CARDIAC THREAT PREDICTION BY SEGMENTING IVUS IMAGES USING FCM ALGORITHM

A Main Project Report submitted in partial fulfillment of the

requirements for the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted By

K. S. V. KARTHIKEYA	**(18A91A0531)**
ANURADHA SINGH	**(18A91A0554)**
B. CHAKRI	**(18A91A0513)**
B. NITHIN SAI	**(18A91A0506)**

Under the esteemed guidance of

Mr. V. RAVI KISHORE, M. Tech, (Ph.D.)
Associate Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ADITYA ENGINEERING COLLEGE (A)

Approved by AICTE, Permanently Affiliated to JNTUK & Accredited by NAAC with 'A' Grade
Recognized by UGC under the sections 2(f) and 12(B) of the UGC act 1956
Aditya Nagar, ADB Road - Surampalem – 533437, E.G.Dist., A.P.,

2021-2022

ADITYA ENGINEERING COLLEGE (A)

Approved by AICTE, Permanently Affiliated to JNTUK & Accredited by NAAC with 'A' Grade
Recognized by UGC under the sections 2(f) and 12(B) of the UGC act 1956
Aditya Nagar, ADB Road - Surampalem – 533437, E.G.Dist., A.P.,
2021-2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project work entitled *"***CARDIAC THREAT PREDICTION BY SEGMENTING IVUS IMAGES USING FCM ALGORITHM***"* is being submitted by

K. S. V. KARTHIKEYA	**(18A91A0531)**
ANURADHA SINGH	**(18A91A0554)**
B. CHAKRI	**(18A91A0513)**
B. NITHIN SAI	**(18A91A0506)**

In partial fulfillment of the requirements for award of the B.Tech degree in Computer Science and Engineering for the academic year 2021-2022.

Project Guide
Mr. V. Ravi Kishore, M. Tech, (Ph.D.)
Associate Professor
Department of CSE

Head of the Department
Prof. A. Vanathi, M.E.,(Ph.D)
Department of CSE

External Examiner

DECLARATION

We hereby declare that the project entitled **"CARDIAC THREAT PREDICTION BY SEGMENTING IVUS IMAGES USING FCM ALGORITHM"** is a genuine project. This work has been submitted to the **ADITYA ENGINEERING COLLEGE,** Surampalem, permanently affiliated to **JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, KAKINADA** in partial fulfillment of the **B.Tech** degree**.** We further declare that this project work has not been submitted in full or part of the award of any degree of this or any other educational institutions.

by

K. S. V. KARTHIKEYA	**(18A91A0531)**
ANURADHA SINGH	**(18A91A0554)**
B. CHAKRI	**(18A91A0513)**
B. NITHIN SAI	**(18A91A0506)**

ACKNOWLEDGEMENT

It is with immense pleasure that we would like to express our indebted gratitude to our project supervisor **Mr. V. Ravi Kishore, M.Tech. (Ph.D.), Associate Professor,** who has guided us a lot and encouraged us in every step of the project work, his valuable moral support and guidance throughout the project helped us to a greater extent.

We wish to thank our Project Coordinator **Dr. S. Prasanth Vaidya**, **Associate Professor** in CSE department for his support and valuable suggestions for the successful completion of this project.

Our deepest thanks to our HOD **Mrs. A. Vanathi, Associate Professor** for inspiring us all the way and for arranging all the facilities and resources needed for our project.

We wish to thank **Dr. S. Rama Sree, Professor** in CSE and Dean (Academics) for her support and suggestions during our project work.

We owe our sincere gratitude to **Dr. M. Sreenivasa Reddy, Principal** for providing a great support and for giving us the opportunity of doing the project.

We are thankful to our **College Management** for providing all the facilities in time to us for completion of our project.

Not to forget, **Faculty, Lab Technicians, non-teaching staff and our friends** who have directly or indirectly helped and supported us in completing our project in time.

ABSTRACT

Cardiac is a medical term which refers to Heart. It is the main part and Cardiac arrest is the dangerous threat that causes a halt or block in Coronary artery that stops blood from reaching heart. There can be various reasons for this problem and can be known only via Intra Vascular Ultra Sound (IVUS) images.

These IVUS images are not understandable by non-medical person and require some analysis. This segmentation of IVUS image will help automate all the analysis process by segmenting/classifying the image and analyzing the major parameters that are involved. Basing on results, it produces a report on whether a threat for heart exists or not and calculates how complicated that threat can be using threat score.

By using Image processing techniques, we can achieve this by transforming our fuzzy image to a clear-cut machine understandable image. Then some mathematical formulae and algorithms help us process the image and produce a report.

We need to classify/segment the image based on some properties based on medical terms. Medical image classification is a two-step process. Initially feature extraction techniques are used to obtain visual features from image data and second step is to use machine intelligence algorithms that use these features and classify images into defined groups or classes.

Among different features that have been used, shape, edge and other global texture features were commonly trusted ones. These reports can be linked to other technologies and projects like alarms, auto messaging systems, so that it becomes more effective.

INDEX   Page

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Introduction to Image Processing

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too.

Image processing basically includes the following three steps:

- Importing the image via image acquisition tools;
- Analyzing and manipulating the image;
- Output in which result can be altered image or report that is based on image analysis.

There are two types of methods used for image processing namely, analogue and digital image processing. Analogue image processing can be used for the hard copies like printouts and photographs. Image analysts use various fundamentals of interpretation while using these visual techniques. Digital image processing techniques help in manipulation of the digital images by using computers. The three general phases that all types of data have to undergo while using digital technique are pre-processing, enhancement, and display, information extraction.

By using Image processing techniques, we can achieve this by transforming our fuzzy image to a clear-cut machine understandable image. Then some mathematical formulae and algorithms help us process the image and produce a report.

We need to classify/segment the image based on some properties based on medical terms. Medical image classification is a two-venture process. At first feature extraction strategies are utilized to get visual features from image information and second step is to utilize machine intelligence algorithms that utilization these features and arrange images into characterized groups or classes. Among various features that have been utilized, shape, edge and other worldwide texture features were regularly trusted ones. These reports can be linked to other technologies and projects like alarms, auto messaging systems, so that it becomes more effective

1.2 Existing System and Disadvantages

Image processing is a way by which an individual can enhance the quality of an image or gather alerting insights from an image and feed it to an algorithm to predict the later things. Many domains are using Image classification techniques. It produces the final stage where the result can be changed to an image or report based on image analysis. There is no past situation where an IVUS image is analyzed using Image Classification. Until now using some algorithms IVUS image is classified and left as a segmented image. This may show the evolution of Image classification technology but it is not quite useful in the real life. If the segmentation is put to use then it may help automate the process and provide suggestions to user without any doctor's involvement.

Disadvantages:

- It requires medical knowledge
- No classification of IVUS before
- FCM is not used for IVUS of blood vessels
- No direct suggestions to end user
- Usual Image processing techniques depends on edges which is quite different in case of IVUS

1.3 Proposed System and Advantages

We propose a method where the system takes an IVUS image as an input and produces a report and also evaluate threat score and display it to the user automatically. This is done by classifying the IVUS image as discussed in the base paper using some classification techniques. As shown in the architecture, the image is non fuzzified to get clear and accurate information and the threat score is decided as per the medical information that is shown in record. Depending upon the intensity of the report which is generated at the end of classification Threat score is calculated using some formulae or any algorithms.

Fuzzy c-means (FCM) clustering algorithm is most commonly used unsupervised clustering technique in the field of medical imaging. Medical image segmentation alludes to the segmentation of medical images using its anatomic structures. Fuzzy C-means (FCM) is a strategy for clustering which permits one piece of information to have a place with at least two clusters. FCM is prominently utilized for delicate segmentations like mind tissue model. And furthermore FCM can give improved outcomes than other clustering algorithms such as KMeans, EM, and KNN.

This method is adaptive as some extensions of hardware or other software components like sending alerts, raising alarms can be added.

Advantages

- No medical knowledge required
- IVUS is classified according to medical theories
- FCM takes advantage of noise
- It is extendable and adaptable
- It improves technology used in medical departments

1.4 Objectives of the Project

High level Objectives

- Main goal of this method is to prevent Cardiac arrest or atherosclerotic diseases.
- Automate IVUS analysis to predict quickly and independently.
- Make sure entire process is automated
- Give user both medical and non medical outputs

Low level Objectives

- Defuzzify (remove noise) the IVUS image which is taken as input.
- Compute FCM algorithm
- Analyze Histogram of images
- Classify the image into different segments like lumen, plaque..etc. using different classification techniques.
- Calculate area of the segments and generate a report.
- Analyze histograms and area of colors
- Evaluate Threat score.

1.5 Organization of the Project

In this, we are going to discuss about the organization of this project, that is how we organized and presented the all-detailed information about the project in a Proper manner. And the total project in this document is divided into 11 chapters and now, we are going to discuss about it as follows.

Chapter 1: In this, we have introduced about Image Processing and its domains and some advantages along with some process of its working.

Chapter 2: In this, we discussed about the necessary requirements we used in building this project along with its specific features and uses.

Chapter 3: In this, we discussed about all the necessary research papers on IVUS and FCM we studied in the literature survey for developing this project.

Chapter 4: Here, we come to know about different modules that are used and divided in our project.

Chapter 5: Architecture diagram and its explanation is discussed here along with the appropriate unified model diagrams to represent the project.

Chapter 6: This Chapter shows how our project is implemented and its Source code

Chapter 7: The various possible test cases that are possible for this are generated and displayed in this chapter.

Chapter 8: Here, the various output that are generated are displayed with proof i.e., along with appropriate screenshots and images regarding the working of the project.

Chapter 9: In this, we conclude by once again giving a quick summary of the working of this project along with some details about its future scope.

Chapter 10: Finally, all the references are mentioned here that means the books and papers that we have done research on for developing this project.

2. REQUIREMENTS ANALYSIS

2.1 Introduction

Requirements analysis or requirements engineering is a process used to determine the needs and expectations of a new product. It involves frequent communication with the stakeholders and end-users of the product to define expectations, resolve conflicts, and document all the key requirements

Step 1. Identify End Users

Every person with Internet and Python Software can use this project

It is useful mostly for Non-medical persons

Step 2. Capture Requirements

Ask each of the stakeholders and end-users their requirements for the new product. Here are some requirements analysis techniques that you can use to capture requirements:

1. Utilize Use Cases

Use cases provide a walkthrough of the entire product through the eyes of the end-user. This technique will help visualize how the product will actually work.

2. Build Prototypes

A prototype provides users a sample look and feel of the final product. This technique will help address feasibility issues and identify problems ahead of time.

Step 3. Types of requirements

Since requirements can be of various types, they should be grouped to avoid confusion. Requirements are usually divided into four categories:

- Functional Requirements - Functions the product is required to perform.
- Technical Requirements - Technical issues to be considered for the successful implementation of the product.
- Transitional Requirements - Steps required to implement a new product smoothly.
- Operational Requirements - Operations to be carried out in the backend for proper functioning of the product.

2.2 Hardware & Software Requirements

Hardware Requirements

Minimum

- Windows 10, 64 bits
- Any CPU (Intel i3/i5/ i7/ Ryzen 7).
- Projects (under 100 images at 14 MP): 4 GB RAM, 10 GB HDD Free Space.

Recommended

- Windows 10, 64 bits
- CPU quad-core or hexa-core Intel i5/Threadripper/Ryzen 9/.
- Hard disk: SSD.
- Projects (under 100 images at 14 MP): 8 GB RAM, 15 GB SSD Free Space.

Software Requirements

- Python
  - Packages required:
    - Os
    - Argparse
    - OpenCv
    - Numpy
    - MatplotLib
    - extColors
- Python IDLE(Spyder/PyCharm)
- Color Supported Terminal for Plots

2.3 Software Requirements Specification

1. Introduction:

- (i) Purpose of this Document –
  This page describes the requirements of end User
- (ii) Scope of this document –
  Scope is restricted to Users holding a Device supporting Python and version which supports required packages

2. Functional Requirements:

System need to provide Threat Score out of 100 as an output
Segmented image is plotted as a debugging output
Color area percentage need to be printed as part of side output
Algorithm's Iteration and its cost can be printed for testing purpose and optional

3. Interface Requirements:

IDLE is required to present output
Command terminal is required to run script and provide default arguments

4. Performance Requirements:

System should be Error free
Any exception needs to be handled and should not be present in output
Memory limit is restricted to 100 Mb of HDD
Time limit of execution is restricted to 8 sec

5. Non-Functional Attributes:

Security
Portability
Data integrity
Scalability

Medical Record Table:

Table 2.1 Medical Record for IVUS Image

	Meaning	Good	Bad
Perivascular	Outer structure of the tube of blood vessel	Perfect circular in shape	Cuts and deformed structure
Plaque	Extra fluid and debris formed inside tube	Very less and no plaque is formed for ideal person	More plaque area
Lumen	Space inside vessel for the flow of blood	Should be spacious and area must be more without contractions	Contracted between plaque and perivascular

3. LITERATURE SURVEY

Image recognition technology based on neural network.

Artificial neural networks are nothing but an arrangement of interconnected handling components, called nodes, which are practically similar to biological neurons. For instance, assuming the organization is given an assignment to recognize a face, the main hidden layer could go about as a line detector as shown in Fig 3.1, the second hidden accepts these lines as information to assembles them to frame a nose, the third hidden layer takes the nose and coordinates it with an eye, etc, until at end the entire face is constructed [2].

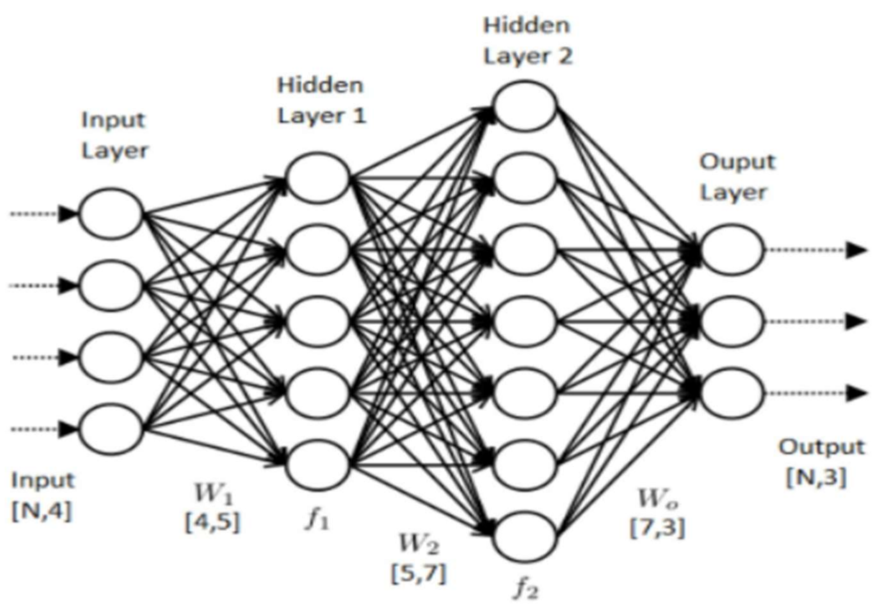


Fig 3.1 ANN Architecture

An improved k-nearest neighbor algorithm and its application to high resolution

K-Nearest Neighbor is a non-parametric technique which is utilized for classification and regression. It is by a wide margin the most straightforward calculation. It is a lazy learning calculation, where the capacity is just approximated locally and all calculation is conceded until work assessment. This calculation just depends on the distance between feature vectors and segments unknown data points by tracking down the most common class among the k-nearest model[3].

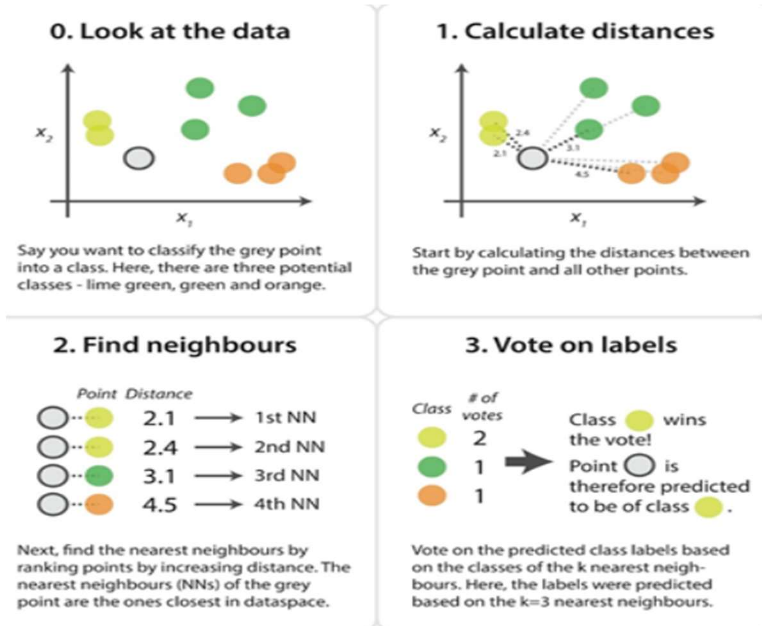


Fig 3.2 KNN Procedure

Image classification via support vector machine

Support vector machines algorithm is very famous as a result of their capacity to deal with numerous continuous and categorical factors. The objective is to isolate the datasets into classes to find a greatest peripheral hyperplane as shown in Fig 3.3. It constructs a hyper-plane or a bunch of hyper-planes in a high dimensional space and great separation between the two classes is accomplished by the hyperplane that has the large distance to the closest training data point of any class[4].

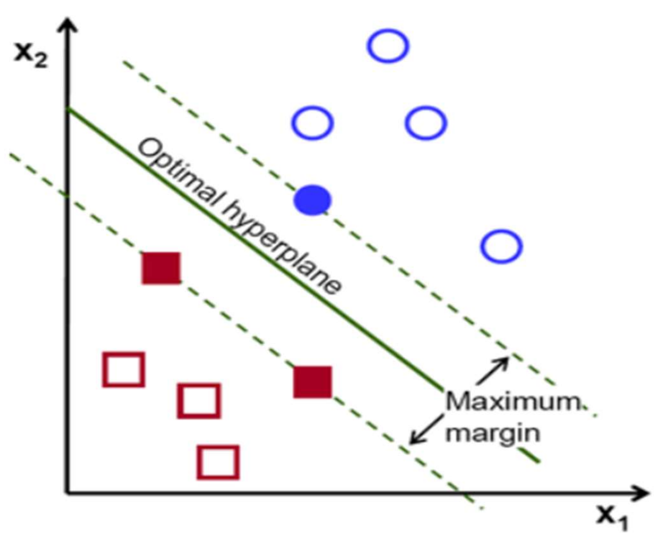


Fig 3.3 SVM Architecture of Hyperplane

Image classification using random forests and ferns.

A random forest is a group of decision trees. Random forests use training data to learn and to make predictions. The fixing utilized here is Shape and Color. Prediction accuracy on complex issues is usually mediocre compared to gradient-boosted trees[5].

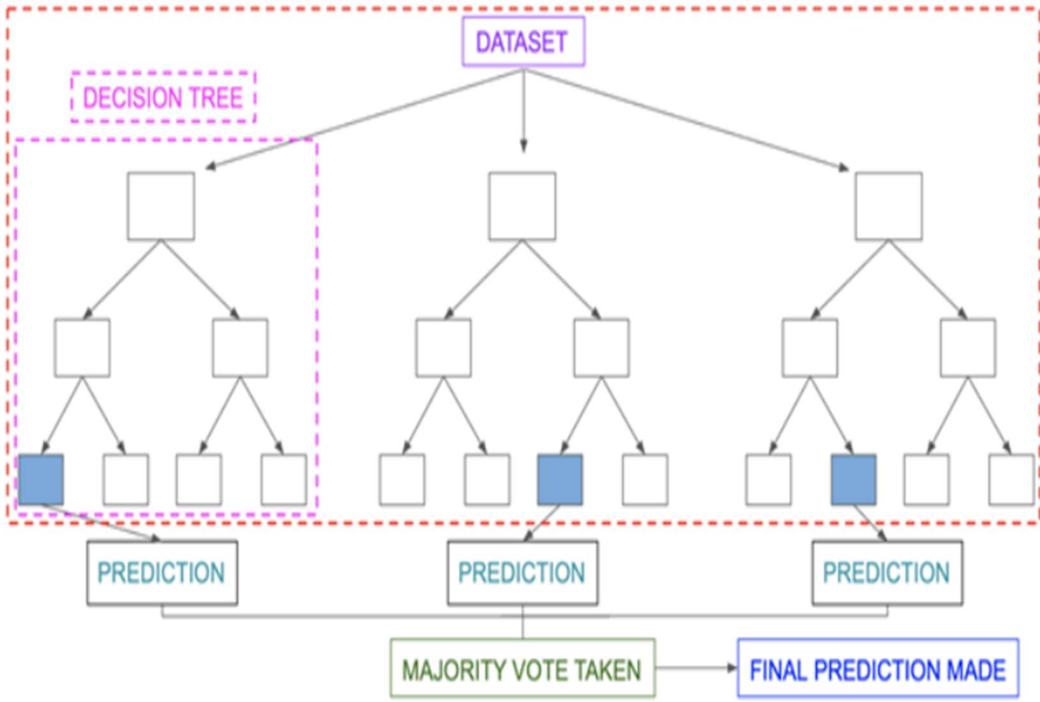


Fig 3.4 Random Forest Classification

Fuzzy c-means algorithm for medical image segmentation

Fuzzy c-means (FCM) clustering algorithm is most commonly used unsupervised clustering technique in the field of medical imaging. Medical image segmentation alludes to the segmentation of medical images using its anatomic structures. Fuzzy C-means (FCM) is a strategy for clustering which permits one piece of information to have a place with at least two clusters. FCM is prominently utilized for delicate segmentations like mind tissue model. And furthermore, FCM can give improved outcomes than other clustering algorithms such as KMeans, EM, and KNN[6].

A generalized spatial fuzzy c-means algorithm for medical image segmentation

Fuzzy clustering is an appropriate method in medical image segmentation. Its applications are very successful in the area of image processing as well as medical imaging. The field of medicine has become a very attractive domain for the application of fuzzy set theory. One basic method of fuzzy clustering is based on fuzzy c-means clustering.[9]

Single color extraction and image query

One common method for characterizing image content is to use color histograms. The color histogram for an image is constructed by counting the number of pixels of each color. Retrieval from image databases using color histograms has been investigated. In these studies the formulations of the retrieval algorithms follow a similar progression: (1) selection of a color space, (2) quantization of the color space, (3) computation of histograms, (4) derivation of the histogram distance function, (5) identification of indexing shortcuts. Each of these steps may be crucial towards developing a successful algorithm. But there has been no consensus about what are the best choices for these parameters.

Comparison Table of Literature Survey

Table 3.1 Comparison of Different Algorithms

Classifier Used	Features required	Data Sets	Accuracy	Match?	Draw Backs
ANN[2]	Size, Length, Edges	NA	90%	NO	IVUS doesn't have edges
SVM[3]	NA	2615	99%	NO	SVM uses high differentiated pixels which IVUS images does not behold
FCM	Texture and Morphologic	NA	90%	YES	NA
KNN[5]	Color	100	100%	NO	KNN is color dependent. IVUS is not a colorful image
Random Forest[4]	Shape and Color	178	NA	NO	RF is Data Set dependent

4. MODULES

4.1 Introduction

In Python, Modules are simply files with the “. py” extension containing Python code that can be imported inside another Python Program. In simple terms, we can consider a module to be the same as a code library or a file that contains a set of functions that you want to include in your application

Inbuilt modules used in our project are:

- Os
- Argparse
- OpenCv
- Numpy
- MatplotLib
- extColors

Programmer defined modules are:

MODULES

1.main

2.FCM

3.ColorAreaExtractor

4.ThreatScorePredictor

4.2 Description of Modules

Inbuilt modules are used to reduce complexity of the project

1. OS: Used to manage paths and directories
2. Argparse: To give default arguments from the terminal
3. OpenCV: Image processing
4. Matplotlib: To plot histograms and images
5. Extcolors: To extract colors

User defined Modules

1.Main: Main module contains the initial code that can take input an IVUS image and controls all other modules and their order of execution.

2.FCM: FCM module is the heart of this project. It contains the complex Fuzzy c means algorithm to classify the image into different segments. It contains methods like “get_filtered image”, “form_clusters”, “CalculateHistogram” and “segment_image”.

- Get filtered image: All images from internet or source does not look similar so It filters the image by converting into compatible image format by adjusting scales.
- Calculate Histogram method will provide a histogram representation of colors vs pixels of an image
- Form clusters contains main algorithm depending upon given number of iterations and fuzzy parameter it segments into different colors
- Segment Image segments the image to different colors by changing 2d matrix to pixels

3.ColorAreaExtractor: It uses “extcolors” module and extract percentage area of the color in the image.It will inform about the parts of Blood vessel and their concentration.

4.ThreatScorePredictor: It has an manual algorithm which is provided so that it uses color percentage to evaluate Threat score. It uses 100 as a limit value. Value near to 100 means threat is high and intense and vice versa.

5. SYSTEM DESIGN

5.1 Introduction

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

A systemic approach is required for a coherent and well-running system. Bottom-Up or Top-Down approach is required to take into account all related variables of the system. A designer uses the modelling languages to express the information and knowledge in a structure of system that is defined by a consistent set of rules and definitions. The designs can be defined in graphical or textual modelling languages.

Design methods:

1) Architectural design: To describes the views, models, behaviour, and structure of the system.

2) Logical design: To represent the data flow, inputs and outputs of the system. Example: ER Diagrams (Entity Relationship Diagrams).

3) Physical design:

Defined as

a) How users add information to the system and how the system represents information back to the user.

b) How the data is modelled and stored within the system.

c) How data moves through the system, how data is validated, secured and/or transformed as it flows through and out of the system.

5.2 Data Dictionary

Keyword	**Meaning**
Cardiac	Cardiac means relating to heart
IVUS	Intra Vascular Ultra Sound
Coronary Blood Vessel	Tube like structure allowing blood to flow
Plaque	Debris of calcium dust
Lumen	Area of hollowness inside blood vessel
Image Processing	It’s a technology used to analyze images
FCM	Fuzzy C means Algorithm
Fuzzy Parameter	Ranges between 1 and infinity
Epsilon	Euclidian distance between centers of clusters
Histogram	Graph showing color intensities with respect to pixels
Threat Score	Score out of 100 showing intensities of threat to a user
Color Extractor	Module which extracts color areas from an image

5.3 Logical Database Design

Logical database design is the process of transforming (or mapping) a conceptual schema of the application domain into a schema for the data model underlying a particular DBMS, such as the relational or object-oriented data model. This mapping can be understood as the result of trying to achieve two distinct sets of goals:

(i) representation goal: preserving the ability to capture and distinguish all valid states of the conceptual schema

(ii) data management goals: addressing issues related to the ease and cost of querying the logical schema, as well as costs of storage and constraint maintenance. This entry focuses mostly on the mapping of (Extended) Entity-Relationship (EER) diagrams to relational databases.

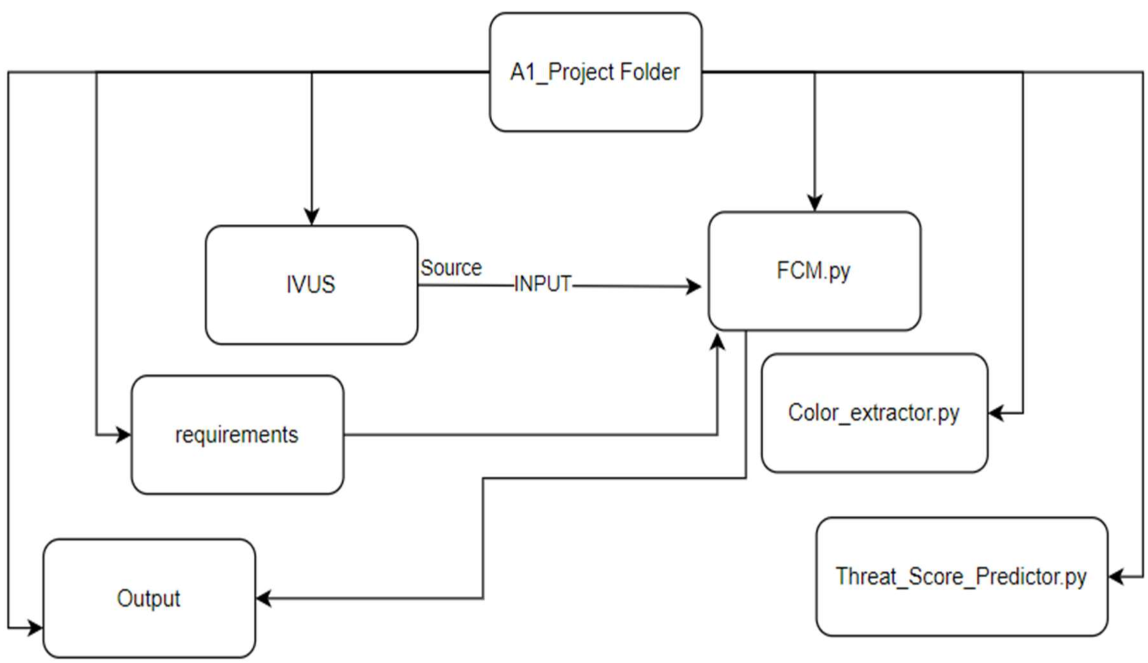


Fig 5.1 Logical Database design

Architecture Design

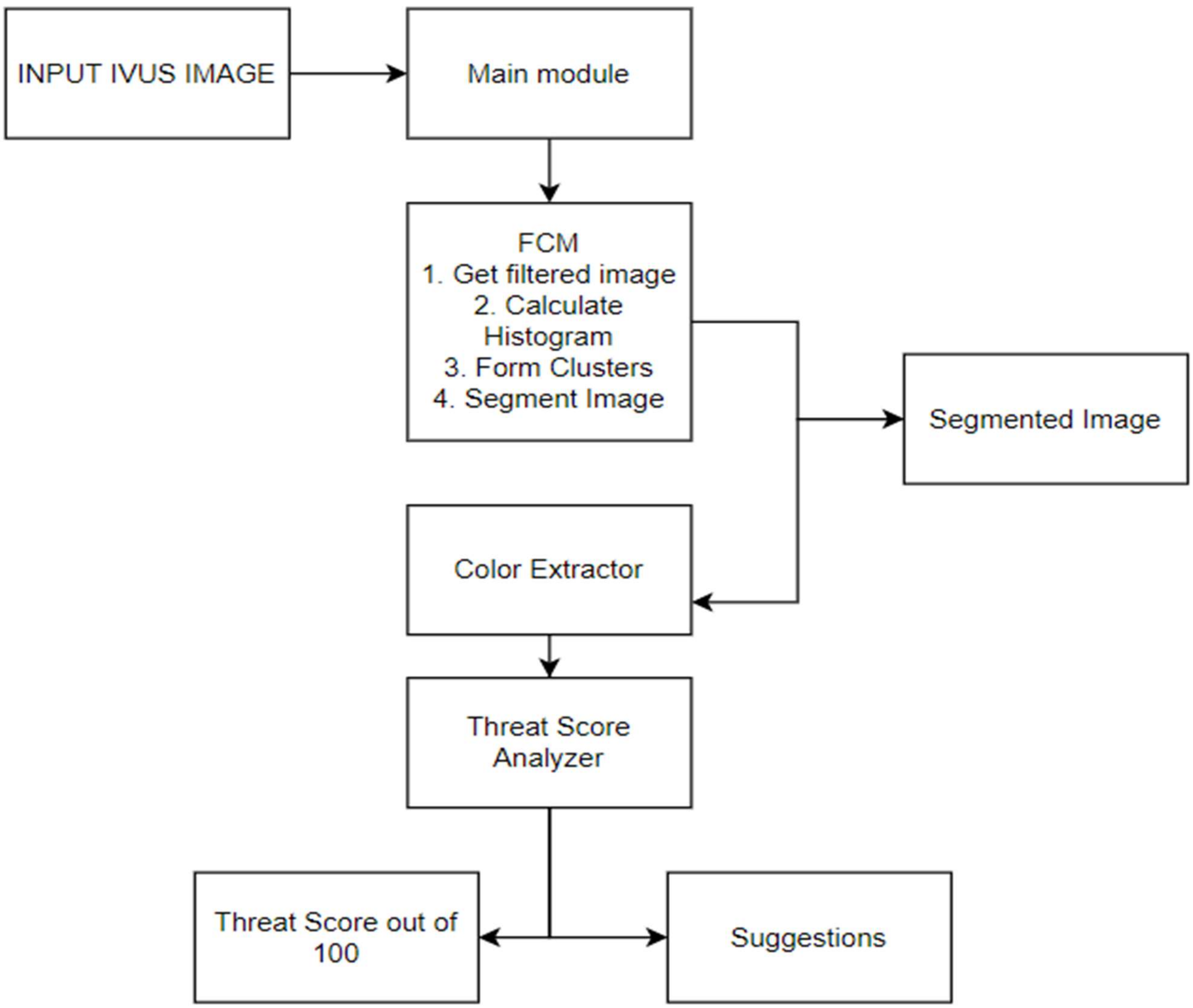
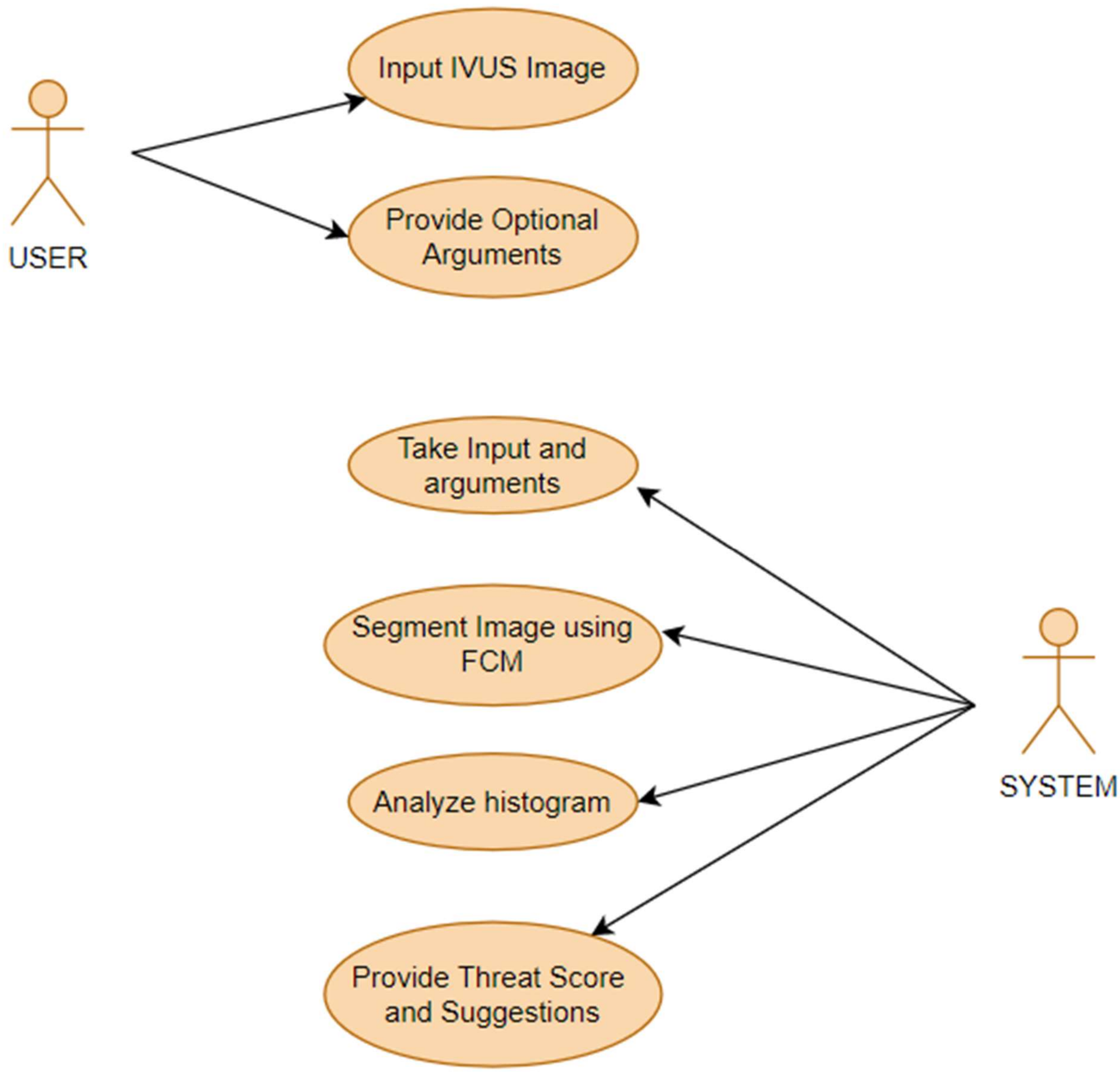


Fig 5.2 Architecture design

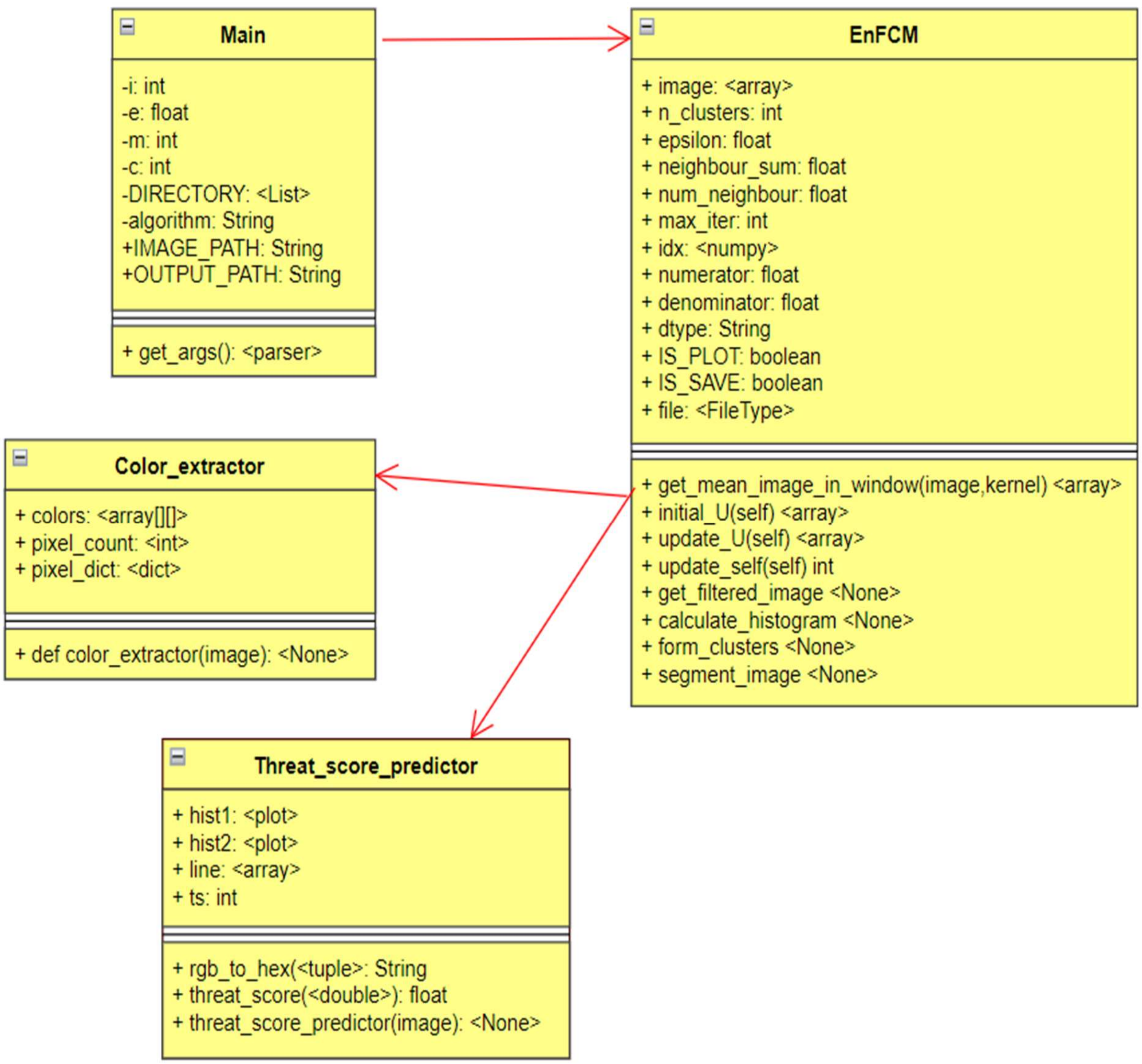
5.4 UML Diagrams

1. Use Case Diagram



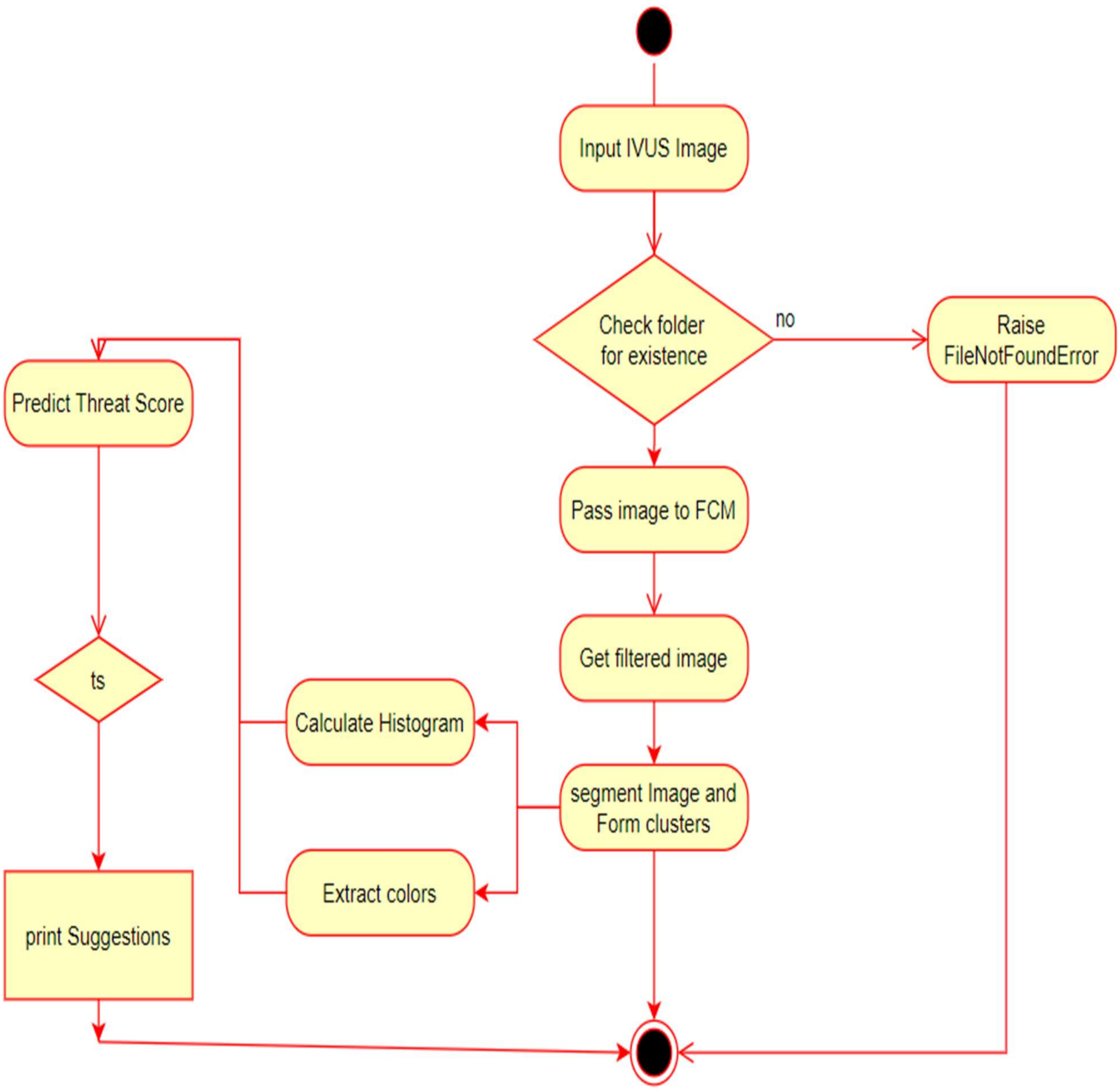
Use case relation between User and the System

2. Class Diagram



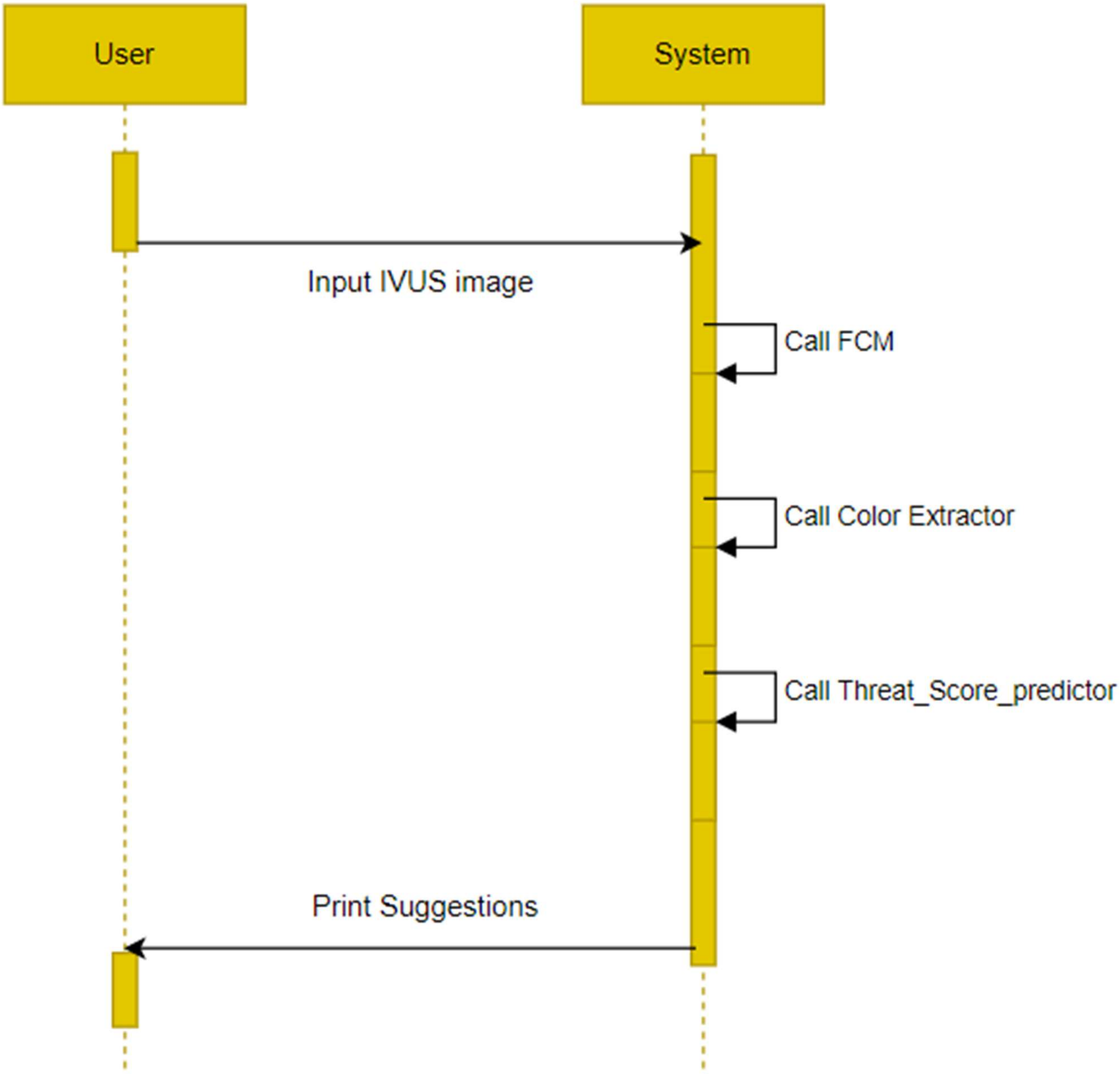
Association between Classes of this Project

3. Activity Diagram



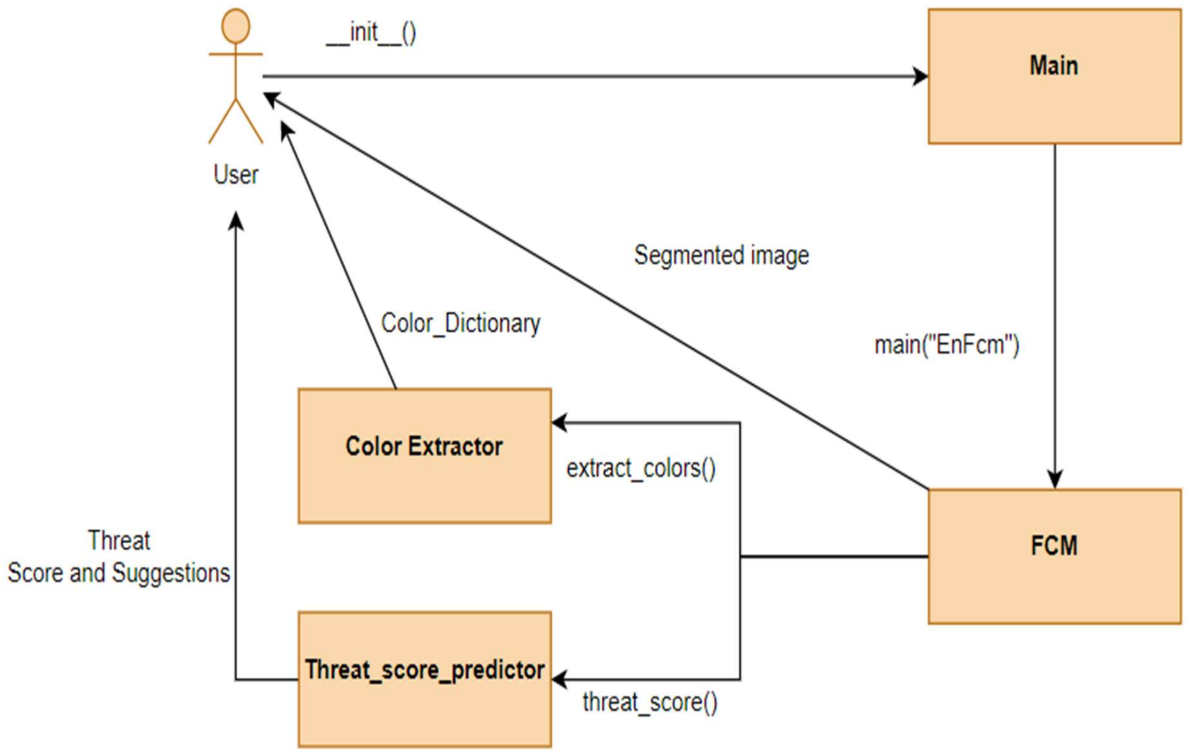
Activities of this Project

4. Sequence Diagram



Sequential actions of this Project

5.Collabaration Diagram



Collaboration between Modules

6. SYTEM IMPLEMENTATION

6.1 Introduction

The purpose of System Implementation can be summarized as follows: making the new system available to a prepared set of users (the deployment), and positioning on-going support and maintenance of the system within the Performing Organization (the transition). At a finer level of detail, deploying the system consists of executing all steps necessary to educate the Consumers on the use of the new system, placing the newly developed system into production, confirming that all data required at the start of operations is available and accurate, and validating that business functions that interact with the system are functioning properly. Transitioning the system support responsibilities involves changing from a system development to a system support and maintenance mode of operation, with ownership of the new system moving from the Project Team to the Performing Organization.

FCM algorithm classifies image by a four step process.

- Get filtered image: All images from internet or source does not look similar so It filters the image by converting into compatible image format by adjusting scales.
- Calculate Histogram method will provide a histogram representation of colors vs pixels of an image
- Form clusters contains main algorithm depending upon given number of iterations and fuzzy parameter it segments into different colors
- Segment Image segments the image to different colors by changing 2d matrix to pixels

6.2 Selected Software

- Anaconda Terminal
- Python IDLE (Spyder)

6.3 Sample Code

```
def get_args():
    parser = argparse.ArgumentParser(description="IVUS")
    parser.add_argument('--num_bit', default=8, type=int,
                        help="number of bits of input images")
    #-----------------Fundamental parameters-----------------
    parser.add_argument('-c', '--num_cluster', default='4', type=int,
                        help="Number of cluster")
    parser.add_argument('-m', '--fuzziness', default='2', type=int,
                        help="fuzziness degree")
    parser.add_argument('-i', '--max_iteration', default='100', type=int,
                        help="max number of iterations.")
    parser.add_argument('-e', '--epsilon', default='0.05', type=float,
                        help="threshold to check convergence.")
    #-----------------User parameters-----------------
    parser.add_argument('--plot_show', default=1, choices=[0,1],
                        help="Show plot about result")
    parser.add_argument('--plot_save', default=1, choices=[0,1],
                        help="Save plot about result")
     #-----------------Parametesr for MFCM/EnFCM-----------------
    parser.add_argument('-w', '--win_size', default='5', type=int,
                        help="Window size of MFCM/EnFCM algorithm")
    parser.add_argument('-n', '--neighbour_effect', default='3', type=float,
                        help="Effect factor of the graylevel which controls the influence extent of
neighbouring pixels.")
    args = parser.parse_args()
    return args
if __name__ == '__main__':

    args = get_args()
    algorithm = "EnFCM"
```

FCM:

```
def form_clusters(self):
    self.get_filtered_image()
    self.calculate_histogram()
    d = 100
    self.U = self.initial_U()
    if self.max_iter != -1:
        i = 0
        while True:
            self.C = self.update_C()
            old_u = np.copy(self.U)
            self.U = self.update_U()
            d = np.sum(abs(self.U - old_u))
            print("Iteration %d : cost = %f" %(i, d))
            if d < self.epsilon or i > self.max_iter:
                break
            i+=1
    else:
        i = 0
        while d > self.epsilon:
            self.C = self.update_C()
            old_u = np.copy(self.U)
            self.U = self.update_U()
            d = np.sum(abs(self.U - old_u))
            print("Iteration %d : cost = %f" %(i, d))

            if d < self.epsilon or i > self.max_iter:
                break
            i+=1
    self.segmentImage()
```

7. Testing

7.1 Introduction

The primary goal of software tests is to eliminate bugs in the code. However, there are additional benefits a project can gain from a good testing process. Benefits such as enhancing performance, user experience, and security of the overall project.

Often, when working on big projects, the team is divided into several sections. Each has its development task, and each task has its standalone functionality. These tasks are then combined to form the overall software product. That's why each part must undergo its own testing process to make sure it functions properly before it is added to the main project.

Types of Software Testing

There are many different types of software testing that you can use to make sure that your code is functioning as it should, and any changes to your code are not missing the entire functionality of the project.

Functional testing

Functional testing involves the testing of the functional aspects of a software application. When you're performing functional tests, you've got to check each functionality in your code. You will need to ensure that each section of the code is functioning correctly.

Non-functional testing

Non-functional testing targets the non-functional aspects of an application, like performance, reliability, usability, and security. Non-functional testing is aimed to improve the quality of your code. It isn't easy to perform this type of test manually.

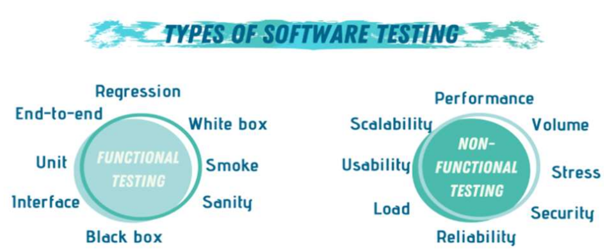


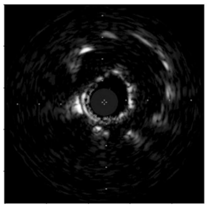
Fig 7.1 Types of Testing

7.2 Test Cases

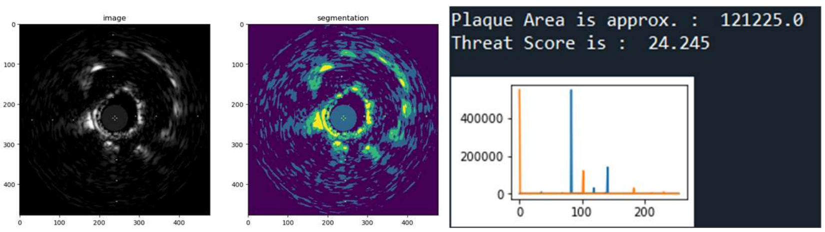
Test Id: 1

Description: Healthy Blood vessel IVUS image

Input:



Output:



Expected Threat Score: 0-30

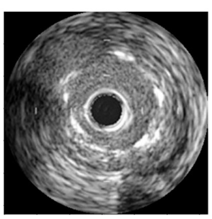
Actual Threat Score:24.245

Result: PASS

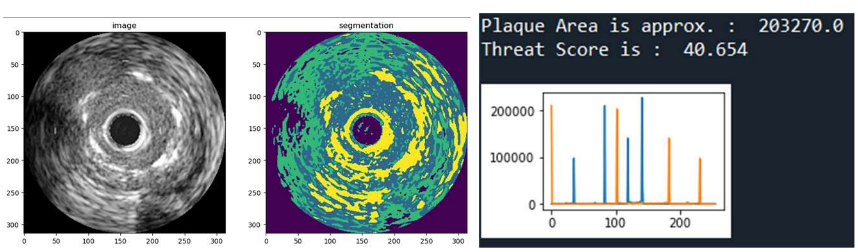
Test Id: 2

Description: Normal Blood vessel IVUS image

Input:



Output:



Expected Threat Score: 30-70

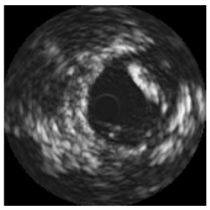
Actual Threat Score:41

Result: PASS

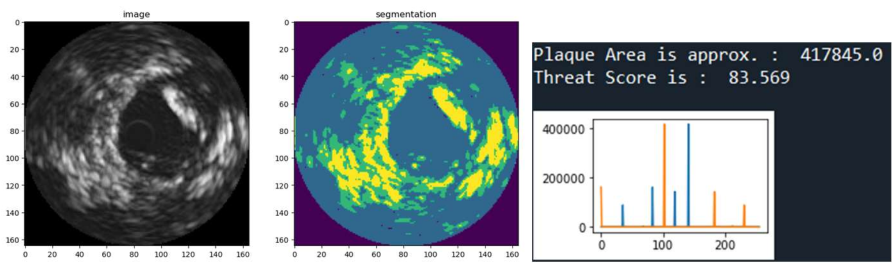
Test Id: 3

Description: Diseased Blood vessel IVUS image

Input:



Output:



Expected Threat Score: 70-100

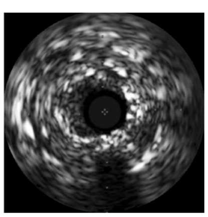
Actual Threat Score: 83.564

Result: PASS

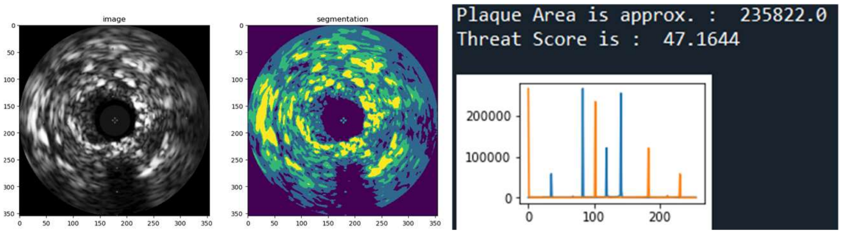
Test Id: 4

Description: Medium IVUS image

Input:



Output:



Expected Threat Score: 30-70

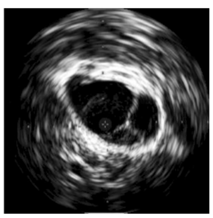
Actual Threat Score: 47.1644

Result: PASS

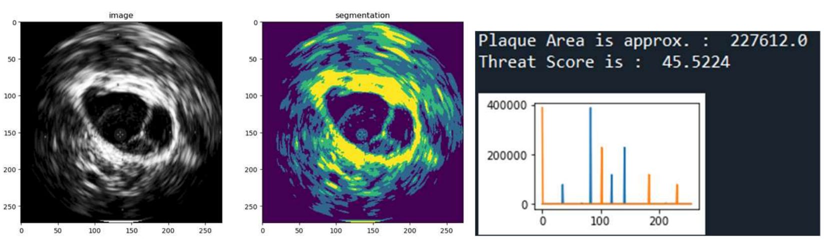
Test Id: 5

Description: Medium Blood vessel IVUS image

Input:



Output:



Expected Threat Score: 30-70

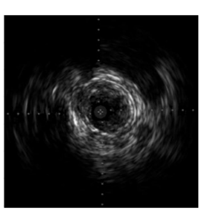
Actual Threat Score: 45.5224

Result: PASS

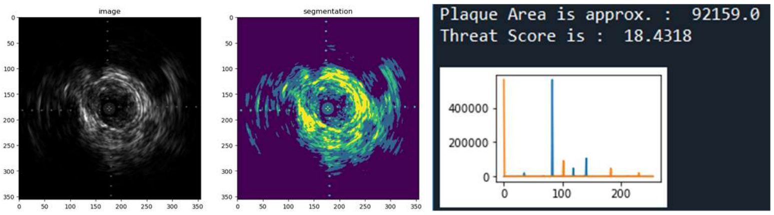
Test Id: 6

Description: Healthy Blood vessel IVUS image

Input:



Output:



Expected Threat Score: 0-30

Actual Threat Score: 18.4318

Result: PASS

Performance Testing

Test ID: 7

Description: Check execution Time

Input: Any random Image

Output execution Time: 6.4 sec

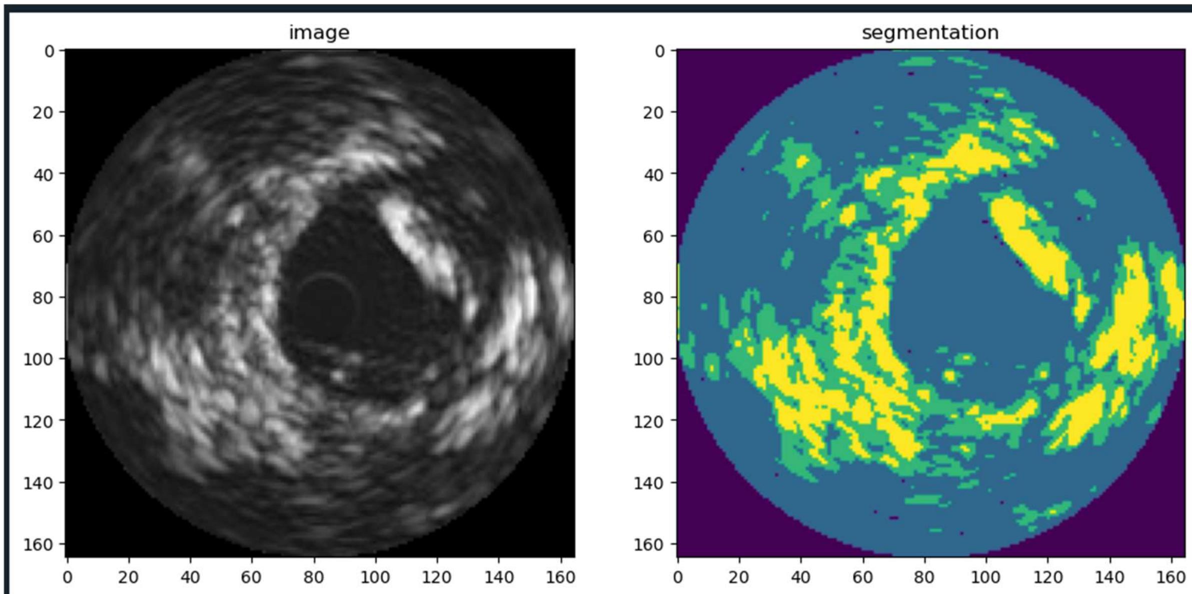
Threshold limit: 8 sec

Result: PASS

8. Screens and Reports

```
In [2]: runfile('C:/Users/Dell/Documents/A1_Project/main.py', wdir='C:/Users/Dell/Documents/A1_Project')
Getting filtered image..
Iteration 0 : cost = 382.310355
Iteration 1 : cost = 52.264397
Iteration 2 : cost = 61.709787
Iteration 3 : cost = 66.175555
Iteration 4 : cost = 58.212791
Iteration 5 : cost = 50.820313
Iteration 6 : cost = 36.568425
Iteration 7 : cost = 25.819680
Iteration 8 : cost = 19.760619
Iteration 9 : cost = 15.290945
Iteration 10 : cost = 10.485303
Iteration 11 : cost = 6.670183
Iteration 12 : cost = 4.854769
Iteration 13 : cost = 4.009060
Iteration 14 : cost = 3.383912
Iteration 15 : cost = 2.859688
Iteration 16 : cost = 2.408177
Iteration 17 : cost = 2.021582
Iteration 18 : cost = 1.692028
Iteration 19 : cost = 1.414463
Iteration 20 : cost = 1.181124
Iteration 21 : cost = 0.985585
Iteration 22 : cost = 0.822116
Iteration 23 : cost = 0.685887
Iteration 24 : cost = 0.572172
Iteration 25 : cost = 0.477269
Iteration 26 : cost = 0.398090
Iteration 27 : cost = 0.332042
Iteration 28 : cost = 0.276951
Iteration 29 : cost = 0.231001
Iteration 30 : cost = 0.192677
Iteration 31 : cost = 0.160713
Iteration 32 : cost = 0.134053
Iteration 33 : cost = 0.111816
Iteration 34 : cost = 0.093269
Iteration 35 : cost = 0.077799
Iteration 36 : cost = 0.064895
Iteration 37 : cost = 0.054131
Iteration 38 : cost = 0.045154
```

Fig 8.1 Iteration Costs of Sample Test Case 1




```
(48, 103, 141) --> 50.123366765089116 %
(68, 1, 84) --> 19.68811638554505 %
(53, 183, 120) --> 17.161966695755083 %
(253, 231, 36) --> 10.661183699329372 %
(0, 0, 0) --> 2.320418873202396 %
(147, 147, 147) --> 0.04494758107898103 %
{(48, 103, 141): 50.123366765089116, (68, 1, 84): 19.68811638554505, (53, 183, 120): 17.161966695755083, (253, 231, 36): 10.661183699329372, (0, 0, 0): 2.320418873202396, (147, 147, 147): 0.04494758107898103}
Plaque Area is approx. :  417845.0
Threat Score is :  83.569
```

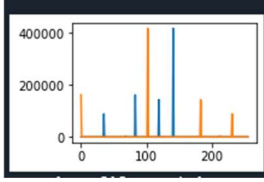


Fig 8.2 Segmented Image and Suggestions for Sample Test case 1

```
Getting filtered image..
Iteration 0 : cost = 383.708901
Iteration 1 : cost = 60.892732
Iteration 2 : cost = 68.489735
Iteration 3 : cost = 70.040306
Iteration 4 : cost = 61.207241
Iteration 5 : cost = 47.429892
Iteration 6 : cost = 33.403786
Iteration 7 : cost = 24.275033
Iteration 8 : cost = 19.250511
Iteration 9 : cost = 15.847721
Iteration 10 : cost = 13.343602
Iteration 11 : cost = 11.390488
Iteration 12 : cost = 9.803836
Iteration 13 : cost = 8.478668
Iteration 14 : cost = 7.351119
Iteration 15 : cost = 6.380716
Iteration 16 : cost = 5.539621
Iteration 17 : cost = 4.808059
Iteration 18 : cost = 4.170391
Iteration 19 : cost = 3.613746
Iteration 20 : cost = 3.129482
Iteration 21 : cost = 2.707405
Iteration 22 : cost = 2.340706
Iteration 23 : cost = 2.022190
Iteration 24 : cost = 1.745616
Iteration 25 : cost = 1.506244
Iteration 26 : cost = 1.299029
Iteration 27 : cost = 1.119718
Iteration 28 : cost = 0.964862
Iteration 29 : cost = 0.831125
Iteration 30 : cost = 0.715710
Iteration 31 : cost = 0.616221
Iteration 32 : cost = 0.530458
Iteration 33 : cost = 0.456562
Iteration 34 : cost = 0.392899
Iteration 35 : cost = 0.338070
Iteration 36 : cost = 0.290860
Iteration 37 : cost = 0.250220
Iteration 38 : cost = 0.215242
Iteration 39 : cost = 0.185141
Iteration 40 : cost = 0.159241
Iteration 41 : cost = 0.136958
Iteration 42 : cost = 0.117789
Iteration 43 : cost = 0.101299
Iteration 44 : cost = 0.087116
Iteration 45 : cost = 0.074916
Iteration 46 : cost = 0.064424
Iteration 47 : cost = 0.055400
Iteration 48 : cost = 0.047640
```

Fig 8.3 Iteration Costs of Sample Test Case 2

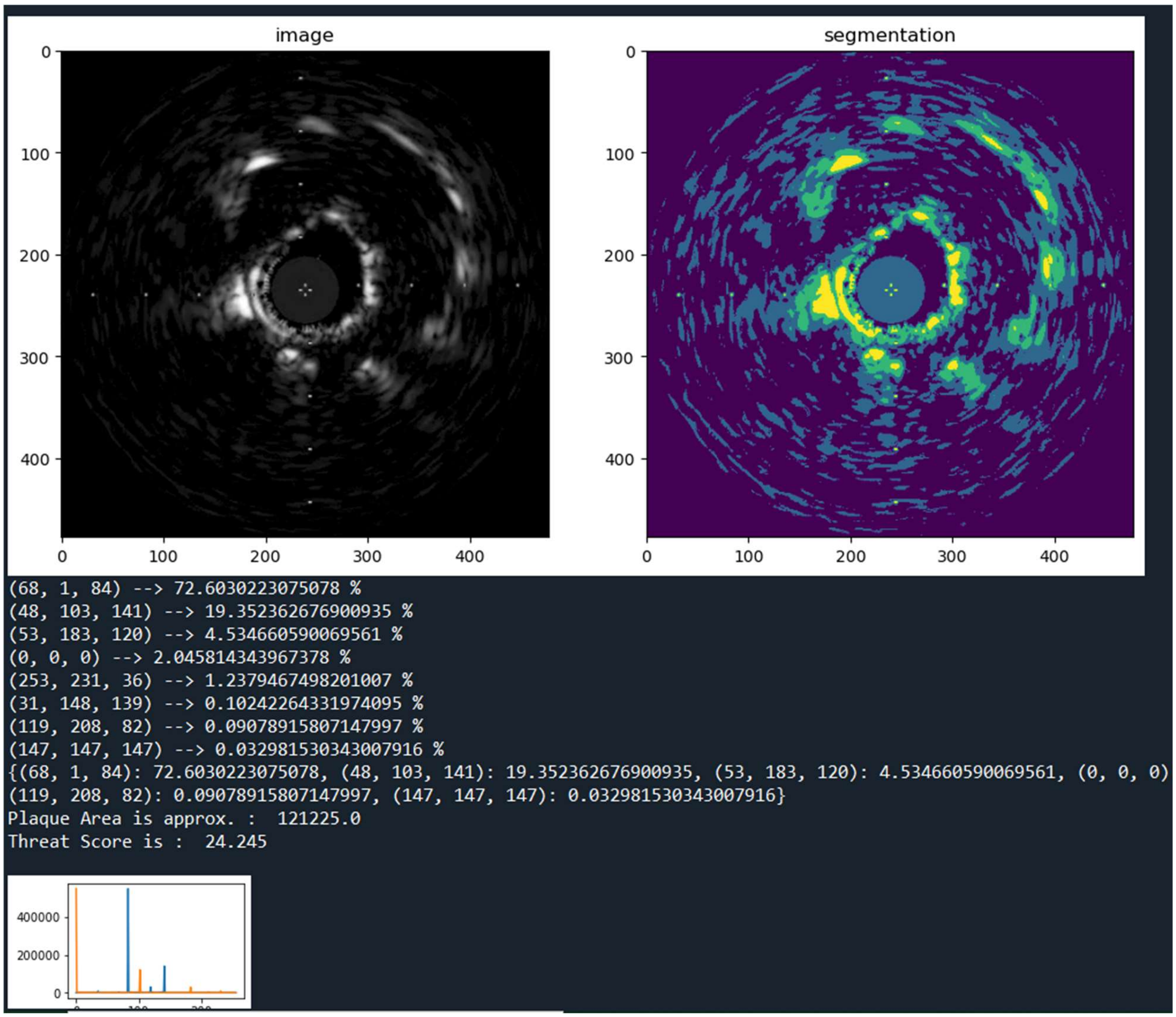


Fig 8.4 Segmented Image and Suggestions for Sample Test case 2

```
Getting filtered image..
Iteration 0 : cost = 381.943229
Iteration 1 : cost = 59.072060
Iteration 2 : cost = 41.160926
Iteration 3 : cost = 47.461667
Iteration 4 : cost = 67.495949
Iteration 5 : cost = 69.888033
Iteration 6 : cost = 57.150228
Iteration 7 : cost = 40.901379
Iteration 8 : cost = 27.455694
Iteration 9 : cost = 17.949147
Iteration 10 : cost = 11.639490
Iteration 11 : cost = 7.566428
Iteration 12 : cost = 4.980180
Iteration 13 : cost = 3.380410
Iteration 14 : cost = 2.451880
Iteration 15 : cost = 1.934276
Iteration 16 : cost = 1.593588
Iteration 17 : cost = 1.361201
Iteration 18 : cost = 1.179311
Iteration 19 : cost = 1.018638
Iteration 20 : cost = 0.876818
Iteration 21 : cost = 0.752697
Iteration 22 : cost = 0.644877
Iteration 23 : cost = 0.551696
Iteration 24 : cost = 0.471481
Iteration 25 : cost = 0.402617
Iteration 26 : cost = 0.343632
Iteration 27 : cost = 0.293173
Iteration 28 : cost = 0.250045
Iteration 29 : cost = 0.213212
Iteration 30 : cost = 0.181775
Iteration 31 : cost = 0.154953
Iteration 32 : cost = 0.132077
Iteration 33 : cost = 0.112571
Iteration 34 : cost = 0.095941
Iteration 35 : cost = 0.081764
Iteration 36 : cost = 0.069680
Iteration 37 : cost = 0.059381
Iteration 38 : cost = 0.050604
Iteration 39 : cost = 0.043123
```

Fig 8.5 Iteration Costs of Sample Test Case 3

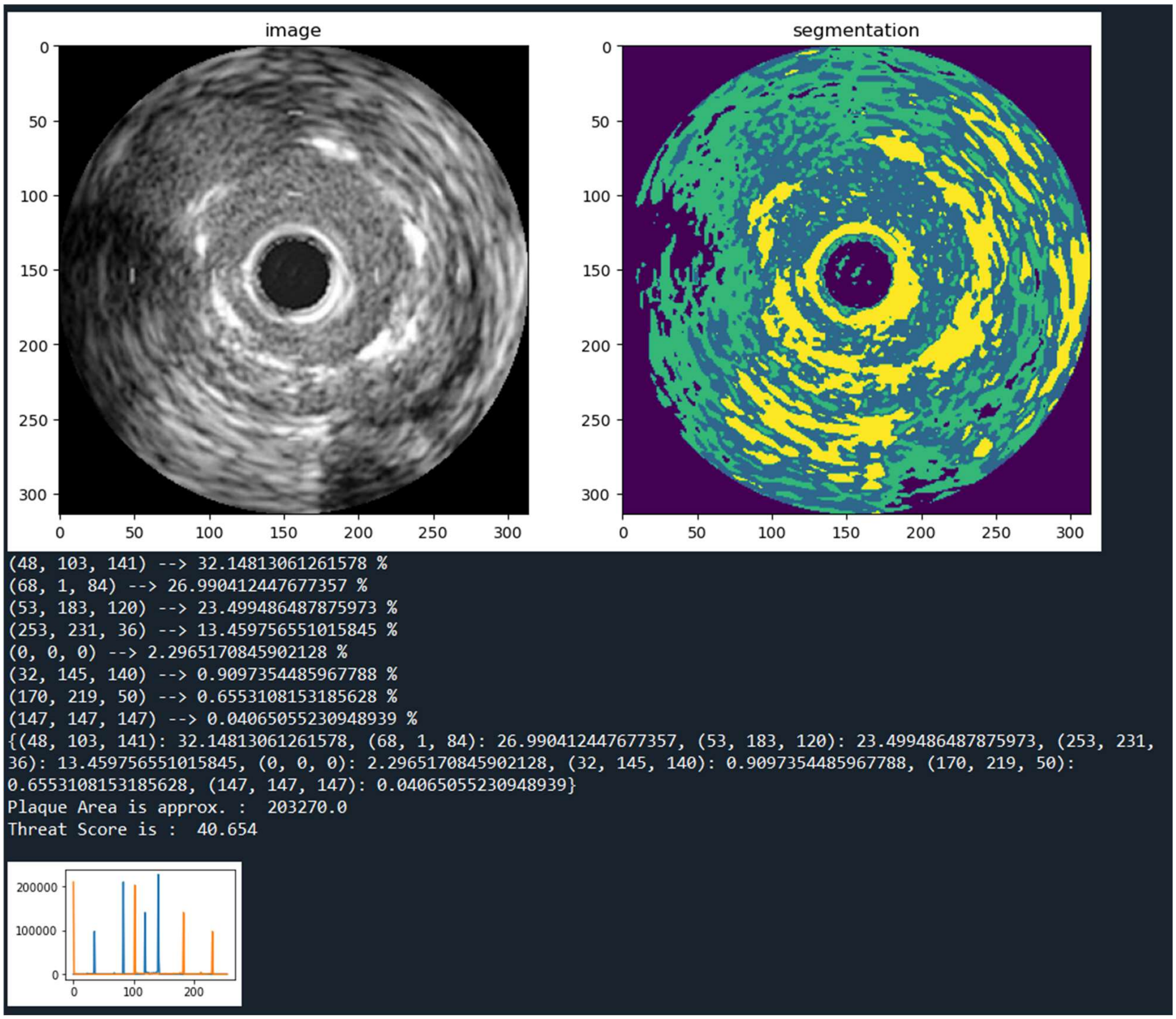


Fig 8.6 Segmented Image and Suggestions for Sample Test case 3

```
Getting filtered image..
Iteration 0 : cost = 383.961505
Iteration 1 : cost = 64.281062
Iteration 2 : cost = 60.649791
Iteration 3 : cost = 52.518489
Iteration 4 : cost = 47.478797
Iteration 5 : cost = 35.814458
Iteration 6 : cost = 23.888965
Iteration 7 : cost = 15.363826
Iteration 8 : cost = 10.369129
Iteration 9 : cost = 7.277333
Iteration 10 : cost = 5.300279
Iteration 11 : cost = 4.030985
Iteration 12 : cost = 3.144929
Iteration 13 : cost = 2.504995
Iteration 14 : cost = 2.029777
Iteration 15 : cost = 1.667308
Iteration 16 : cost = 1.383822
Iteration 17 : cost = 1.157768
Iteration 18 : cost = 0.974499
Iteration 19 : cost = 0.824228
Iteration 20 : cost = 0.699593
Iteration 21 : cost = 0.595315
Iteration 22 : cost = 0.507553
Iteration 23 : cost = 0.433358
Iteration 24 : cost = 0.370402
Iteration 25 : cost = 0.316844
Iteration 26 : cost = 0.271192
Iteration 27 : cost = 0.232222
Iteration 28 : cost = 0.198919
Iteration 29 : cost = 0.170435
Iteration 30 : cost = 0.146059
Iteration 31 : cost = 0.125190
Iteration 32 : cost = 0.107315
Iteration 33 : cost = 0.092000
Iteration 34 : cost = 0.078875
Iteration 35 : cost = 0.067627
Iteration 36 : cost = 0.057985
Iteration 37 : cost = 0.049719
```

Fig 8.7 Iteration Costs of Sample Test Case 4

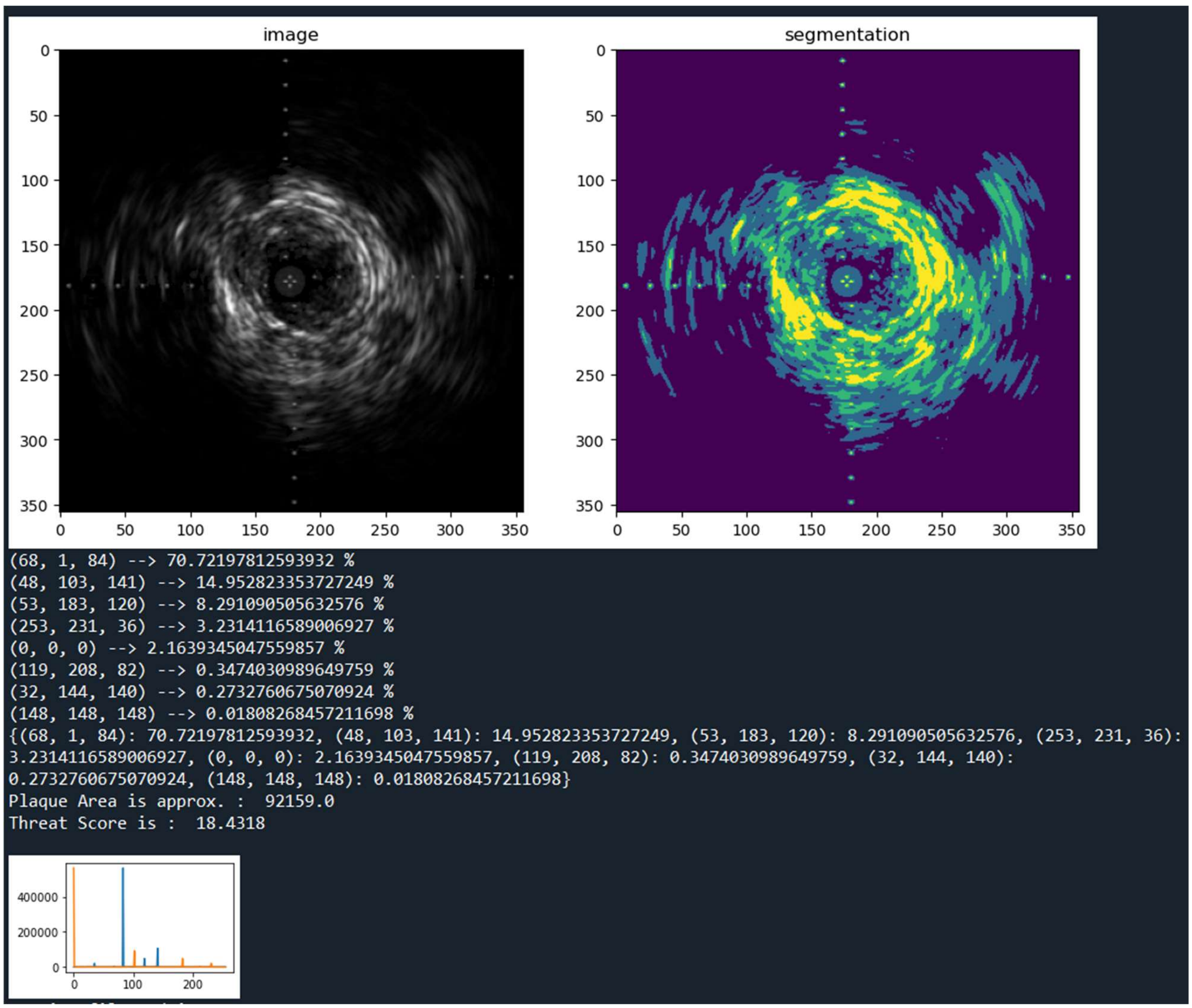
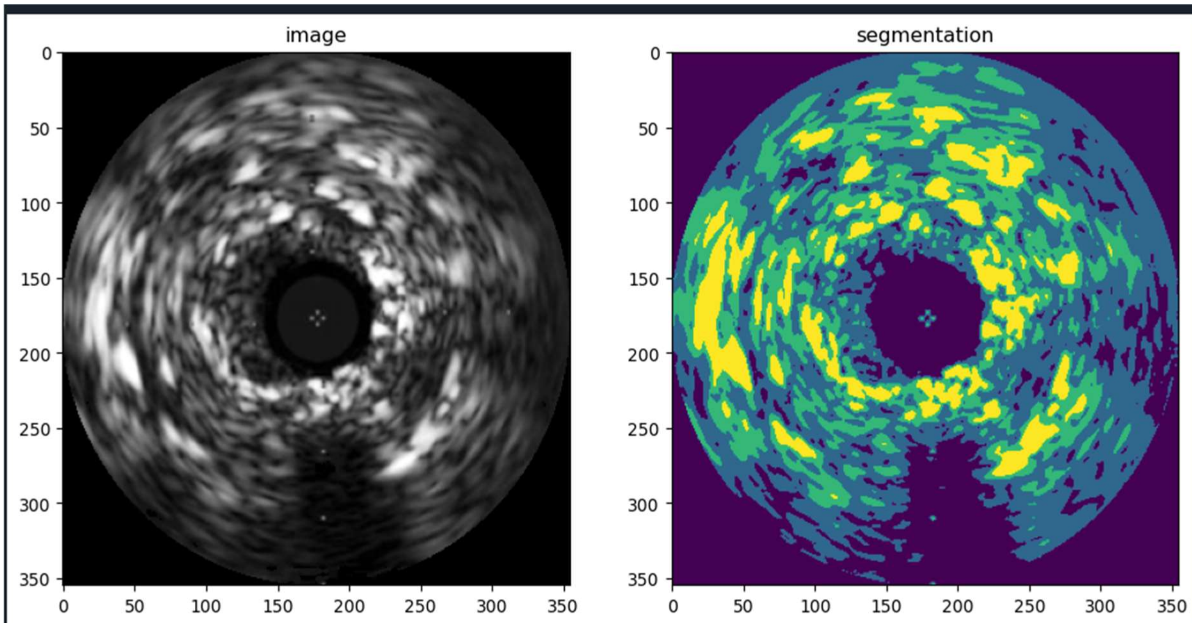


Fig 8.8 Segmented Image and Suggestions for Sample Test case 4

```
Getting filtered image..
Iteration 0 : cost = 382.265296
Iteration 1 : cost = 57.220360
Iteration 2 : cost = 55.934197
Iteration 3 : cost = 73.451231
Iteration 4 : cost = 85.500513
Iteration 5 : cost = 71.687340
Iteration 6 : cost = 51.490101
Iteration 7 : cost = 34.655121
Iteration 8 : cost = 22.524047
Iteration 9 : cost = 14.275988
Iteration 10 : cost = 8.898749
Iteration 11 : cost = 5.493567
Iteration 12 : cost = 3.379293
Iteration 13 : cost = 2.083292
Iteration 14 : cost = 1.299098
Iteration 15 : cost = 0.843160
Iteration 16 : cost = 0.599467
Iteration 17 : cost = 0.485640
Iteration 18 : cost = 0.423962
Iteration 19 : cost = 0.384394
Iteration 20 : cost = 0.350091
Iteration 21 : cost = 0.315688
Iteration 22 : cost = 0.282448
Iteration 23 : cost = 0.251386
Iteration 24 : cost = 0.222942
Iteration 25 : cost = 0.197234
Iteration 26 : cost = 0.174198
Iteration 27 : cost = 0.153674
Iteration 28 : cost = 0.135458
Iteration 29 : cost = 0.119335
Iteration 30 : cost = 0.105089
Iteration 31 : cost = 0.092519
Iteration 32 : cost = 0.081436
Iteration 33 : cost = 0.071671
Iteration 34 : cost = 0.063071
Iteration 35 : cost = 0.055499
Iteration 36 : cost = 0.048833
```

Fig 8.9 Iteration Costs of Sample Test Case 5




```
(68, 1, 84) --> 35.39571739757836 %
(48, 103, 141) --> 34.02694200248367 %
(53, 183, 120) --> 19.224648016353452 %
(253, 231, 36) --> 8.162236408946498 %
(0, 0, 0) --> 2.1639345047559857 %
(129, 211, 76) --> 0.5373311634111847 %
(31, 149, 139) --> 0.45074983264535307 %
(147, 147, 147) --> 0.03844067382549371 %
{(68, 1, 84): 35.39571739757836, (48, 103, 141): 34.02694200248367, (53, 183, 120): 19.224648016353452, (253, 231, 36):
8.162236408946498, (0, 0, 0): 2.1639345047559857, (129, 211, 76): 0.5373311634111847, (31, 149, 139):
0.45074983264535307, (147, 147, 147): 0.03844067382549371}
Plaque Area is approx. :  235822.0
Threat Score is :  47.1644
```

Fig 8.10 Segmented Image and Suggestions for Sample Test case 5

9. Conclusion and Future Scope

9.1 Conclusion

In this work we have explored the problem of bottom-up figure-ground segmentation, both as an image segmentation task, and as a perceptual grouping problem. We presented an comprehensive overview of current research in both fields, and discussed reasons why despite a vast research effort, image segmentation and perceptual grouping on unconstrained images continue to be extremely challenging

A major challenge for automatic image analysis is that the sheer complexity of the visual task which has been mostly ignored by the current approaches. New technological breakthrough in the areas of digital computation and telecommunication has relevance for future applications of image processing1. The satellite imaging and remote sensing applications programs of the future will feature a variety of sensors orbiting the earth. This technology is required for military and other types of surveillance, statistical data collection in the fields of forestry, agriculture, disaster prediction, weather prediction. In order to extract scientifically useful information, it will be necessary to develop techniques to register real-time data recorded by a variety of sensors for various applications.

From this Project any User without any non-medical knowledge can be aware of his cardiac threats by using his/her own IVUS images. This can be used in hospitals and ambulances. Technology not only upgrades our lives but also it can save them.

9.2 Future Scope

The future of image processing will involve scanning the heavens for other intelligent life out in space. Also new intelligent, digital species created entirely by research scientists in various nations of the world will include advances in image processing applications. Due to advances in image processing and related technologies there will be millions and millions of robots in the world in a few decades time, transforming the way the world is managed. Advances in image processing and artificial intelligence6 will involve spoken commands, anticipating the information requirements of governments, translating languages, recognizing and tracking people and things, diagnosing medical conditions, performing surgery, reprogramming defects in human DNA, and automatic driving all forms of transport. With increasing power and sophistication of modern computing, the concept of computation can go beyond the present limits and in future, image processing technology will advance and the visual system of man can be replicated. The future trend in remote sensing will be towards improved sensors that record the same scene in many spectral channels. Graphics data is becoming increasingly important in image processing app1ications. The future image processing applications of satellite based imaging ranges from planetary exploration to surveillance applications. The cellular neural network is an implementable alternative to fully connected neural networks and has evolved into a paradigm for future imaging techniques. The usefulness of this technique has applications in the areas of silicon retina, pattern formation, etc.

In future It can be extended to alarms and information systems projects so that they are efficiently utilized. There may be no Cardiac arrests in the future and it almost decreases death rate by cardiac arrest.

10. BIBILOGRAPHY

10.1 Web Links

[6] https://medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac

[7] https://www.researchgate.net/publication/32552718

10.2 References

[1] Arun, C., Akshatha Prabhu, Mohammed Zeeshan, and N. Shobha Rani. ""A Study on Various Classifier Techniques Used in Image Processing."" In 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 1132-1140. IEEE, 2020."

[2] Chen J. Image recognition technology based on neural network. IEEE Access.

[3] Bosch A, Zisserman A, Munoz X. Image classification using random forests and ferns. In2007 IEEE 11th international conference on computer vision 2007 Oct 14 (pp. 1-8). Ieee."

[4] Sun X, Liu L, Wang H, Song W, Lu J. Image classification via support vector machine. In2015 4th International Conference on Computer Science and Network Technology (ICCSNT) 2015 Dec 19 (Vol. 1, pp. 485-489). IEEE."

[5] Li Y, Cheng B. An improved k-nearest neighbor algorithm and its application to high resolution remote sensing image classification. In2009 17th International Conference on Geoinformatics 2009 Aug 12 (pp. 1-4). Ieee.

[6] Christ MJ, Parvathi RM. Fuzzy c-means algorithm for medical image segmentation. In2011 3rd International Conference on Electronics Computer Technology 2011 Apr 8 (Vol. 4, pp. 33-36). IEEE."

[9] H. Van Lung and J. Kim, "A generalized spatial fuzzy c-means algorithm for medical image segmentation," 2009 IEEE International Conference on Fuzzy Systems, 2009, pp. 409-414, doi: 1109/FUZZY.2009.5276878.

[10] J. R. Smith and Shih-Fu Chang, "Single color extraction and image query," Proceedings., International Conference on Image Processing, 1995, pp. 528-531 vol.3, doi: 10.1109/ICIP.1995.537688