A Project Report on

LUNG CANCER DETECTION USING

IMAGE SEGMENTATION

*is submitted in partial fulfillment of the requirement for the award of the Degree of*

***BACHELOR OF TECHNOLOGY***

*to*

**G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY**

### (Autonomous)

Approved proved by AICTE | NAAC Accreditation with ‘A’ Grade Accredited by NBA (CSE, ECE & EEE) | Permanently Affiliated to JNTUA

### By

**P PRANOY DAVIS (20AT1A05A6) P PAVAN KUMAR REDDY (20AT1A05A2) L MANJUNATH NAIDU (20AT1A0583) PD MAHAMMAD HUSSAIN (20AT1A0575)**

**Under the Guidance of**

**MR VIJAYA CHANDRA RAOM Tech.**

**Assistant Professor**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY**

### (Autonomous)

Approved by AICTE | NAAC Accreditation with ‘A’ Grade Accredited by NBA (CSE, ECE & EEE) | Permanently Affiliated to JNTUA Nandikotkur Road, Venkayapalli (V), Kurnool - 518452, Andhra Pradesh

### 2020-2024

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

## G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY

### (Autonomous)

Approved by AICTE | NAAC Accreditation with ‘A’ Grade Accredited by NBA (CSE, ECE & EEE) | Permanently Affiliated to JNTUA Nandikotkur Road, Venkayapalli (V), Kurnool - 518452, Andhra Pradesh



**CERTIFICATE**

This is to certify that the project report entitled **“LUNG CANCER DETECTION USING IMAGE SEGMENTATION”** being submitted by **PRANOY DAVIS (20AT1A05A6) PAVAN KUMAR REDDY (20AT1A05A2) MANJUNATH NAIDU (20AT1A0583) MAHAMMAD HUSSAIN (20AT1A0575)** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering of G. Pullaiah College of Engineering and Technology, Kurnool is a record of bonafide work carried out by them under my guidance and supervision. The results embodied in this project report have not been submitted to any other university or institute for the award of any Degree or Diploma.

Internal Guide Head of the Department

### Mr. VIJAYA CHANDRA RAO M.Tech. Dr. M. SRI LAKSHMI M.Tech.,Ph.D

Assistant Professor Associate Professor

Date of Viva-Voce

INTERNAL EXAMINER EXTERNAL EXAMINER

# ACKNOWLEDGEMENT

We thank our project supervisor, **Mr VIJAYA CHANDRA RAO M.Tech.** for his guidance, valuable suggestions, and support in the completion of the project.

We would like to express our deep sense of gratitude and our sincere thanks to HOD **Dr. M. SRI LAKSHMI M.Tech.,Ph.D.** Department of Computer Science and Engineering, G. Pullaiah College of Engineering and Technology, Kurnool for providing the necessary facilities and encouragement towards the project work.

We owe indebtedness to our principal **Dr. C. SREENIVASA RAO M.Tech., Ph.D.** G. Pullaiah College of Engineering and Technology, Kurnool for providing us the required facilities.

We are extremely grateful to Chairman **Mr. G. V. M. MOHAN KUMAR** of G. Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh for their good blessings. We gratefully acknowledge and express our thanks to the teaching and non-teaching staff of the CSE Department.

### Project Associates

|  |  |
| --- | --- |
| **P PRANOY DAVIS** | **(20AT1A05A6)** |
| **P PAVAN KUMAR REDDY**  **PD MAHAMMAD HUSSAIN** | **(21AT5A05A2)**  **(20AT1A0575)** |
| **L MANJUNATH NAIDU** | **(20AT1A0583)** |

# ABSTRACT

The computer-based process of identifying the boundaries of lung from surrounding thoracic tissue on computed tomographic images, which is called segmentation, is a vital first step in radiologic pulmonary image analysis. Many algorithms and software platforms provide image segmentation routines for quantification of lung abnormalities; however, nearly all of the current image segmentation approaches apply well only if the lungs exhibit minimal or no pathologic conditions. In particular, abnormalities such as pleural effusions and masses often cause inaccurate lung segmentation, which greatly limits the use of image processing methods in clinical and research. In this review, a summary of the current methods for lung segmentation on CT images is provided, with special emphasis on the accuracy and performance of the methods in cases with abnormalities and cases with exemplary pathologic findings. The feasibility of each class and its shortcomings are illustrated with the most common lung abnormalities observed on CT images. The proposed approach expresses a method for segmenting the lung region from lung Computer Tomography (CT) images. This method is proposed to obtain an optimal segmented region

**Keywords**: Lungs Images, K means clustering image segmentation, Marker controlled watershed image segmentation.

### Abstract

**List of Figures and Tables**

### CHAPTER 1 INTRODUCTION 1

1.1 Introduction

### CHAPTER 2 LITERATURE REVIEW 2

2.1 Literature survey

### CHAPTER 3 PROPOSED METHOD AND EXISTING METHOD

* 1. Existing System
  2. Proposed System
  3. System Architecture
  4. System Requirements
     1. Hardware Requirements
     2. Software Requirements
  5. Software Environment
     1. Python
     2. Python Features
     3. Python Libraries
  6. Visual Studio
     1. Importance of Visual Studio
  7. Machine learning
     1. Algorithms
  8. Data Flow Diagram
  9. Modules
  10. Input Design and Output Design
      1. Input Design
      2. Output Design

CHAPTER 4 APPLICATIONS 22

[4.1 Advantages](#_TOC_250003)

CHAPTER 5 SOURCE CODE AND IMPLEMENTATIONS 24

5.1 Source Code

5.2 Implementation

CHAPTER 6 CONCLUSION AND FUTURE SCOPE 29

BIBILIOGRAPHY 32

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Fig No.** | **Name of the Figure** | **Page No** |
| 1 | Block Diagram Of Proposed Method | 5 |
| 2 | Architecture Of Classification | 6 |
| 3 | Data Flow Diagram | 15 |
| 4 | Home Page | 25 |
| 5 | About Page | 26 |
| 6 | Upload Page | 26 |
| 7 | Result Page | 3 |

**CHAPTER 1 INTRODUCTION**

# CHAPTER 1 INTRODUCTION

Lung cancers have been identified as one of the world's most serious causes of death [1]. It is among the most malignant tumors that can affect human wellbeing. Its death rate scores among all tumor deaths, and is also the top killer towards male and female cancer death [2-3]. There have been nearly 1.8 million fresh cases of lung cancer annually (13 percent among all cancers), 1.6 million deaths worldwide (19.4 percent among all cancers). Lung cancer is a proliferation of expanding and developing irregular cells into a tumour. Of the other forms of cancer, the death rate of lung cancer is the greatest. Cigarette smoke induces an approximate 85 percent of cases of lung cancer in males and 75 percent in females. Lung cancer is amongst the most terrible illnesses in the developing countries, with a death rate of 19.4 percent. Lung cancer is among the most dangerous cancer worldwide, with lowest success rate following diagnosis, with a steady rise in casualty count per year [4-6]. Advantages of Fuzzy logic in the earlier predictions will lead to result oriented analysis [5]. Survival of lung cancer as a result of diagnosis is directly related to its progress. Yet individuals have a greater success rate it will be found in the early stages of life. Cancer cells are distributed in blood from the lungs, the lymph fluid that covers the lung tissues. The lymph passes into lymph vessels that discharge through lymph nodes in the lungs and chest region. Examination and treatment of lung disease has become one of the biggest obstacles that humanity faces in recent years. Early tumor diagnosis will reliably promote its survival of vast numbers of life around the world. This paper introduces a method that uses a convolutional neural Network (CNN) to identify the lung tumors as malignant/benign.

# CHAPTER 2 LITERATURE REVIEW

**CHAPTER 2 LITERATURE REVIEW**

**Deep ensemble learning for automatic identification: Silky Sachar, Anuj Kumar.**

The therapeutic nature of medicinal plants and their ability to heal many diseases raises the need for their automatic identifcation. Diferent parts of plants that help in their identifcation include root, fruit, bark, stem but leaf images have been widely used as they are an abundant source of information and are also easily available. This work explores the branch of Artifcial Intelligence, called deep learning, and proposes an Ensemble learning approach to rapidly detect medicinal plants using the leaf image. The medicinal leaf dataset consists of 30 classes. Transfer learning approach was used to initialize the parameters and pretrain Neural networks namely MobileNetV2, InceptionV3, and ResNet50. These component models were used to extract features from the input images and the softmax layer connected to the Dense Layer was used as the classifer to train the models on the concerned dataset. The obtained accuracies were validated using threefold and fvefold crossvalidation. The Ensemble Deep Learning- Automatic Medicinal Leaf Identifcation (EDL-AMLI) classifer based on the weighted average of the component model outputs was used as the fnal classifer. It was observed that the EDL-AMLI outperformed the state-of-the-art pre-trained models such as MobileNetV2, InceptionV3, and ResNet50 by achieving 99.66% accuracy on the test set and average accuracy of 99.9% using threefold and fvefold cross validation.

**Summary:** Automatic detection of medicinal plants opens new doors for the development of medicines to cure diseases that have not yet been cured by allopathy. It will allow the layman to be aware of the plants growing in their surroundings and make utmost use of them to cure common ailments with no possible side efects. Artifcial Intelligence makes this purpose even more achievable. We proposed an Ensemble of deep learning models to automatically detect medicinal plants. The medicinal leaf images were obtained from a medicinal leaf dataset published in Mendeley. By employing Transfer learning.

**Real-Time Identification of Medicinal Plants using Machine Learning Techniques: Sivaranjani.C, Lekshmi Kalinathan , Amutha.R**

The lighting condition of the environment are uncontrolled, so the segmentation of a leaf from the background is considered as a complex task. Here we propose a system which can identify the plant species based on the input leaf sample. An improved vegetation index, ExG-ExR is used to obtain more vegetative information from the images. The reason here is, it fixes a built-in zero threshold and hence there is no need to use otsu or any threshold value selected by the

user. Inspite of the existence of more vegetative information in ExG with otsu method, our ExG-ExR index works well irrespective of the lighting background. Therefore, the ExG-ExR index identifies a binary plant region of interest. The original color pixel of the binary image serves as the mask which isolates leaves as sub-images. The plant species are classified by the color and texture features on each extracted leaf using Logistic Regression classifier with the accuracy of 93.3%.

**Summary:** In this work, we addressed the problem of identifying the medicinal plant species by the analysis of leaf images obtained directly from their habitat and irrespective of lighting conditions. The fixed zero threshold, ExG-ExR vegetative index is successfully tested for image dataset. The result shows that the algorithm can adequately segment the leaf region. This method worked well in images with reflection. The feature extraction based on the color and texture features is done. The classification of medicinal plant species is done by using Weka and the accuracy of 93.3% is measured. In future we have planned to design and develop a system which automatically identifies plant species through the analysis of not only the leaf images also the other parts of the plant acquired directly in their habitat irrespective of complex backgrounds and various lighting condition. R.

**Identification of Medicinal Plants using Deep Learning: R. Upendar Rao, M. Sai Lahari, K. Pavana Sri and team**

Identification of the correct medicinal plants that goes in to the preparation of a medicine is very important in ayurvedic, folk and herbal medicinal industry. The main features required to identify a medicinal plant is its leaf shape, color and texture. Color and texture from both sides of the leaf contain deterministic parameters to identify the species. In this project we explore feature vectors from both the front and back side of a green leaf along with morphological features to arrive at a unique optimum combination of features that maximizes the identification rate. A database of medicinal plant leaves is created from scanned images of front and back side of leaves of commonly used medicinal plants. The leaves are classified based on the shape and dimension combination. It is expected that for the automatic identification of medicinal plants this system will help the community people to develop their knowledge on medicinal plants, help taxonomists to develop more efficient species identification techniques and also participate significantly in the pharmaceutical drug manufacturing

**Summary:** Plants are necessary for human survival. Herbs, particularly, are employed by indigenous populations as folk medicines from old period. Herbs are typically recognized by clinicians based on decades of intimate sensory or olfactory experience. Recent improvements in analytical technology have made it much easier to identify herbs depending on scientific

evidence. This helps a lot of individuals, particularly those are not used to recognising herbs. additionally for time-consuming methods, laboratory-based analysis necessitates expertise in sample healing and data explanation. As a result, a simple and reliable method for identifying herbs is required. Herbal identification anticipated to benefit from the combination of computation and statistical examination. This non-destructive technique will be the preferred approach for quickly identifying herbs, especially for individuals who cannot able to use expensive analytical equipment. This work reviews about different methods for plants recognition and also reviews their advantages and disadvantages.

# CHAPTER 3

**PROPOSED METHOD AND EXISTING METHOD**

# CHAPTER 3

**PROPOSED METHOD AND EXISTING METHOD**

**3.1 Existing System:**

In the existing there are methods implemented to classify lung disesase classification in deep learning. In method we are performing the classification of lung cancer identification using vgg16 of deep learning along with the Machine learning methods. As image analysis based approaches for classification of medical images.

**Disadvantages:**

* Less accuracy.
* More Losses.

**3.2 Proposed system:**

In purposed method we are performing the classification of either the image is lung cancer or normal identification using Mobile net and CNN of deep learning along with the Machine learning methods. As image analysis based approaches for medical lung classification and authentication. Hence, proper classification is important for the lung dataset that which will be possible by using our proposed method. Block diagram of proposed method is shown below.

**Block Diagram:**

A diagram of a software development

Description automatically generated

**Fig 1. Block diagram of proposed method**

**Advantages**:

* Accurate classification
* Less complexity
* High performance
* Easy Identification

**3.3 ARCHITECTURE**

A diagram of a computer

Description automatically generated

**FIG.2. Architecture of classification**

## SYSTEM REQUIREMENTS

### HARDWARE REQUIREMENTS

* + - System : i3 Processor or above
    - Hard Disk : 1 TB.
    - Monitor : 15‟LED
    - Input Devices : Keyboard, Mouse
    - Ram : 8GB.

### SOFTWARE REQUIREMENTS

* + - Operating system : Windows
    - Coding Language : Python
    - IDE : Visual Studio Code

## SOFTWARE ENVIRONMENT

* + 1. **Python:**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

* + - * **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
      * **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
      * **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
      * **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

## Python Features

Python's features include −

* + - * **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
      * **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
      * **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
      * **A broad standard library** − Python's bulk of the library is very portable and cross- platform compatible on UNIX, Windows, and Macintosh.
      * **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
      * **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
      * **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
      * **Databases** − Python provides interfaces to all major commercial databases.
      * **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
      * **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

* + - * It supports functional and structured programming methods as well as OOP.
      * It can be used as a scripting language or can be compiled to byte-code for building large applications.
      * It provides very high-level dynamic data types and supports dynamic type checking.
      * It supports automatic garbage collection.
      * It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

## Python Libraries Used

### Pandas

* + - * Pandas is a library for the Python programming language.
      * Pandas is one of the tools in Machine Learning which is used for data cleaning and analysis. It has features which are used for exploring, cleaning, transforming and visualizing from data.
      * In easy words Pandas used to manipulate or edit like add remove create etc in the Data frame ex:- adding new columns, Remove unwanted columns, visualization purpose.

### Numpy

* + - * Numpy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. Moreover, Numpy forms the foundation of the Machine Learning stack.
      * Numpy stands for „Numerical Python‟. It is an open-source Python library used to perform various mathematical and scientific tasks. It contains multi-dimensional arrays and matrices, along with many high-level mathematical functions that operate on these arrays and matrices.
  1. **Visual Studio**

Visual Studio is a popular integrated development environment (IDE) developed by Microsoft that is used for creating software applications, websites, web services, and other digital solutions. It provides a comprehensive set of tools and features that are designed to help developers write, debug, and test their code efficiently.

Visual Studio supports a wide range of programming languages, including C++, C#, .NET, JavaScript, Python, and many others. It offers various project templates, debugging capabilities, code editors, and other productivity tools that streamline the software development process.

Some key features of Visual Studio include:

* **Code editor**: Visual Studio provides a rich code editing experience with features such as code completion, syntax highlighting, code refactoring, and navigation.
* **Debugger**: Visual Studio includes a powerful debugger that allows developers to identify and fix issues in their code, step through code line by line, and inspect variables and objects during runtime.
* **Integrated version control**: Visual Studio has built-in support for popular version control systems such as Git and Team Foundation Version Control (TFVC), allowing developers to easily manage their source code and collaborate with team members.
* **Project templates**: Visual Studio offers a wide range of project templates for different types of applications, such as Windows desktop applications, web applications, mobile apps, and cloud-based applications, which can save development time and effort.
* **Testing tools**: Visual Studio includes various testing tools, such as unit testing and performance testing, that help developers ensure the quality and reliability of their code.
* **Extensibility:** Visual Studio supports a rich ecosystem of extensions and plugins that can be used to enhance its functionality and customize the development environment to suit individual needs.
* **Cloud development**: Visual Studio integrates with cloud services such as Azure, allowing developers to easily build, deploy, and manage cloud-based applications from within the IDE.

Visual Studio is widely used by professional software developers and organizations for building a wide range of applications, from desktop applications to web services to cloud- based solutions. It is available in different editions, including Visual Studio Community (free for individual developers and small teams), Visual Studio Professional, and Visual Studio Enterprise, each offering different features and capabilities to cater to different development needs..

## Importance of Visual Studio for Projects

Visual Studio is a widely used integrated development environment (IDE) that offers a comprehensive set of tools and features that are crucial for modern software development projects. Here are some key reasons why Visual Studio is important for projects:

* **Efficient project management**: Visual Studio provides a robust set of tools for managing software projects, including project templates, source control integration, project configuration, and project file management. These tools help streamline project setup, organization, and version control, making it easier to manage and collaborate on complex software projects.
* **Powerful code editing and debugging**: Visual Studio offers a rich code editing experience with advanced features such as code completion, syntax highlighting, refactoring, and navigation, which help developers write code faster and with fewer errors. The built-in debugger allows for efficient debugging of code, enabling developers to identify and fix issues quickly.
* **Extensive language and platform support**: Visual Studio supports a wide range of programming languages and platforms, including .NET, C++, C#, JavaScript, Python, and many others. This versatility makes Visual Studio suitable for a variety of projects, from desktop applications to web applications, mobile apps, cloud-based applications, and more.
* **Testing and quality assurance**: Visual Studio includes built-in tools for writing and running unit tests, performance tests, and other types of tests to ensure code quality and reliability. This helps developers identify and fix issues early in the development process, reducing the risk of introducing bugs or performance problems in the final product.
* **Integration with other tools and services:** Visual Studio integrates with a wide range of external tools, libraries, and services, including version control systems, build systems, bug tracking systems, project management tools, cloud services, and more. This integration enables seamless collaboration and enhances the development workflow by automating repetitive tasks and providing access to external resources.
* **Customization and extensibility**: Visual Studio is highly extensible, allowing developers to customize their development environment by installing extensions and plugins from the Visual Studio Marketplace. These extensions can add additional functionality, tools, and templates tailored to specific project requirements, enhancing productivity and flexibility.
* **Cloud development capabilities:** Visual Studio integrates with cloud services such as Microsoft Azure, providing tools and templates for building, deploying, and managing cloud-based applications. This enables developers to develop, test, and deploy cloud-based applications directly from within the IDE, making it easier to build scalable and distributed applications.
* **Community and support**: Visual Studio has a large and active community of developers, users, and Microsoft support resources, including documentation, forums, tutorials, and online resources. This community provides a wealth of knowledge, support, and troubleshooting options, making it easier to resolve issues and stay updated with the latest best practices.

Overall, Visual Studio is a powerful and feature-rich IDE that offers a wide range of tools and features that are essential for efficient and effective software development projects. Its

capabilities in project management, code editing, debugging, testing, extensibility, and integration with other tools and services make it an important tool for developers working on various types of projects, from small applications to large-scale enterprise projects.

## MACHINE LEARNING:

Machine learning is a branch of Artificial Intelligence that involves training algorithms to make predictions or decisions based on patterns found in data. Machine learning models are designed to learn from examples and improve their accuracy over time through experience. There are different types of machine learning approaches, including supervised learning, unsupervised learning, and reinforcement learning. Machine learning has a wide range of applications, from image and speech recognition to natural language processing and autonomous vehicles.

Python is a popular programming language for machine learning due to its simplicity, readability, and extensive libraries and frameworks. Some of the most commonly used Python libraries and frameworks for machine learning include Numpy, Pandas, Scikit-learn, Keras. Python is widely used for machine learning, thanks to its simplicity, powerful libraries, and vast community support. With Python, you can easily implement machine learning algorithms and models, visualize data, and analyses results.

## ALGORITHMS

Here are some of the most commonly used algorithms in each category:

### Supervised Learning:

* + - * Linear Regression
      * Logistic Regression
      * Decision Trees
      * Random Forest
      * Naive Bayes
      * Support Vector Machines (SVM)
      * K-Nearest Neighbors (K-NN)
      * Gradient Boosting

### Unsupervised Learning:

* + - * K-Means Clustering
      * Hierarchical Clustering
      * Principal Component Analysis (PCA)
      * Association Rule Learning (Apriori Algorithm)
      * t-Distributed Stochastic Neighbor Embedding (t-SNE)

### Reinforcement Learning:

* + - * Q-Learning
      * Deep Q-Network (DQN)
      * Policy Gradient
      * Monte Carlo Tree Search (MCTS)

These are just some of the most commonly used algorithms in each category. There are many more algorithms and variations of these algorithms that are used in different applications and domains. The choice of algorithm depends on the problem at hand, the type of data, and the desired outcome.

Machine learning algorithms such as Random Forest, Naive Bayes, and K-Nearest Neighbors can be used for depression detection and prediction purposes.

* + - * **Random Forest** is a machine learning algorithm used for both classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to make a prediction. The Random Forest algorithm works by constructing a multitude of decision trees during the training phase. Each decision tree is constructed using a random subset of the training data and a random subset of the features. In a random forest, a large number of decision trees are created, each of which is trained on a randomly selected subset of the available data. Each tree in the forest independently makes a prediction, and the final prediction is based on the mode or the mean of the predictions made by all the trees in the forest. The random selection of subsets of data and features helps to reduce overfitting and improve the generalization of the model. Additionally, because the trees are constructed independently, the algorithm is highly scalable and can handle large datasets. The number of trees and

the size of the subsets are hyper parameters that can be optimized based on the performance of the model.

* + - * **Naive Bayes algorithm** is a probabilistic classification algorithm based on Bayes' theorem. It is a simple but powerful algorithm that is widely used in various applications such as text classification, spam filtering, and sentiment analysis. In the context of machine learning, Naive Bayes is a probabilistic algorithm that makes predictions by computing the probability of each class given a set of input features. It assumes that each feature is independent of all other features, hence the term "naive". It also works well with high-dimensional data, and can be used for multi-class classification problems.
      * **K-Nearest Neighbors (KNN)** is a machine learning algorithm used for classification and regression tasks. It is a non-parametric algorithm, meaning it does not make any assumptions about the distribution of the data. Instead, it uses the local information of the data points to make a prediction. In classification tasks, the predicted class is the mode of the classes of the k nearest neighbors, while in regression tasks, the predicted value is the mean of the values of the k nearest neighbors. The choice of the value of k is a hyperparameter of the algorithm and can have a significant impact on the accuracy of the predictions. A small value of k can result in a more flexible model that may overfit the data, while a large value of k can result in a more general model that may underfit the data. The k-NN algorithm is simple and easy to implement, and can work well for small datasets or datasets with a low number of features.

In depression detection, the algorithms can be trained on a dataset of features (such as symptoms, behavioral patterns, and demographic information) of individuals with and without depression. The trained algorithm can then be used to classify a new input data point (i.e., a set of features of an individual) as either having depression or not. This can help in early detection and treatment of depression. The algorithms can be trained on longitudinal data (i.e., data collected over time) of individuals with and without depression. The trained algorithm can then be used to predict the likelihood of an individual developing depression in the future based on their current features. This can help in identifying high-risk individuals and providing them with preventive interventions.

## DATA FLOW DIAGRAM:

1. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

A diagram of a system

Description automatically generated

**Fig .3 Data Flow Diagram**

### MODULES

In depression detection, we use four modules data collection, data pre-processing, feature extraction and prediction.

### Data collection

Data collection is the process of gathering information and data from various sources. It involves collecting data in a systematic and structured way using various methods such as surveys, questionnaires, interviews, observation, and experiments. It‟s an important step in the research process, as it provides the necessary information for analysis and interpretation. The quality and accuracy of the data collected can have a significant impact on the validity and reliability of the research findings

### DATA PRE-PROCESSING

Data pre-processing is the process of cleaning, transforming, and preparing raw data for analysis. It is a critical step in the data analysis pipeline because the quality of the output generated by any data analysis technique depends heavily on the quality of the input data.

### FEATURTE EXTRACTION

### In the feature extraction phase, relevant characteristics such as texture, shape, intensity, and size of lung nodules or lesions are extracted from the segmented images. Techniques like histogram analysis, edge detection, and texture analysis are employed to capture key features indicative of lung cancer presence. These features serve as informative descriptors for subsequent analysis and classification. By distilling complex image information into meaningful features, the stage is set for building predictive models that accurately detect lung cancer based on the segmented regions within the images. Optional post-processing techniques can refine predictions by addressing false positives or false negatives, further enhancing the system's accuracy and reliability. Upon successful validation, the model is poised for deployment in clinical settings, integrated seamlessly with healthcare infrastructure for real-world lung cancer detection applications.

### PREDICTION

In the prediction phase of lung cancer detection via image segmentation, a machine learning or deep learning model is trained on the extracted features or segmented images using labeled data. The dataset is divided into training, validation, and testing sets for robust model evaluation. The trained model undergoes rigorous evaluation on the validation set to optimize hyperparameters and prevent overfitting. Subsequently, the model's performance is assessed on unseen testing data, evaluating metrics such as accuracy, sensitivity, specificity, and ROC curve Post-processing techniques may be applied to refine predictions, mitigating false positives or false negatives and enhancing overall system accuracy. Once successfully validated, the model is rimed for deployment in clinical settings, integrated seamlessly with healthcare infrastructure.

## 3.10INPUT DESIGN AND OUTPUT DESIGN

### 3.10.1INPUT DESIGN

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

* Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −
* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding

-What are the inputs needed for the system?

How end users respond to different elements of forms and screens?

### Objectives for Input Design:

The objectives of input design are

To design data entry and input procedures

To reduce input volume

To design source documents for data capture or devise other data capture methods

To design input data records, data entry screens, user interface screens, etc.

To use validation checks and develop effective input controls.

**Output Design:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

### Objectives of Output Design:

The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.

### To make the output available on time for making good decisions.

### SYSTEM STUDY

A system study of depression detection involves a comprehensive analysis of data collection, pre-processing, feature selection, machine learning, statistical techniques, model evaluation, validation, ethical considerations, clinical integration, deployment and maintenance, and stakeholder involvement to development.

* + Data Sources: The first step in a system study of depression detection involves identifying the various sources of data that can be used for depression detection. This may include self-reported data from individuals, such as questionnaires or surveys, as well as objective data, such as physiological measurements or social media posts.
  + Data Pre-processing: The collected data needs to be pre-processed to ensure its quality and suitability for analysis. This may involve cleaning, normalization, and feature extraction from the raw data to convert it into a structured format that can be used for further analysis
  + Feature Selection: Relevant features need to be selected from the pre-processed data. These features could include demographic information, symptoms of depression, lifestyle factors, and other relevant variables that can be indicative of depression
  + Machine Learning Algorithms: Machine learning algorithms can be used to develop models that can detect depression based on the selected features. Different types of algorithms, such as logistic regression, support vector machines, and deep learning models like neural networks, can be used for this purpose.
  + Performance Metrics: Various performance metrics, such as accuracy, sensitivity, specificity, and area under the ROC curve, can be used to evaluate the effectiveness of the depression detection model.

.

* + Deployment and Integration: Once the model is developed, it needs to be deployed and integrated into a system that can be used for depression detection in real-world settings. This may involve developing user interfaces, integrating the model with other clinical information systems, and ensuring compliance with relevant privacy and security regulations.
  + Human Factors: It is important to consider the human factors involved in the use of the depression detection system. This includes the preferences and needs of the end- users, such as mental health professionals and patients, as well as the ethical and social implications of using such technologies for depression detection.
  + System Evaluation: The final step in a system study of depression detection involves evaluating the overall effectiveness of the system in detecting depression. This may involve conducting usability studies, gathering feedback from end-users, and assessing the impact of the system on patient outcomes.

By studying these aspects, a system study of depression detection can provide insights into the design, development, and evaluation of technological systems that can be used to improve the detection and treatment of depression.

### FEASIBILITY STUDY

The feasibility of the project is analysed 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

* Economic Feasibility
* Technical Feasibility
* Social Feasibility

**Economic 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.

**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.

**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..

# CHAPTER 4 APPLICATIONS

**CHAPTER 4 APPLICATIONS**

The applications of lung cancer detection using image segmentation are diverse and impactful across various domains:

* **Early Detection and Diagnosis**: Image segmentation techniques enable the early detection and accurate diagnosis of lung cancer from medical imaging scans such as X-rays, CT scans, MRI, and PET scans. Early detection is crucial for timely intervention and improved patient outcomes.
* **Treatment Planning and Monitoring**: Segmentation helps in delineating tumor boundaries and tracking changes in tumor size and shape over time. This information is vital for treatment planning, including surgery, radiation therapy, and chemotherapy, as well as for monitoring treatment response and disease progression.
* **Radiomics and Biomarker Discovery:** Image segmentation facilitates the extraction of quantitative imaging features (radiomics) from medical images. These features can be correlated with clinical outcomes and genetic markers, aiding in the discovery of imaging biomarkers for predicting patient prognosis and treatment response.
* **Computer-Aided Diagnosis (CAD):** Segmentation-based CAD systems assist radiologists in interpreting medical images by highlighting suspicious regions and providing quantitative metrics for decision support. This improves diagnostic accuracy and reduces the risk of human error.
* **Research and Clinical Trials**: Segmentation techniques are valuable tools for researchers and clinicians involved in studying lung cancer epidemiology, tumor biology, and therapeutic interventions. They support the design and execution of clinical trials by providing objective measures of treatment efficacy and patient response.
* **Personalized Medicine**: By analyzing individual patient imaging data, segmentation-based approaches contribute to personalized treatment strategies tailored to the specific characteristics of each patient's tumor. This personalized approach improves treatment outcomes and minimizes adverse effects.
* **Education and Training**: Image segmentation serves as an educational tool for medical students, residents, and healthcare professionals, providing visual representations of anatomical structures and pathological conditions. It enhances understanding of lung cancer morphology and aids in the development of diagnostic skills.
* **Telemedicine and Remote Consultation**: Segmentation-based lung cancer detection systems can be integrated into telemedicine platforms, enabling remote consultation and collaboration between healthcare providers. This expands access to specialized diagnostic expertise and improves patient care in underserved areas.

Overall, lung cancer detection using image segmentation has transformative implications for clinical practice, research, and patient care, driving advances in precision medicine and improving outcomes for individuals affected by this devastating disease.

## Advantages

There are potential advantages to predicting depression using quiz-based and sentiment-based tests, as proposed in the project you mentioned. Some potential advantages of depression prediction include:

* Early Detection: Image segmentation facilitates the early detection of lung cancer, allowing for timely intervention and improved treatment outcomes. Early detection is associated with higher survival rates and better prognosis for patients.
* Accurate Diagnosis: Segmentation techniques provide precise delineation of tumor boundaries and identification of suspicious regions within medical imaging scans. This enhances the accuracy of lung cancer diagnosis, reducing the risk of false positives and false negatives.
* Objective Analysis: Image segmentation enables objective and quantitative analysis of imaging data, reducing reliance on subjective interpretations by healthcare professionals. This improves diagnostic consistency and reliability across different observers.
* Treatment Planning: Segmentation-based analysis assists in treatment planning by providing detailed information about tumor size, shape, and location. This helps clinicians determine the most appropriate treatment strategies, including surgery, radiation therapy, and chemotherapy.

Overall, lung cancer detection using image segmentation offers numerous advantages, ranging from early detection and accurate diagnosis to personalized treatment and research advancements. These benefits contribute to improved patient outcomes and quality of care in the fight against lungcancer.

# CHAPTER 5

# SOURCE CODE AND EXPERIMENTAL RESULTS

**CHAPTER 5**

**SOURCE CODE AND EXPERIMENTAL RESULTS**

**5.1 Source Code**

from django.shortcuts import render

# Create your views here.

#Importing Libraries required for the code

from django.shortcuts import render

from .models import Monument

import numpy as np

import os

from tensorflow.keras.preprocessing import image

from tensorflow.keras.models import load\_model

# Create your views here.

#Creating variables

indexhtml='index.html'

abouthtml='about.html'

uploadhtml='upload.html'

resulthtml='result.html'

#Rendering Pages

def home(request):

    return render(request,indexhtml)

def about(request):

    return render(request,abouthtml)

#Loading the Model

#Saving the image file uploaded

#Using the saved file to predict

#Assigining the message to predictions

def upload(request):

    pathss=os.listdir("app/dataset/test")

    classes=[]

    for i in pathss:

        classes.append(i)

    if request.method=='POST':

        m1 = int(request.POST['alg'])

        file=request.FILES['data']

        img=Monument(image=file)

        img.save()

        if m1==1:

            path="app/static/saved/"+ img.filename()

            path1="/static/saved/"+ img.filename()

            models=load\_model("app/models/CNN\_1.h5")

        elif m1==2:

            path="app/static/saved/"+ img.filename()

            path1="/static/saved/"+ img.filename()

            models=load\_model("app/models/mobilenet.h5")

        x=image.load\_img(path,target\_size=(224,224))

        x=image.img\_to\_array(x)

        x=np.expand\_dims(x,axis=0)

        x/=255

        results=models.predict(x)

        b=np.argmax(results)

        prediction=classes[b]

        return render(request,resulthtml,{'result':prediction,'path':path1})

    return render(request,uploadhtml)

## 5.2 IMPLEMENTATION

A screenshot of a computer

Description automatically generated

Fig 4 Home Page

A screenshot of a computer

Description automatically generated

Fig.5 About Page

A screenshot of a computer

Description automatically generated

Fig .6 Upload Page

**Results**

To acquire the very best version accuracy, the statistics ought to be thoroughly cleaned and pre-processed till it's far nicely perfect. For this we used a Python library along with NumPy, pandas. In order to reap the excellent result for our work, we needed to skip each of our data thru unique gadget gaining knowledge of algorithms along with mini batch gradient, logistic regression, and naive bayas.

A screenshot of a computer screen

Description automatically generated

Fig.7 Result Page

**TEST CASES:**

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Result** |
| Input text | Tested for the classification of Lung Cancer | Success |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **Test cases** | **I/O** | **Expected O/T** | **Actual O/T** | **P/F** |
| 1 | Read the dataset. | Dataset path. | Dataset need to read successfully. | Dataset fetched successfully. | P |
| 2 | Performing pre-processing on the dataset | Pre-processing part takes place | Pre-processing should be performed on dataset | Pre-processing successfully completed. | P |
| 3 | Model Building | Model Building for the clean data | Need to create model using required algorithms | Model Created Successfully. | P |
| 4 | Classification | Input image provided. | Output should be Lung Classification. | Model classified successfully | P |

**TEST CASES MODEL BUILDING:**

# CHAPTER 6 CONCLUSIONS & FUTURE SCOPE

**Conclusion**

# CHAPTER 6 CONCLUSIONS AND FUTURE SCOPE

Lung cancer detection through image segmentation offers:

* Early detection, facilitating timely intervention and improved treatment outcomes.
* Accurate diagnosis, with precise delineation of tumor boundaries and reduced risk of false positives/negatives.
* Objective analysis, providing quantitative metrics and enhancing diagnostic consistency.
* Tailored treatment planning, utilizing detailed information about tumor size, shape, and location.
* Personalized medicine, optimizing treatment strategies based on individual patient characteristics.
* Advancements in research, including biomarker discovery and evaluation of treatment outcomes.
* Enhanced efficiency, streamlining workflow and resource allocation for healthcare providers.
* Integration with technology, facilitating seamless data sharing and collaboration among healthcare professionals.
* Educational value, aiding in medical training and professional development in oncology and diagnostic imaging.

Moreover, the integration of segmentation-based systems with healthcare technology enables seamless data sharing and collaboration among healthcare professionals, facilitating efficient and effective patient care. As we continue to refine and innovate in this area, lung cancer detection through image segmentation holds promise for revolutionizing clinical practice, driving advancements in precision medicine, and ultimately, making a significant impact.

## Future work:

The future scope for lung cancer detection using image segmentation is promising, with several avenues for further development and enhancement:

* **Advanced Segmentation Techniques:** Continued research into advanced segmentation algorithms, such as deep learning-based approaches, can improve the accuracy and efficiency of tumor delineation in lung imaging. This includes exploring techniques like convolutional neural networks (CNNs), generative adversarial networks (GANs), and attention mechanisms.
* **Multi-Modal Fusion:** Integrating multiple imaging modalities, such as CT, MRI, and PET scans, can provide complementary information for more comprehensive tumor characterization. Future projects can focus on developing fusion techniques to combine information from different modalities for improved diagnostic accuracy.
* **Real-Time Processing:** The development of real-time segmentation algorithms can enable instantaneous analysis of lung scans during imaging procedures or screenings. This would facilitate prompt decision-making by healthcare professionals and expedite patient care.
* **Automated Lesion Classification:** In addition to segmentation, future projects can focus on developing automated lesion classification systems that categorize detected abnormalities into benign or malignant lesions. This can further assist radiologists in making accurate diagnoses and treatment decisions.
* **Integration with Clinical Decision Support Systems:** Integrating segmentation-based lung cancer detection systems with clinical decision support systems (CDSS) can enhance diagnostic accuracy and assist clinicians in treatment planning. This integration can provide tailored treatment recommendations based on the segmented tumor characteristics and patient-specific data.
* **Quantitative Image Analysis:** Expanding the scope of quantitative image analysis beyond tumor segmentation to include radiomic feature extraction can provide additional insights into tumor heterogeneity, treatment response, and prognosis. This can aid in the development of predictive models for patient outcomes and personalized treatment strategies.
* **Validation and Clinical Trials:** Future projects should focus on rigorous validation of segmentation algorithms using large-scale datasets and multicenter studies. Additionally, collaboration with clinicians and participation in clinical trials can further validate the utility and effectiveness of segmentation-based lung cancer detection systems in real-world clinical settings.
* **Accessibility and Affordability**: Efforts should be made to ensure the accessibility and affordability of segmentation-based lung cancer detection systems, particularly in resource-limited settings. This may involve the development of open-source software platforms, cloud-based solutions, and low-cost hardware options.

In summary, the future scope for lung cancer detection using image segmentation is characterized by ongoing advancements in algorithm development, multi-modal fusion, real-time processing, automated classification, integration with clinical decision support systems, quantitative image analysis, validation, and accessibility. These developments hold the potential to revolutionize lung cancer diagnosis and management, ultimately improving patient outcomes and reducing the burden of this disease in the most probable future outcomes effectively.

# BIBILIOGRAPHY

[1]. Parekh J, Chanda S (2010) Antibacterial and phytochemical studies on twelve species of Indian medicinal plants. Afr J Biomed Res 10:175–181. <https://doi.org/10.4314/ajbr.v10i2.50624>

[2]. Benarba B, Pandiella A (2018) Colorectal cancer and medicinal plants: principle fndings from recent studies. Biomed Pharmacother 107:408–423. <https://doi.org/10.1016/j.biopha.2018.08.006>

[3]. Adhikari B, Marasini BP, Rayamajhee B et al (2021) Potential roles of medicinal plants for the treatment of viral diseases focusing on COVID-19: a review. Phyther Res 35:1298–1312. https:// doi.org/10.1002/ptr.6893

[4]. de Freitas Junior LM, de Almeida EB (2017) Medicinal plants for the treatment of obesity: ethnopharmacological approach and chemical and biological studies. Am J Transl Res 9:2050–2064

[5]. Wäldchen J, Mäder P (2018) Plant species identifcation using computer vision techniques: a systematic literature review. Arch Comput Methods Eng 25:507–543. https://doi.org/10.1007/ s11831-016-9206-z

[6]. Sachar S, Kumar A (2020) Survey of feature extraction and classifcation techniques to identify plant through leaves. Expert Syst Appl. <https://doi.org/10.1016/j.eswa.2020.114181>

[7]. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22:1345–1359. [https://doi.org/10.1109/TKDE. 2009.191](https://doi.org/10.1109/TKDE.%202009.191)

[8]. Lee SH, Chan CS, Mayo SJ, Remagnino P (2017) How deep learning extracts and learns leaf features for plant classifcation. Pattern Recognit 71:1–13. <https://doi.org/10.1016/j.patcog.2017.05.015>

[9]. Grinblat GL, Uzal LC, Larese MG, Granitto PM (2016) Deep learning for plant identifcation using vein morphological patterns. Comput Electron Agric 127:418–424. [https://doi.org/10.1016/j. compag.2016.07.003](https://doi.org/10.1016/j.%20compag.2016.07.003)

[10]. Gogul I, Kumar VS (2017) Flower species recognition system using convolution neural networks and transfer learning. 2017 4th Int Conf Signal Process Commun Networking. ICSCN 2017:1–6. <https://doi.org/10.1109/ICSCN.2017.8085675>

[11]. Liu Z, Zhu L, Zhang XP et al (2015) Hybrid deep learning for plant leaves classifcation. Lect Notes Comput Sci (Include Subser Lect Notes Artif Intell Lect Notes Bioinform) 9226:115–123. https://doi.org/10.1007/978-3-319-22186-1\_11