed. "Quality Inspection Of Bakery Products Using A Color-Based Machine Vision System." Journal of Food Quality J Food Quality 23.1 (2000): 39-50. Web.

[12] - Du, Cheng-Jin, and Da-Wen Sun. "Learning Techniques Used in Computer Vision for Food Quality Evaluation: A Review." Journal of Food Engineering 72.1 (2006): 39-55. Web.

[13] - Ma, Ji, Da-Wen Sun, Jia-Huan Qu, Dan Liu, Hongbin Pu, Wen-Hong Gao, and Xin-An Zeng. "Applications of Computer Vision for Assessing Quality of Agri-food Products: A Review of Recent Research Advances." Critical Reviews in Food Science and Nutrition 56.1 (2014): 113-27. Web.

[14] - Ding, Zhuqing, Ruoyu Zhang, and Za Kan. "Quality and Safety Inspection of Food and Agricultural Products by LabVIEW IMAQ Vision." Food Analytical Methods Food Anal. Methods 8.2 (2014): 290-301. Web.

[15] - Miraglia, M., H.j.p. Marvin, G.a. Kleter, P. Battilani, C. Brera, E. Coni, F. Cubadda, L. Croci, B. De Santis, S. Dekkers, L. Filippi, R.w.a. Hutjes, M.y. Noordam, M. Pisante, G. Piva, A. Prandini, L. Toti, G.j. Van Den Born, and A. Vespermann. "Climate Change and Food Safety: An Emerging Issue with Special Focus on Europe." Food and Chemical Toxicology 47.5 (2009): 1009-021. Web.

[16] - "In Brief: Achieving a Transparent, Actionable Framework for Public–Private Partnerships for Food and Nutrition Research." Journal of the Academy of Nutrition and Dietetics 115.8 (2015): 1318-319. Web.

[17] - Goel, Lavika, Daya Gupta, V. K. Panchal, and Ajith Abraham. "Taxonomy of Nature Inspired Computational Intelligence: A Remote Sensing Perspective." 2012 Fourth World Congress on Nature and Biologically Inspired Computing (NaBIC) (2012): n. pag. Web.

[18] - Babak E. Bejnordi ; Geert Litjens ; Meyke Hermsen ; Nico Karssemeijer and Jeroen A. W. M. van der Laak. " A multi-scale superpixel classification approach to the detection of regions of interest in whole slide histopathology images ", *Proc. SPIE* 9420, Medical Imaging 2015: Digital Pathology, 94200H (March 19, 2015);

[19] - Hong-Min Zhu and Chi-Man Pun. "An Adaptive Superpixel Based Hand Gesture Tracking and Recognition System"

[20] Zhao, Mingru, Hengliang Tang, Jian Guo, and Yuan Sun. "Data Clustering Using Particle Swarm Optimization." Lecture Notes in Electrical Engineering Future Information Technology (2014): 607-12. Web.

[21] - Li, Haiyang, Hongzhou He, and Yongge Wen. "Dynamic Particle Swarm Optimization and K-means Clustering Algorithm for Image Segmentation." Optik - International Journal for Light and Electron Optics 126.24 (2015): 4817-822. Web.

[22] - Koonsanit, Kitti. "Determination of the Initialization Number of Clusters InK-means Clustering Application Using Co-OccurrenceStatistics Techniques for Multispectral Satellite Imagery." International Journal of Information and Electronics Engineering IJIEE (2012): n. pag. Web.

[23] - Kumar, Tara Sai, Mahesh Chandra M, and Sreenivasa Murthy P. "Colour Based Image Segmentation Using Fuzzy C-Means Clustering." Int. Jour. Intel. Elec. Sys. International Journal on Intelligent Electronic Systems 5.2 (2011): 47-51. Web.

[24] - Dubey, Shiv Ram, and Anand Singh Jalal. "Adapted Approach for Fruit Disease Identification Using Images." International Journal of Computer Vision and Image Processing 2.3 (2012): 44-58. Web.

[25] - Zhang, Y.j. "A Survey on Evaluation Methods for Image Segmentation." Pattern Recognition 29.8 (1996): 1335-346. Web.

[27] - Ahmadyfard, Alireza, and Hamidreza Modares. "Combining PSO and K-means to Enhance Data Clustering." 2008 International Symposium on Telecommunications (2008): n. pag. Web.

[30] - Narwade, Jyoti, and Binod Kumar. "Local and Global Color Histogram Feature for Color Content-Based Image Retrieval System." Proceedings of the International Congress on Information and Communication Technology Advances in Intelligent Systems and Computing (2016): 293-300. Web.

[31] - Huang, Di, Caifeng Shan, Mohsen Ardabilian, Yunhong Wang, and Liming Chen. "Local Binary Patterns and Its Application to Facial Image Analysis: A Survey." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 41.6 (2011): 765-81. Web.

[32] - Szeliski, Richard. Computer Vision: Algorithms and Applications. London: Springer, 2011. Print.