**An Industry-Oriented Mini Project Report**

**On**

# “Integrating domain knowledge into Deep Networks for lung ultrasound with applications to Lung cancer”

**Submitted in Partial Fulfillment of the Academic Requirement for the Award of Degree of**

# BACHELOR OF TECHNOLOGY

**in**

### Computer Science & Engineering

**Submitted by**

|  |  |
| --- | --- |
| **ALDOORI CHARAN** | **20R01A0501** |
| **ALLADI R CHINMAYI** | **20R01A0502** |
| **BAYYAPU NAGA VARSHA** | **20R05A0512** |

### Under the esteemed guidance of Mrs. P. Deepika

(Assistant Professor, CSE Department)



# CMR INSTITUTE OF TECHNOLOGY

**(UGC AUTONOMUS)**

**Approved by AICTE, Permanent Affiliation to JNTUH, Accredited by NBA and NAAC**

**Kandlakoya (V), Medchal Dist – 501 401** [**www.cmritonline.ac.in**](http://www.cmritonline.ac.in/)

# 2023-2024

**CMR INSTITUTE OF TECHNOLOGY**

**(UGC AUTONOMUS)**

**Approved by AICTE, Permanent Affiliation to JNTUH, Accredited by NBA and NAAC**

**Kandlakoya (V), Medchal Dist – 501 401** [**www.cmritonline.ac.in**](http://www.cmritonline.ac.in/)



**CERTIFICATE**

This is to certify that an industry oriented Mini Project entitled with **“Integrating domain knowledge into Deep Networks for lung ultrasound with applications to Lung cancer”** is being submitted by

|  |  |
| --- | --- |
| **ALDOORI CHARAN** | **20R01A0501** |
| **ALLADI R CHINMAYI** | **20R01A0502** |
| **BAYYAPU NAGA VARSHA** | **20R05A0512** |

To JNTUH, Hyderabad in partial fulfillment of the requirement for award of the degree of B- Tech in CSE and is a record of a bonafide work carried out under our guidance and supervision. The results in this project have been verified and are found to be satisfactory. The results embodied in this work have not been submitted to have any other University for award of any other degree or diploma.

|  |  |
| --- | --- |
| **Signature of Guide** | **Signature of HOD** |
| **Mrs. P. DEEPIKA** | **Mr. A. PRAKASH** |

**EXTERNAL EXAMINER**

# ACKNOWLEDGEMENT

We are extremely grateful to **Dr. M. Janga Reddy**, **Director**, **Dr. B. Satyanarayana**, **Principal** and **Mr. A. Prakash**, **Head of Department**, Dept of Computer Science and Engineering, CMR Institute of Technology for their inspiration and valuable guidance during entire duration.

We are extremely thankful to our Mini Project internal guide **Mrs. P. Deepika,** Dept of Computer Science and Engineering, CMR Institute of Technology for her constant guidance, encouragement and moral support throughout the project.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this Project.

We express our thanks to all staff members and friends for all the help and coordination extended in bringing out this Project successfully in time.

Finally, we are very much thankful to our parents and relatives who guided directly or indirectly for every step towards success.

|  |  |
| --- | --- |
| **ALDOORI CHARAN** | **20R01A0501** |
| **ALLADI R CHINMAYI** | **20R01A0502** |
| **BAYYAPU NAGA VARSHA** | **20R01A0512** |

# ABSTRACT

In this computer era we are totally going with the automation of everything, in the same way the medical industry is also automated with the help of image processing and data analytics .The best way to control the death cause by cancer is early detection. The medical image or a CT scan image is pre-processed .The contrast of the image is increased with the CLAHE Equalization technique .Then it is segmented with the help of random walk segmentation method. In segmentation the three process will happen the ROI of image is segmented and then then the border correction is done. As third part the continuous pixel change is segmented. The classification is the major portion where the cancerous and non-cancerous is identified with the pre trained model. All the methods used above deals with the traditional way of image processing and data analytics. In Future this accuracy will be boosted with the modern XGboost algorithm where less data is used to get high accuracy

# INDEX

|  |  |
| --- | --- |
| **CONTENT**  **ACKNOWLEDGEMENT**  **ABSTRACT** | **PAGE NO**  **i**  **ii** |
| **1. INTRODUCTION** | **1-3** |
| 1.1 ABOUT PROJECT | **1-3** |
| **2. SYSTEM ANALYSIS** | **4-7** |
| 2.1 EXISTING SYSTEM | **4** |
| 2.2 DISADVANTAGES OF EXISTING SYSTEM | **5** |
| 2.3 PROPOSED SYSTEM | **6** |
| 2.4 ADVANTAGES OF PROPOSED SYSTEM | **7** |
| **3. SYSTEM STUDY** | **8-9** |
| 3.1 FEASIBILITY STUDY | **8-9** |
| **4. HARDWARE AND SOFTWARE REQUIREMENTS** | **10** |
| 4.1 HARDWARE REQUIREMENTS | **10** |
| 4.2 SOFTWARE REQUIREMENTS  **5. SOFTWARE ENVIRONMENT**  5.1 WHAT IS PYTHON  5.2 ADVANTAGES OF PYTHON  5.3 ADVANTAGES OF PYTHON OVER OTHER  LANGUAGES  **6. MACHINE LEARNING**  6.1 WHAT IS MACHINE LEARNING?  6.2 TYPES OF MACHINE LEARNING  6.3 ADVANTAGES OF MACHINE LEARNING | **10**  **11-14**  **11**  **12-13**  **14**  **15-16**  **15**  **15-16**  **16** |
| **7. ARCHITECTURE** | **17** |
| **6. MODULES** | **18** |
| **7. DIAGRAMS** | **19-22** |
| 7.1 DATA FLOW DIAGRAM | **19** |
| 7.2 USE CASE DIAGRAM | **20** |
| 7.3 CLASS DIAGRAM | **21** |
| 7.4 SEQUENCE DIAGRAM | **21** |
| 7.5 ACTIVITY DIAGRAM | **22** |
| **8. IMPLEMENTATION** | **23-30** |
| 8.1 ALGORITHMS | **23-26** |
| 8.2 SOURCE CODE | **26-30** |
| **9. SCREEN SHOTS** | **31-34** |
| **10. TESTING** | **35-37** |
| **11. CONCLUSION** | **38** |
| **12. REFERENCES** | **39-40** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| FIGURE NO | PARTICULARS | PAGE NO |
| 7 | Architecture Diagram | 17 |
| 9.1 | Data Flow Diagram | 19 |
| 9.2 | Use Case Diagram | 20 |
| 9.3 | Class Diagram | 21 |
| 9.4 | Sequence Diagram | 21 |
| 9.5 | Activity Diagram | 22 |

Iv

**LIST OF SCREENSHOTS**

|  |  |  |
| --- | --- | --- |
| FIGURE NO | PARTICULARS | PAGE NO |
| 11.1 | Test Samples | 31 |
| 11.2 | Upload Lung Cancer Data Set | 31 |
| 11.3 | Training and Testing Data Set | 32 |
| 11.4 | Executing SVM Algorithm | 32 |
| 11.5 | Executing KNN Algorithm | 33-34 |
| 11.6 | Predicting the disease | 33 |
| 11.6.1 | Example 1 | 33 |
| 11.6.2 | Example 2 | 34 |
| 11.7 | Error Rate Graph | 34 |

v

**1.INTRODUCTION**

## ABOUT PROJECT

Lung cancer growth has turned out to be a standout amongst the most widely recognized reasons for disease in the two people. Countless bite the dust each year because of lung malignancy. The illness has diverse stages whereby it begins from the little tissue and spreads all through the distinctive territories of the lungs by a procedure called metastasis. It is the uncontrolled development of undesirable cells in the lungs. It is assessed that around 12,203 people had lung disease in 2016, 7130 guys and 5073 females; passing from lung malignant growth in 2016 were 8839. . Biomedical image handling is the most recent rising apparatus in medicinal research utilized for the early recognition of malignancies. Biomedical image handling strategies can be utilized in the restorative field to analysis maladies at the beginning time. It utilizes biomedical images, for example, X-beams, Computed innovation and MRIs. The principle commitment of image handling in the restorative field is to analysis the malignant growth at the beginning time, expanding survival rates. The time factor is basic for tumors of the mind, the lungs, and bosoms. image handling can identify these malignant growths in the early periods of the maladies encouraging an early treatment process. The image preparing procedure comprises of four essential stages, pre-handling, division, including extraction and grouping. This paper presents image preparing procedures whereby the CT examine image is utilized as information image, is handled and beginning period lung disease is distinguished utilizing an SVM (bolster vector machine) calculation as a classifier in the grouping stage to improve exactness, affectability, and explicitness. First the image is pre-handled and divided. After that Features are removed from the sectioned image lastly the image is delegated ordinary or destructive. Advanced image handling is the utilization of PC calculations to perform image preparing on computerized images.

As a subfield of advanced flag preparing, computerized image handling has numerous points of interest over simple image preparing.[19][2] It permits a lot more extensive scope of calculations to be connected to the information data — the point of advanced image handling is to improve the image information (Features) by stifling undesirable mutilations as well as upgrade of some vital image includes with the goal that our AI Computer Vision models can profit by this improved information to take a shot at. Feature extraction begins from an underlying arrangement of estimated information and assembles determined qualities (Features) proposed to be useful and non-excess, encouraging the resulting learning and speculation steps, and at times prompting better human elucidations.

[12] Feature extraction is a dimensionality decrease process, where an underlying arrangement of crude factors is diminished to progressively sensible gatherings (Features) for handling, while still precisely and totally portraying the first informational collection.

At the point when the information to a calculation is too substantial to be in any way handled and it is suspected to be repetitive (for example a similar estimation in the two feet and meters, or the redundancy of images introduced as pixels), at that point it very well may be changed into a decreased arrangement of Features (additionally named a component vector). Deciding a subset of the underlying Features is called include choice. The chose Features are relied upon to contain the pertinent data from the information, with the goal that the ideal undertaking can be performed by utilizing this decreased portrayal rather than the total introductory information. Feature extraction includes lessening the measure of assets required to depict a substantial arrangement of information. When performing examination of complex information one of the serious issues originates from the quantity of factors included. Examination with countless for the most part requires a lot of memory and calculation control, likewise it might make an arrangement calculation overfit to preparing tests and sum up ineffectively to new examples.[08] Feature extraction is a general term for strategies for building mixes of the factors to get around these issues while as yet portraying the information with adequate exactness.

Many AI specialists trust that appropriately streamlined component extraction is the way to successful model development. The advent of deep learning has revolutionized bridge the gap between advanced deep neural networks and the intricate understanding of lung pathology by integrating domain-specific insights into the model architecture. This research aims to enhance the diagnostic capabilities of deep neural networks by incorporating expert knowledge from the domain of lung ultrasound, specifically focusing on applications related to the detection and diagnosis of lung cancer medical image analysis, offering unprecedented capabilities for automated detection and diagnosis. However, in the domain of lung ultrasound, where subtle patterns and nuanced features are critical for accurate diagnosis, the incorporation of domain knowledge becomes imperative. Lung cancer is a significant global health concern, and early and accurate detection is critical for effective treatment and improved patient outcomes. This research leverages deep learning techniques, which excel at pattern recognition in complex data, and integrates them with domain-specific insights provided by medical professionals in the field of lung ultrasound.

The integration of domain knowledge involves encoding a comprehensive understanding of lung anatomy, cancer pathology, and the nuances of ultrasound imaging characteristics into the architecture and training process of deep neural networks. By combining the power of machine learning algorithms with the nuanced expertise of healthcare practitioners, this approach aims to create a more robust and clinically relevant model for identifying potential signs of lung cancer in ultrasound images.

## 

## 

## 2.SYSTEM ANALYSIS

**2.1. EXISTING SYSTEM**

In existing paper, a picture handling procedures has been utilized to recognize beginning time lung malignant growth in CT examine pictures. The CT filter picture is pre-prepared pursued by division of the ROI of the lung. Discrete waveform Transform is connected for picture pressure and highlights are extricated utilizing a GLCM. The outcomes are encouraged into a SVM classifier to decide whether the lung picture is carcinogenic or not. The SVM classifier is assessed dependent on a LIDC dataset. The existing paper describes a system for the early detection of lung cancer in CT (computed tomography) scan images using a combination of image processing techniques, feature extraction methods, and machine learning.

The CT scan images undergo pre-processing to enhance image quality and facilitate subsequent analysis. This step may involve noise reduction, contrast enhancement, or other techniques to prepare the images for further processing. The system identifies and isolates the Region of Interest (ROI) in the lung from the CT scan images. This is a critical step to focus the analysis on the relevant area for lung cancer detection. DWT, a technique commonly used for signal processing and image compression, is applied to compress the lung CT scan images. This step helps in reducing the dimensionality of the data while preserving important features. GLCM is employed to extract texture features from the compressed lung images. GLCM is a statistical method that captures the spatial relationships of pixel intensities, providing information about the texture and patterns within the images. The extracted features are then used as input to a Support Vector Machine (SVM) classifier. SVM is a supervised machine learning algorithm capable of categorizing data into different classes. In this case, it is trained to classify lung images as either cancerous or non-cancerous based on the extracted features. The system is evaluated using a dataset, specifically mentioned as the Lung Image Database Consortium (LIDC) dataset. LIDC is a well-known dataset in the field of lung cancer imaging, containing labeled images for training and testing machine learning models. The performance of the SVM classifier is assessed based on metrics such as accuracy, sensitivity, specificity, or other relevant measures. This evaluation helps determine how well the system can distinguish between cancerous and non-cancerous lung images.

**2.2. DISADVANTAGES OF EXISTING SYSTEM**

## The existing system has some drawbacks:

## Complexity: The process involves multiple steps, making it somewhat complex.

## Information Loss: Image compression may result in some loss of information.

## Accuracy: The existing approach often falls short in providing precise ,accurate and reliable predictions, potentially leading to delayed diagnoses and compromised patient outcomes.

**2.3 PROPOSED SYSTEM**

The proposed model applies a range of algorithms to the different stages of image processing. In this proposed model, first the CT scan image is pre-processed and the ROI (region of interest) is separated in preparation for segmentation.[17] At the segmentation stage, Discrete Wavelet Transform (DWT) is applied and the feature is extracted by using a GLCM (Gray level co-occurrence matrix) such as correlation, entropy, variance, contrast, dissimilarity and energy. After the feature extraction stage, classification is carried out by an SVM (support vector machine) for classification of cancerous and non-cancerous nodules.

The proposed system outlined in the description focuses on enhancing the detection of cancerous and non-cancerous nodules in CT scan images through a multi-stage process involving various algorithms. The CT scan image undergoes pre-processing to prepare it for subsequent analysis. This step is crucial for enhancing image quality, reducing noise, and ensuring that the image is optimized for further processing. The pre-processed image is then segmented to isolate the Region of Interest (ROI). This is the area of the image that is deemed relevant for the subsequent analysis, specifically focusing on the region where nodules may be present.

DWT is applied during the segmentation stage. DWT is known for its ability to provide both localization in both time and frequency domains. In the context of image processing, DWT can be used for multi-resolution analysis, aiding in the extraction of important features.

Features are extracted from the segmented and wavelet-transformed image using a Gray Level Co-occurrence Matrix (GLCM). GLCM is a statistical method that quantifies the spatial relationships of pixel intensities.

The extracted features serve as input for a Support Vector Machine (SVM) classifier. SVM is a supervised learning algorithm capable of categorizing data into different classes. In this case, it classifies the nodules as either cancerous or non-cancerous based on the features extracted from the GLCM.

The SVM classifier makes the final determination of whether a nodule is cancerous or non-cancerous. This classification is based on the learned patterns from the features extracted during the earlier stages of the process.

## 2.4.ADVANTAGES OF PROPOSED SYSTEM

## Early Detection: By integrating advanced image processing techniques and machine learning, the model is designed to detect lung nodules at an early stage.

## Flexible and Adaptable: The proposed model can potentially be adapted for different datasets and CT scan imaging conditions.

## Multi-Stage Processing: The model employs a multi-stage processing approach, including pre-processing, segmentation, feature extraction, and classification.

## Non-invasive Nature: The use of CT scan images for nodule detection is a non-invasive approach this is beneficial for patient comfort and safety.

## 

**3.SYSTEM STUDY**

**3.1 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

♦ ECONOMICAL FEASIBILITY

♦ TECHNICAL FEASIBILITY

♦ SOCIAL FEASIBILITY

**3.1.1 ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### 3.1.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**3.1.3 SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**4. HARDWARE AND SOFTWARE REQUIREMENTS:**

**4.1. HARDWARE REQUIREMENTS:**

* **PROCESSOR :** Intel(R) Core(TM) i7-8650U CPU @ 1.90GHz 2.11 GHz
* **RAM :** 8 GB
* **SYSTEM TYPE :** 64-bit operating system, x64-based processor

**4.2 SOFTWARE REQUIREMENTS:**

* **Operating system** : Windows.
* **Coding Language** : Python 3.7

Python 3.7 is a programming language known for its simplicity and readability. Released in 2018, it introduced new features like data classes for simpler data representation and improved performance through various optimizations. Developers appreciate its syntax, making code writing straightforward. Additionally, Python 3.7 includes asyncio improvements for asynchronous programming, ensuring efficient handling of concurrent tasks. Overall, Python 3.7 enhances the user experience by providing a more robust and efficient platform for developers to create diverse applications with ease.

**5. SOFTWARE ENVIRONMENT**

# 5.1 What is Python?

Below are some facts about Python. Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc. The biggest strength of Python is huge collection of standard library which can be used for the following –

* + [Machine Learning](https://www.geeksforgeeks.org/machine-learning/)
  + GUI Applications (like Kivy, Tkinter, PyQt etc. )
  + Web frameworks like Django (used by YouTube, Instagram, Dropbox)
  + Image processing (like Opencv, Pillow)
  + Web scraping (like Scrapy, BeautifulSoup, Selenium)
  + Test frameworks
  + Multimedia

### 5.2 Advantages of Python: -

Let’s see how Python dominates over other languages.

#### 1. Extensive Libraries

#### Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don’t have to write the complete code for that manually.

#### 2. Extensible

As we have seen earlier, Python can be**extended to other languages**. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

#### 3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add **scripting capabilities**to our code in the other language.

#### 4. Improved Productivity

The language’s simplicity and extensive libraries render programmers**more productive**than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

#### 5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

#### 6. Simple and Easy

When working with Java, you may have to create a class to print **‘Hello World’**. But in Python, just a print statement will do. It is also quite **easy to learn, understand,** and**code.**

#### 7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and **indentation is mandatory.** This further aids the readability of the code.

#### 8. Object-Oriented

This language supports both the procedural and object-orientedprogramming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the **encapsulation of data** and functions into one.

#### 9. Free and Open-Source

Like we said earlier, Python is **freely available.** But not only can you[**download Python**](https://data-flair.training/blogs/install-python-windows/) for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

#### 10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn’t the same with Python. Here, you need to**code only once**, and you can run it anywhere. This is called **Write Once Run Anywhere (WORA).** However, you need to be careful enough not to include any system-dependent features.

### 5.3 Advantages of Python Over Other Languages

#### 1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don’t have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

#### 2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

#### 3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and[**machine learning**](https://data-flair.training/blogs/machine-learning-tutorials-home/), automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

**6. MACHINE LEARNING**

**6.1 What is Machine Learning?**

#### Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

#### Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

#### 6.2 Types of Machine Learning

* **Supervised Learning –**This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
* **Unsupervised Learning –**This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
* **Semi-supervised Learning –**This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.
* **Reinforcement Learning –**This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

### 6.3 Advantages of Machine learning

#### 1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

#### 2. No human intervention needed (automation)

With ML, you don’t need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

#### 3. Continuous Improvement

As [**ML algorithms**](https://data-flair.training/blogs/machine-learning-algorithms/) gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

#### 4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

#### 5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

**7. ARCHITECTURE**

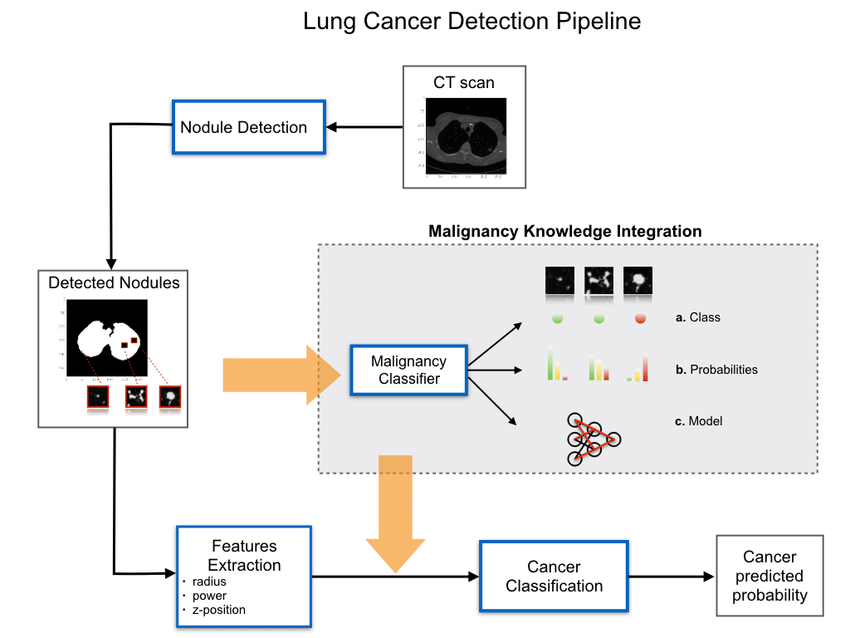


Fig 7: Architecture of Lung cancer detection

**8.MODULES**

* **TENSORFLOW:** TensorFlow is an open-source machine learning library developed by the Google Brain team. It is widely used for various machine learning and deep learning applications, offering a flexible platform for building and deploying machine learning models.
* **NUMPY:** Numpy which stands for Numerical Python, is a fundamental package for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.
* **SCIKIT-LEARN:** Scikit-learn is a widely used open-source machine learning library for Python. It provides simple and efficient tools for data analysis and modeling.
* **PANDAS:** Pandas is an open-source data manipulation and analysis library for Python. It provides data structures for efficiently handling large datasets and tools for data cleaning, exploration, and analysis.
* **MATPLOTLIB:** Matplotlib is a comprehensive data visualization library.It is widely used for generating a variety of charts and graphs to visualize data in a clear and concise manner.

**9.DIAGRAMS**

**User**

**Unauthorized user**

**Check**

Upload Lung Cancer Dataset

Read &split Dataset To Train & Test

Execute SVM Algorithms

Execute K-Means Algorithm

Predict Lung Cancer

Accuracy Graph

**End process**

FIG 9.1 : DATA FLOW DIAGRAM

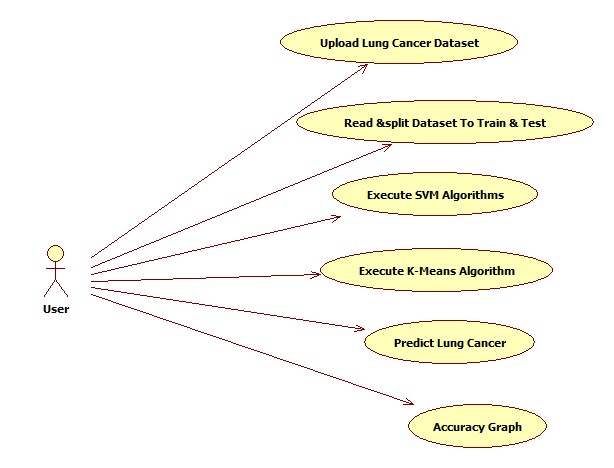
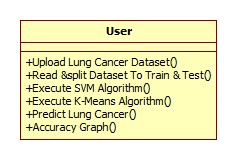


FIG : 9.2 USE CASE DIAGRAM

****

.

FIG : 9.3 CLASS DIAGRAM

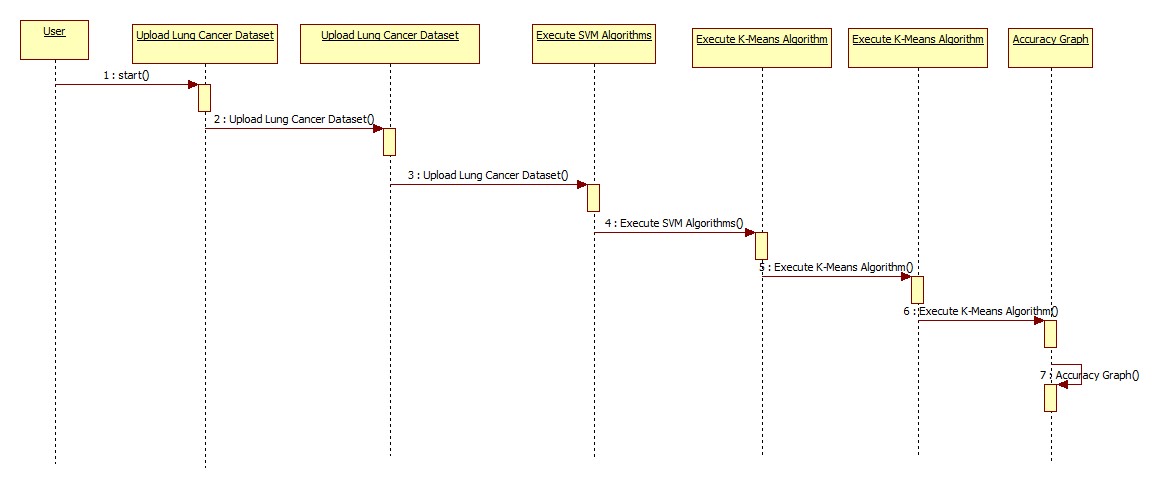
****

FIG : 9.4 SEQUENCE DIAGRAM

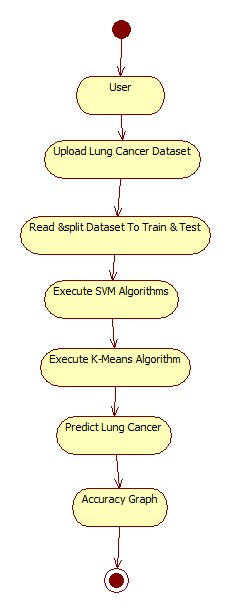
****

FIG : 9.5. ACTIVITY DIAGRAM

**10. IMPLEMENTATION**

**10.1 ALGORITHMS**

**K-Nearest Neighbors (KNN) :**

K-Nearest Neighbors (KNN) is a supervised machine learning algorithm used for classification and regression tasks. It is a simple and intuitive algorithm that makes predictions based on the majority class or average value of the k-nearest data points in the feature space.

**Basic Idea:**

The fundamental idea behind KNN is to predict the class or value of a data point based on the classes or values of its k-nearest neighbors. In other words, it assumes that similar data points are likely to belong to the same class or have similar values**.**

**Distance Metric:**

KNN relies on a distance metric (e.g., Euclidean distance, Manhattan distance) to measure the similarity between data points in the feature space. The choice of distance metric depends on the nature of the data and the problem at hand.

**Training Phase:**

KNN is often referred to as a "lazy learner" because it doesn't explicitly learn a model during the training phase. Instead, it memorizes the training dataset, which consists of labeled examples with known classes or values.

**Prediction:**

For a given data point in the testing phase, KNN identifies its k-nearest neighbors based on the chosen distance metric. The class or value of the new data point is then determined by a majority vote (for classification) or an average (for regression) of the k-nearest neighbors.

**Hyperparameter k:**

The choice of the parameter k (number of neighbors) is a crucial aspect of KNN. A smaller value of k

can make the algorithm more sensitive to noise, while a larger value of k can make it more robust but potentially less sensitive to local variations in the data.

**Decision Boundaries:**

KNN doesn't explicitly create a decision boundary during training. The decision boundary emerges naturally based on the distribution of data points in the feature space. The algorithm can adapt well to complex decision boundaries and is non-parametric.

**Applicability:**

KNN is versatile and can be applied to both classification and regression tasks. It is effective for datasets with a moderate number of features and instances. However, its performance may degrade with high-dimensional data due to the curse of dimensionality.

**Scalability:**

The computational cost of making predictions with KNN grows with the size of the training dataset, as it requires calculating distances for each test instance. This can make the algorithm computationally expensive for large datasets.

**Limitations:**

KNN can be sensitive to outliers, and its performance may be affected by irrelevant or redundant features. Additionally, the choice of k and the distance metric can significantly impact the algorithm's effectiveness.

In summary, K-Nearest Neighbors is a straightforward and interpretable algorithm suitable for various classification and regression tasks. Its simplicity and effectiveness make it a popular choice, especially for small to moderately sized datasets with clear patterns in the feature space.

**Support Vector Machine (SVM) :**

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks. SVM is particularly effective in high-dimensional spaces and is well-suited for tasks where clear margins of separation exist between different classes.

**Hyperplane:**

In a binary classification scenario, a hyperplane is a decision boundary that separates the data into two classes. SVM aims to find the hyperplane that maximizes the margin, which is the perpendicular distance from the hyperplane to the nearest data points (support vectors).

**Support Vectors:**

Support vectors are the data points that lie closest to the decision boundary or hyperplane. These points are critical in determining the optimal hyperplane and contribute to defining the margin.

**Margin:**

The margin is the distance between the decision boundary and the nearest data points from each class. SVM aims to maximize this margin, providing a robust separation between classes. A larger margin often leads to better generalization to unseen data.

**Kernel Trick:**

SVM can use a kernel trick to transform the input features into a higher-dimensional space, making it possible to find a hyperplane that separates the data when a linear separation is not possible in the original feature space. Common kernel functions include linear, polynomial, radial basis function (RBF), and sigmoid.

**C Parameter:**

SVM introduces a regularization parameter, C, that controls the trade-off between achieving a smooth decision boundary and classifying training points correctly. A smaller C allows for a softer margin with misclassifications, while a larger C enforces a stricter margin with fewer misclassifications.

**Multi-Class Classification:**

SVM inherently supports binary classification, but it can be extended to handle multi-class classification through techniques like one-vs-one or one-vs-all. In one-vs-one, a separate binary classifier is trained for each pair of classes, while in one-vs-all, a binary classifier is trained for each class against the rest.

**Regression (Support Vector Regression - SVR):**

SVM can also be used for regression tasks. In Support Vector Regression (SVR), the objective is to fit as many data points as possible within a specified margin, rather than classifying them into discrete categories.

**Effective in High-Dimensional Spaces:**

SVM performs well in high-dimensional spaces, making it suitable for tasks with many features, such as text classification, image recognition, and bioinformatics.

**Scalability:**

SVMs can be computationally expensive, especially for large datasets. The time complexity is generally between O(n^2) and O(n^3), where n is the number of data points. For large datasets, optimized implementations and techniques such as stochastic gradient descent can be employed.

**Interpretability:**

SVMs can be less interpretable compared to some other algorithms, as the decision boundary is often complex, especially in high-dimensional spaces.

**Use Cases:**

SVM is used in a variety of applications, including text classification, image classification, handwriting recognition, and bioinformatics.

**10.2 SOURCE CODE :**

from tkinter import messagebox

from tkinter import \*

from tkinter import simpledialog

import tkinter

from tkinter import filedialog

import matplotlib.pyplot as plt

import numpy as np

from tkinter.filedialog import askopenfilename

import pandas as pd

import os

import cv2

import numpy as np

from sklearn import svm

from sklearn.metrics import accuracy\_score

from sklearn.model\_selection import train\_test\_split

from sklearn.decomposition import PCA

from sklearn.neighbors import KNeighborsClassifier

main = tkinter.Tk()

main.title("Integrating Domain Knowledge Into Deep Networks for Lung Ultrasound With Applications to COVID-19")

main.geometry("2000x2000")

global filename

global classifier

global svm\_er, knn\_er

global X, Y

global X\_train, X\_test, y\_train, y\_test

global pca

def uploadDataset():

global filename

filename = filedialog.askdirectory(initialdir=".")

text.delete('1.0', END)

text.insert(END,filename+" loaded\n");

def splitDataset():

global X, Y

global X\_train, X\_test, y\_train, y\_test

global pca

text.delete('1.0', END)

X = np.load('features/X.txt.npy')

Y = np.load('features/Y.txt.npy')

X = np.reshape(X, (X.shape[0],(X.shape[1]\*X.shape[2]\*X.shape[3])))

pca = PCA(n\_components = 100)

X = pca.fit\_transform(X)

print(X.shape)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size=0.2)

text.insert(END,"Total CT Scan Images Found in dataset : "+str(len(X))+"\n")

text.insert(END,"Train split dataset to 80% : "+str(len(X\_train))+"\n")

text.insert(END,"Test split dataset to 20% : "+str(len(X\_test))+"\n")

def executeSVM():

global classifier

global svm\_er

text.delete('1.0', END)

cls = svm.SVC()

cls.fit(X\_train, y\_train)

predict = cls.predict(X\_test)

svm\_er = 1 - (accuracy\_score(y\_test,predict))

classifier = cls

text.insert(END,"SVM Error Rate : "+str(svm\_er)+"\n")

def executeKNN():

global knn\_er

cls = KNeighborsClassifier(n\_neighbors = 2)

cls.fit(X\_train, y\_train)

predict = cls.predict(X\_test)

knn\_er= 1 - accuracy\_score(y\_test,predict)

text.insert(END,"KNN Error Rate : "+str(knn\_er)+"\n")

def predictDisease():

filename = filedialog.askopenfilename(initialdir="testSamples")

img = cv2.imread(filename)

img = cv2.resize(img, (64,64))

im2arr = np.array(img)

im2arr = im2arr.reshape(64,64,3)

im2arr = im2arr.astype('float32')

im2arr = im2arr/255

test = []

test.append(im2arr)

test = np.asarray(test)

test = np.reshape(test, (test.shape[0],(test.shape[1]\*test.shape[2]\*test.shape[3])))

test = pca.transform(test)

predict = classifier.predict(test)[0]

msg = ''

if predict == 0:

msg = "Uploaded CT Scan is Normal"

if predict == 1:

msg = "Uploaded CT Scan is Abnormal"

img = cv2.imread(filename)

img = cv2.resize(img, (400,400))

cv2.putText(img, msg, (10, 25), cv2.FONT\_HERSHEY\_SIMPLEX,0.7, (0, 255, 255), 2)

cv2.imshow(msg, img)

cv2.waitKey(0)

def graph():

height = [svm\_er, knn\_er]

bars = ('SVM Error Rate','KMeans Error Rate')

y\_pos = np.arange(len(bars))

plt.bar(y\_pos, height)

plt.xticks(y\_pos, bars)

plt.show()

font = ('times', 14, 'bold')

title = Label(main, text='Integrating Domain Knowledge Into Deep Networks for Lung Ultrasound With Applications to COVID-19')

title.config(bg='deep sky blue', fg='white')

title.config(font=font)

title.config(height=3, width=120)

title.place(x=0,y=5)

font1 = ('times', 12, 'bold')

text=Text(main,height=20,width=150)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)

text.place(x=50,y=110)

text.config(font=font1)

font1 = ('times', 13, 'bold')

uploadButton = Button(main, text="Upload Lung Cancer Dataset", command=uploadDataset)

uploadButton.place(x=50,y=500)

uploadButton.config(font=font1)

readButton = Button(main, text="Read & Split Dataset to Train & Test", command=splitDataset)

readButton.place(x=350,y=500)

readButton.config(font=font1)

svmButton = Button(main, text="Execute SVM Algorithms", command=executeSVM)

svmButton.place(x=50,y=550)

svmButton.config(font=font1)

kmeansButton = Button(main, text="Execute KNN Algorithm", command=executeKNN)

kmeansButton.place(x=350,y=550)

kmeansButton.config(font=font1)

predictButton = Button(main, text="Predict Lung Disease", command=predictDisease)

predictButton.place(x=50,y=600)

predictButton.config(font=font1)

graphButton = Button(main, text="Error Rate Graph", command=graph)

graphButton.place(x=350,y=600)

graphButton.config(font=font1)

main.config(bg='LightSteelBlue3')

main.mainloop()

**11. SCREEN SHOTS**

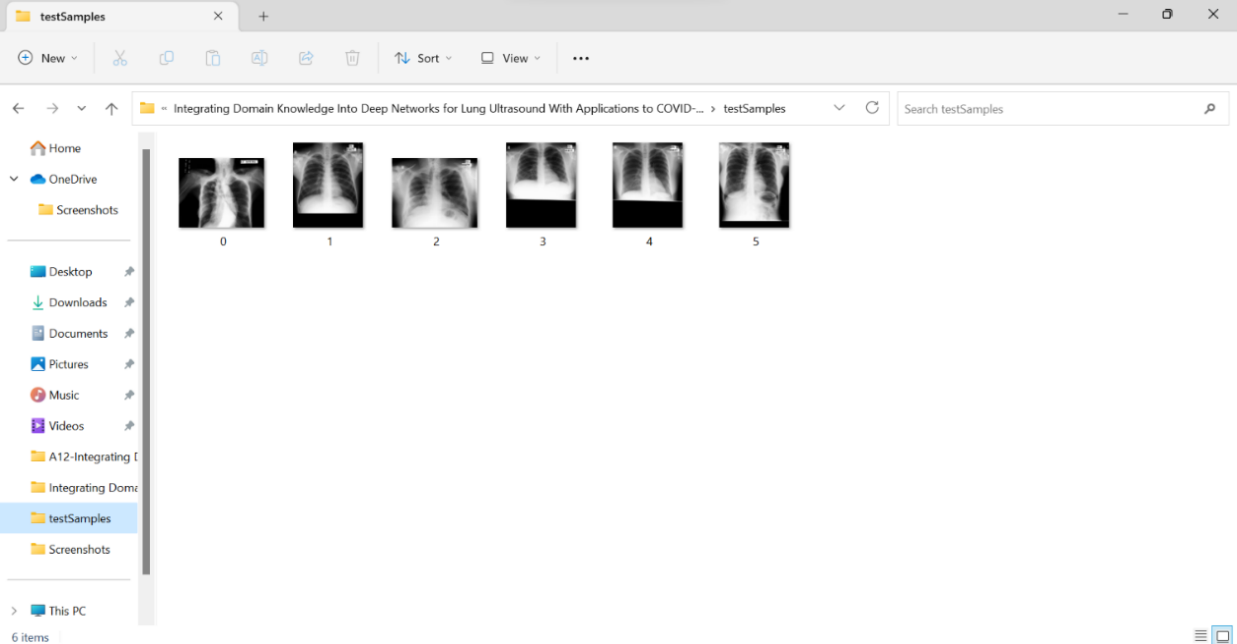


FIG : 11.1.Test Samples

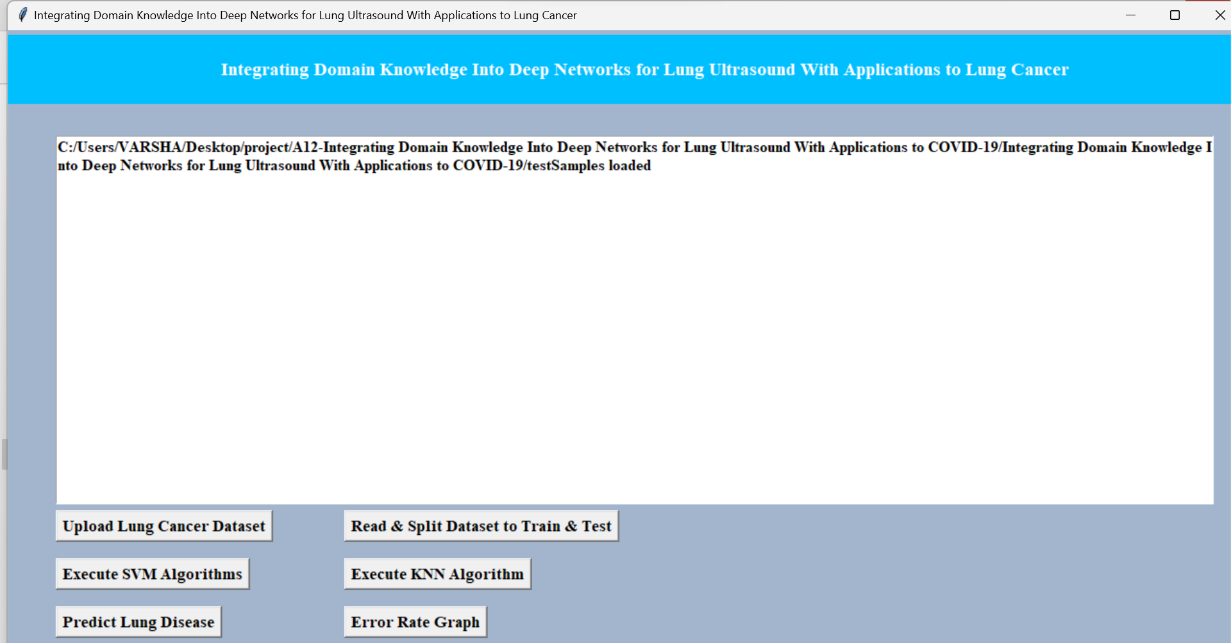


FIG : 11.2 Uploading lung cancer data set

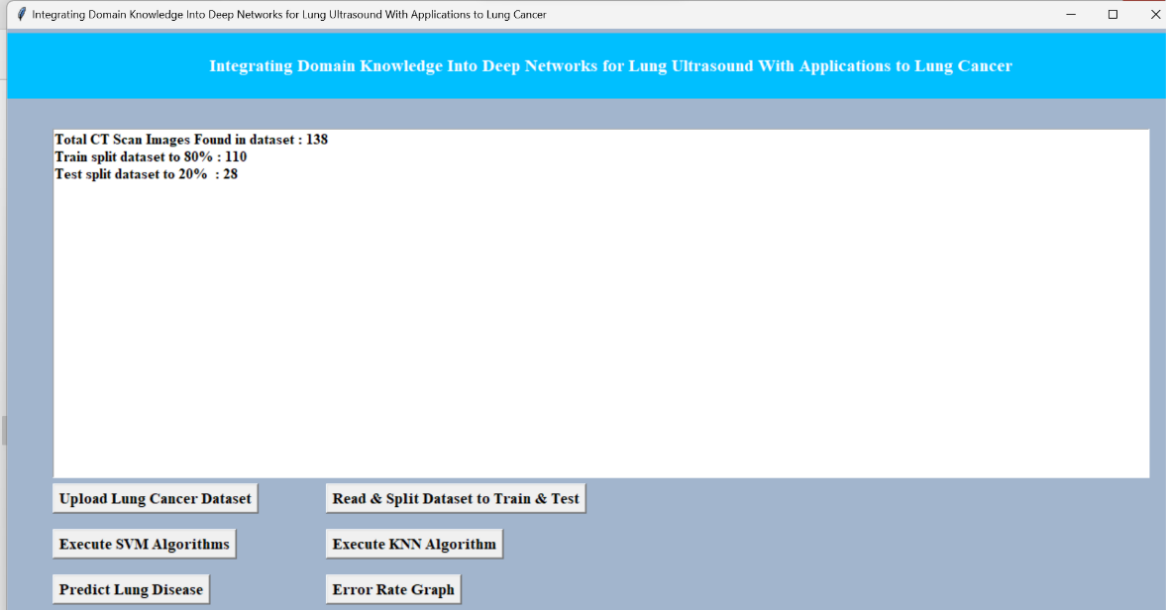
****

FIG : 11.3.Training and Testing Data Set

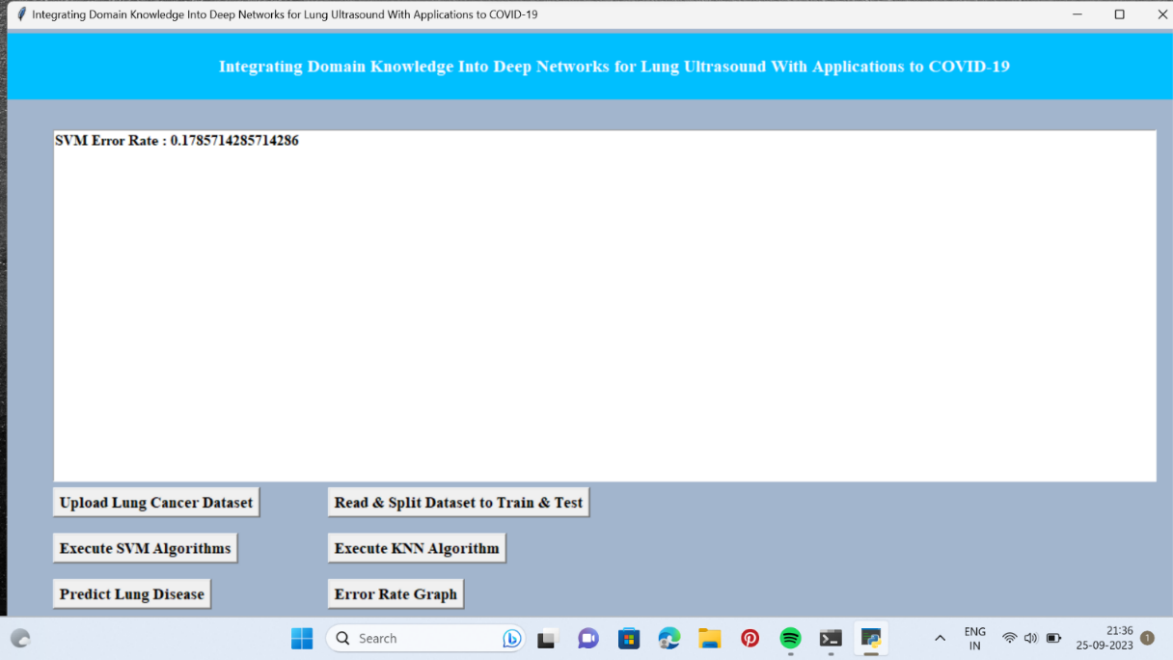
****

FIG : 11.4. Executing SVM Algorithm

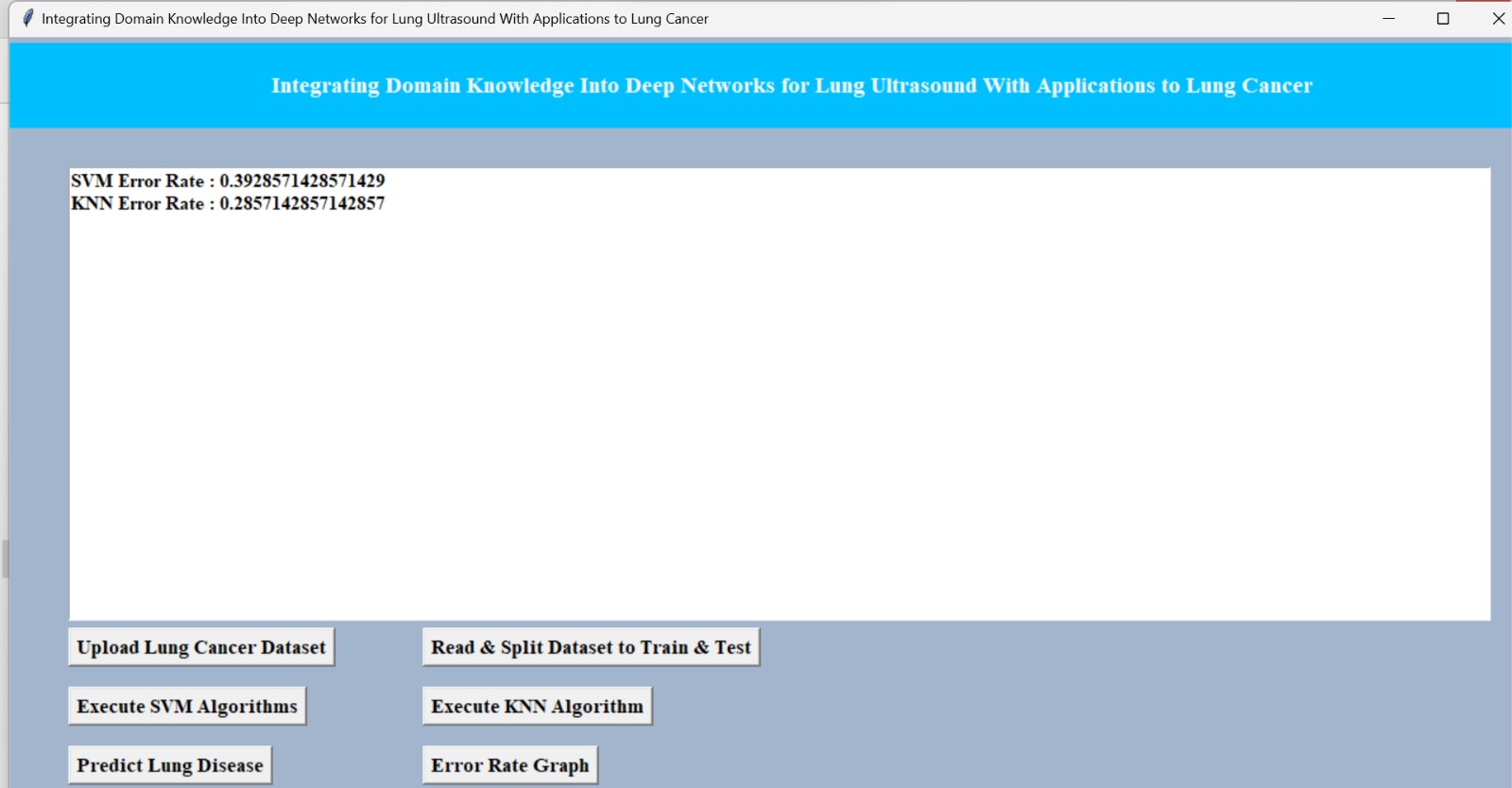


FIG : 11.5.Executing KNN Algorithm

**11.6 Predicting the disease**

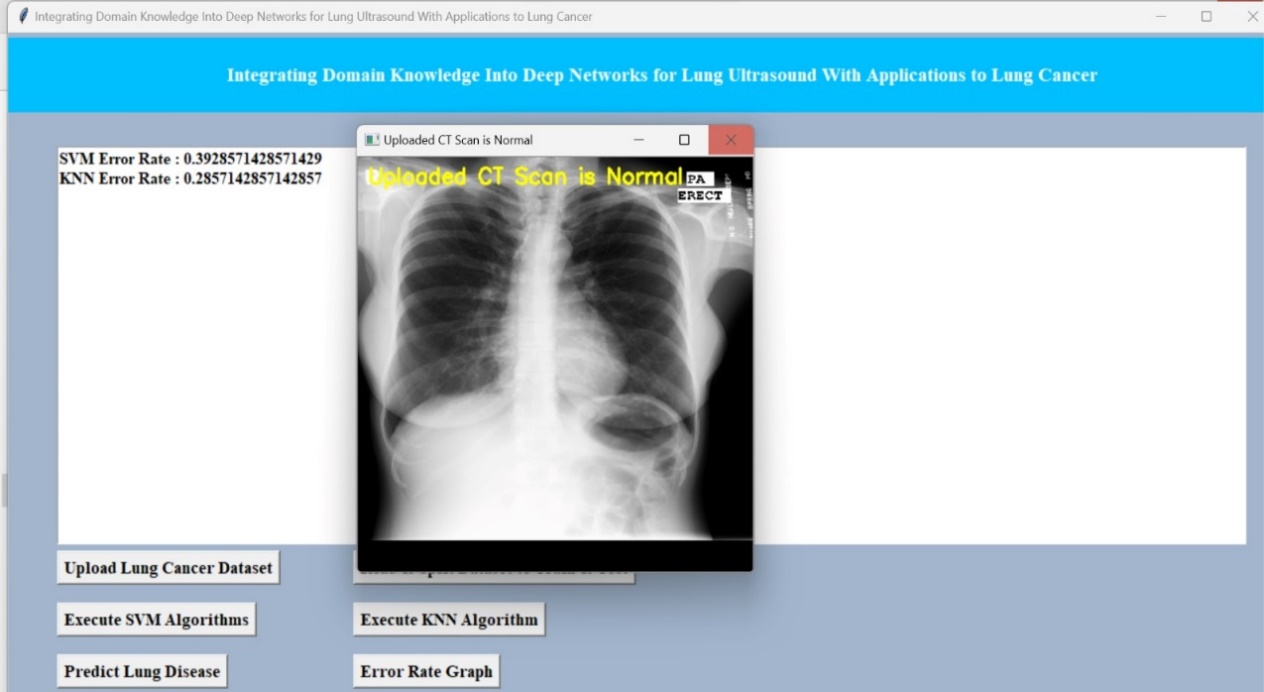


FIG : 11.6.1 Example 1

FIG : 11.6.2 Example 2

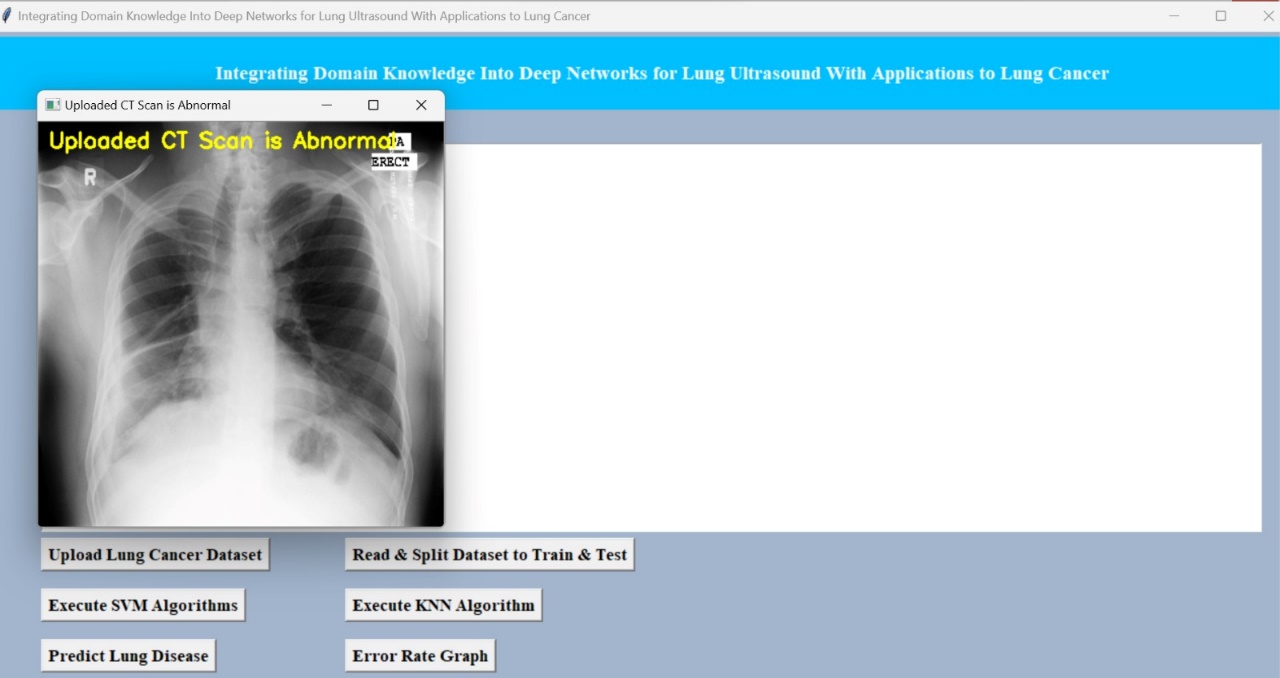
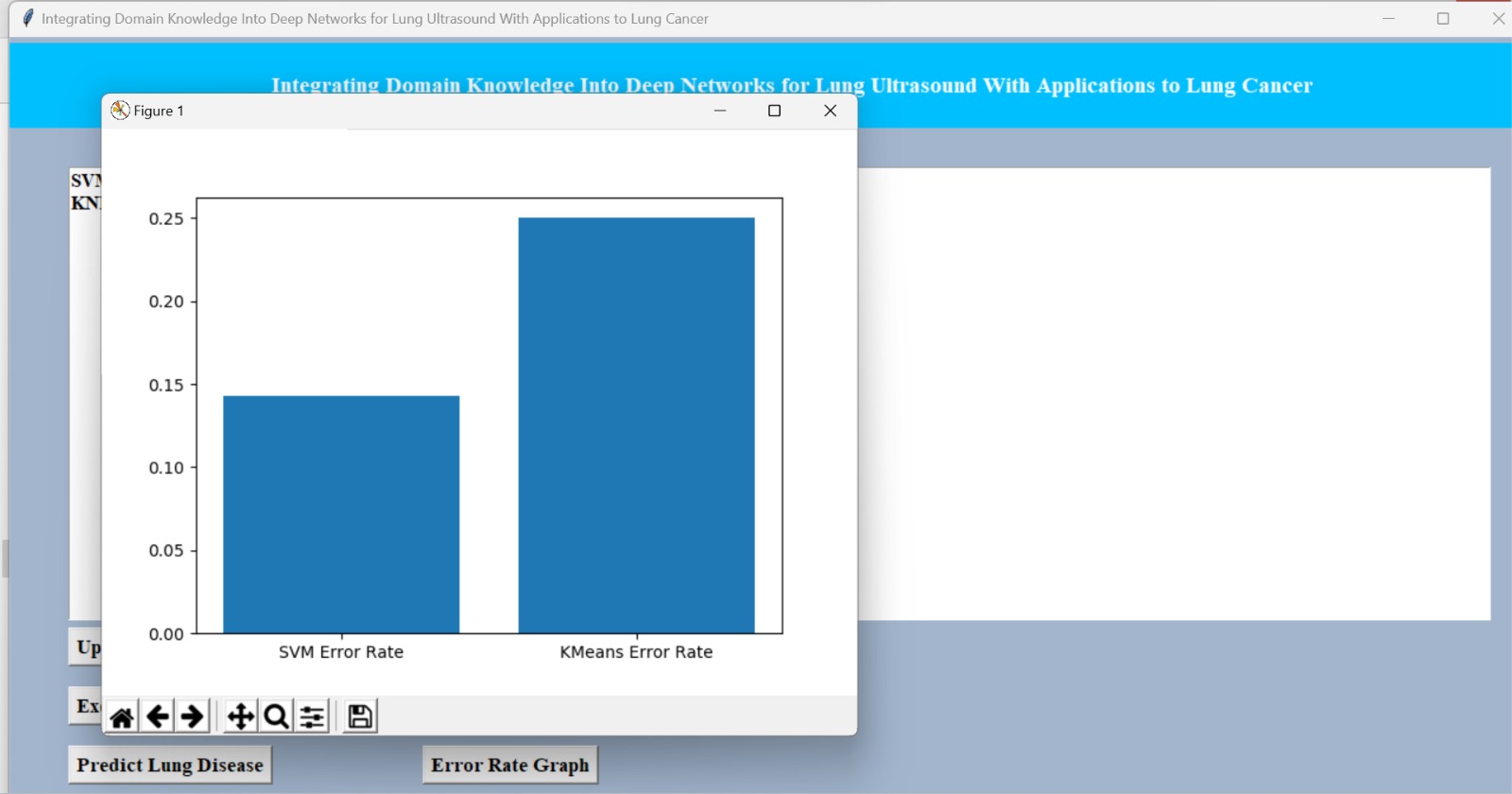


FIG : 11.7 Error rate graph



**12. TESTING**

**SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as

specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a test in which the software under test is treated, as a black box. you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

# Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**13.CONCLUSION**

In the principal period of the venture the Region of Interest in a picture is distinguished. The Identified district is situated in an item. The highlights in the picture are distinguished by utilizing some picture handling system. In second period of the task the component removed information is then used to arrange the picture is destructive or not utilizing a portion of the SVM – bolster vector machine grouping. At that point some boosting calculation is utilized to expand the exactness of the instrument.

In existing paper, a picture handling procedures has been utilized to recognize beginning time lung malignant growth in CT examine pictures. The CT filter picture is pre-prepared pursued by division of the ROI of the lung. Discrete waveform Transform is connected for picture pressure and highlights are extricated utilizing a GLCM. The outcomes are encouraged into a SVM classifier to decide whether the lung picture is carcinogenic or not. The SVM classifier is assessed dependent on a LIDC dataset. In future the advanced level of algorithm is used to increase the level of prediction while we are in process to include the Extreme gradient boosting Algorithm to use the data set more effectively.

**14.REFERENCES**

[1] M. Debois, “TxNxM1 : the anatomy and clinics of metastatic cancer.Boston,” Kluwer Academic Publishers, 2006.

[2]J.M. Fitzpatrick, & M. Sonka, “Medical imaging 2003: image processing. USA: SPIE Society of Photo-Optical Instrumentation Engineers,” 2003.

[3] C. M. Haskell, & J.S. Berek, “ Cancer treatment. Philadelphia: W.B. Saunders, 2001.

[4] K.T.Manivannan, “Development of gray level co-occurrence matrix based support vector machines for particulate matter characterization,”Retrieved fromhttp://rave.ohiolink.edu/etdc/view?acc\_num=toledo1341577486, 2012.

[5] Sathish Kumar R, R. Logeswari, N. Anitha Devi, S. DivyaBharathy Efficient Clustering using ECATCH Algorithm to Extend Network Lifetime in Wireless Sensor Networks. International Journal of Engineering Trends and Technology. 45. 476-481. 10.14445/22315381/IJETT-V45P290 – March 2017.

[6] C. Charalambous, Conjugate gradient algorithm for efficient training of artificial neural networks, IEEE Proceedings 139 (3) (1992) 301–310.

[7] B. H. Boyle, “Support vector machines : data analysis, machine learning, and applications,” Retrievedhttp://public.eblib.com/choice/publicfullrecord.aspx?p=3021500, 2011.

[8] A.A. Abdulla, S.M. Shaharum, Lung cancer cell classification method using artificial neural network, Information Engineering Letters 2 (March) (2012) 50–58.

[9] Sathish Kumar R , Nive tha M, Madhu mita G & Santhoshy P. Image Enhancement using NHSI Model Employed in Color Retinal Images. International Journal of Engineering Trends and Technology. 58. 14-19. 10.14445/22315381/IJETT-V58P203- April 2018.

[10] D. Harini, & D. Bhaskari, “Image retrieval system based on feature extraction and relevance feedback,” ACM, 2 Penn Plaza, Suite 701, New York,NY 10121-0701, 2012.

[11] D. S. Taubman, & M. W. Marcellin, “JPEG2000: image compression fundamentals, standards, and practice,” Boston: Kluwer Academic Publishers,2002.

[12]Sathish Kumar R ,Akthar unissa A ,Koperundevi S & Suganthi S. Enhanced Trust Based Architecture in MANET using AODV Protocol to Eliminate Packet Dropping Attacks. International Journal of Engineering Trends and Technology. 34. 21-27. 10.14445/22315381/IJETT-V34P204- April 2016.

[13] A. Motohiro, H. Ueda, H. Komatsu, N. Yanai, T. Mori,Prognosis of non-surgically treated, clinical stage I lung cancer patients in Japan, Lung Cancer 36 (2002) 65–69.

[14] R.N. Strickland, Tumor detection in non stationary backgrounds, IEEE Transactions on Medical Imaging 13(June) (1994) 491–499.

[15]Sathishkumar R , Dhinesh T, Kathirresh V. Consensus Based Algorithm to Detecting Malicious Nodes in Mobile Adhoc Network. International Journal of Engineering Research and Technology. V6. 10.17577/IJERTV6IS030144 ISSN: 2278-0181- March 2017.