DETECTION OF LUNG CANCER TYPES USING CT SCANS

A.A Nimesha Hansani Amarasinghe

219182P

Master of Science in Computer Science Specialized in Data Analytics and Engineering

Department of Computer Science and Engineering

University of Moratuwa
Sri Lanka

April 2023

DETECTION OF LUNG CANCER TYPES USING CT SCANS

A.A Nimesha Hansani Amarasinghe

219182P

Thesis report submitted in partial fulfilment of the requirements for the degree Master of Science in Computer Science specialization in Data Analytics and Engineering

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

April 2023

DECLARATION

I declare that this is my own work, and this thesis/dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other University or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. I retain the right to use this content in whole or part in future works (such as articles or books)

Signature: Date:

The above candidate has carried out research for the master's thesis under my supervision. I confirm that the declaration made above by the student is true and correct.

Name of the supervisor: Dr.Thanuja Ambegoda

Signature: Date:

ABSTRACT

Lung cancer is a devastating global health issue that can be fatal if not detected early. In order to increase the chances of successful treatment and prevent the loss of lives, doctors need to identify the type, which the lung cancer is belong to. Currently, CT scans are commonly used in medical practice to detect and diagnose lung tumours. However, implementing deep learning models to identify the lung cancer types, poses a significant challenge. Because acquiring a large number of medical images for each type of the lung cancer can be really difficult.

The effectiveness of deep learning algorithms which were implemented for image recognition are relies on the diversity the of data samples. However, creating a comprehensive dataset of significant size requires a significant amount of human effort for manual labelling, which make it's a time-consuming process. The challenge of requiring a large amount of data samples for each category in traditional deep learning models was addressed in this research by implementing a prototypical network, which is a few-shot learning technique. This method requires only a few samples per category and was used in conjunction with a pre-trained model to extract features from lung CT scans. The accuracy of the model was analysed based on the number of samples per category.

Overall, the results of the study demonstrate that implementing a prototypical network for lung cancer type detection is feasible. Human interpretation of medical images can very among different medical professional. Therefore, this approach can be act as decision support tool for medical professional which make it more accessible and cost-effective.

KEYWORDS: Prototypical Network, Lung Cancer, Few Shot Learning, Segmentation, VGG16

ACKNOWLEDGEMENT

I want to convey my sincere thanks to Dr. Thanuja Ambegoda for assisting me in identifying a fascinating research topic and for his support and encouragement. His guidance was quite beneficial in terms of creating goals and engaging in the study. I would also like to thank Dr.Upul Narandeniya from Colombo General Hospital who explained to me how to identify lung cancer patients. I'd like to convey my heartfelt gratitude to the Department of Computer Science and Engineering at the University of Moratuwa for their assistance in overcoming this challenge. Finally, I want to express my heartfelt gratitude to my parents and friends for their unwavering support during this journey.,

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CAD: Computer Aided Diagnosis

CNN: Convolution Neural Network

PET: Positron Emission Tomography

DICOM: Digital Imaging and Communication in Medicine

FSL: Few Shot Learning

MRI: Magnetic Resonance Imaging

TCIA: The Cancer Imaging Archive

1 INTRODUCTION

Around the world, lung cancer results in the deaths of 422 individuals every day, making it one of the deadliest types of tumors and this disease develops in lung tissues. The growth of cells in the lungs that is not properly regulated leads to lung cancer. According to the American Cancer Society, there are main two types of lung cancers ,which are non-small cell lung cancer (NSCLC) and small lung cancer.non-small cell lung cancer is the most common kind which accounting 80% to 85% of all the lung malignancies [2].In non-small cell lung cancer has to pass few stages before critical stage and there's no obvious symptoms in the early stages. When it started to show symptoms, it is nearly impossible to save a person's life. Non-small lung cancers can further categorize in to three main sub types. The are Adenocarcinoma, Squamous cell carcinoma and large cell carcinoma.

Recognition of lung cancer type can cause for reducing the mortality rate of patients. Because medical professionals can start the relevant treatments for each type. The use of computer aided design (CAD) systems is becoming more common among doctors as a preferred tool for diagnostic procedures. This has made CAD systems a significant area of research in diagnostic radiology and medical imaging, particularly in helping doctors interpret computed tomography (CT) images. Computer Tomography is the most common and accurate imaging technique that use for diagnosis of lung cancers. Modern CT scanners have the capability to capture as many as 320 CT slices simultaneously. For a regular chest examination, a 2.5 mm slice (2D) is the norm. In a single chest CT scan, there can be approximately 400 slices, with each slice comprising a 512×512 pixels image. In the field of image processing and deep learning, there many impressive breakthroughs in medical image analysis to diagnosis of diseases using medical images. It has proven that automation of medical images recognition is really effective to diagnosis and treatment of diseases using CT scan images, mammograms, chest X-rays. [3], [4]. However, all these various recognition task need large quantity of annotated data.

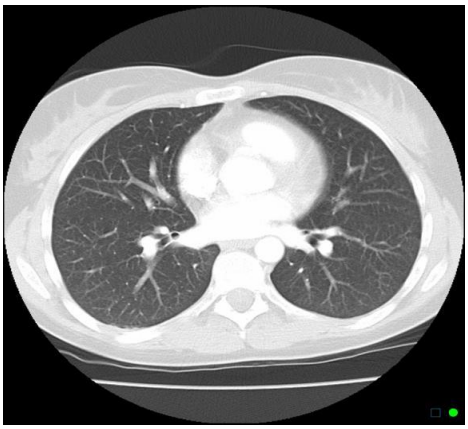


Figure 1.2 CT scan of a normal person. Taken from Radiopedia

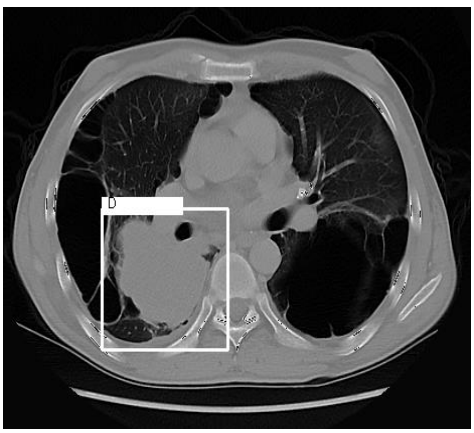


Figure 1.1 CT scan of a patient with Large Cell Carcinoma. Taken from [26]

Deep learning algorithms for lung cancer diagnosis utilizing CT images have been proposed by the motivated researchers [5], [6] with large amount of labeled data. But detecting the lung cancer type is really challenging task, because collecting medical images for each type is infeasible. In certain situations, researchers may not be able to obtain the necessary extensive data that they require for their studies due to confidentiality and security concerns. To overcome this challenge, there has been increasing sub-area in machine learning called few shots learning. The aim of this approach is to achieve favorable learning results despite having a restricted amount of labeled data in the training dataset, which comprises instances of inputs paired with their respective outputs.

The main target of few shots learning is built accurate machine learning models with less training data, and this can be considered as a Meta learning problem. Mainly there are two datasets in few shots learning which are called support set and query set. Support set or query set is comparatively small from the datasets that we use for traditional deep learning methods, but it has samples from every class.

Basically, in few shots learning is to train a function that predicts similarity. Upon completion of training, the acquired similarity function can be utilized to make predictions for new queries. The similarity function can be applied to compare the query with each sample in the support set, and subsequently determine the similarity scores. The sample with the highest similarity score can then be selected and used as the prediction.

2 RESEARCH PROBLEM

Lung cancer occurs as a result of uncontrolled growth of abnormal cells in the lungs, which disrupt the normal functioning and development of lung tissue. Without medical treatment, these abnormal cells continue to grow and form tumors, eventually causing harm to the lungs, responsible for supplying oxygen to the body through the bloodstream. Lung cancers can be classified as non-small cell lung cancer (NSCLC) or small lung cancer. But non-small cell lung cancer is the most common type [7].The subcategories of non-small cell lung cancer are adenocarcinoma, small cell carcinoma, squamous cell carcinoma, and large cell carcinoma.

Adenocarcinoma is a type of cancer that originates in the glandular cells that line certain organs, such as the lungs, colon, pancreas, and prostate. Adenocarcinoma of the lung is one of the most common types of lung cancer, accounting for about 40% of all cases. Adenocarcinoma of the lung usually starts in the outer regions of the lungs and can grow and spread to other parts of the body. A specific type of cancer known as small cell carcinoma mainly affects the lungs, although it can also develop in the prostate or gastrointestinal tract. It is characterized by small cells that rapidly grow and divide and can quickly spread to other organs. Other type of cancer known as squamous cell carcinoma begins in the flat cells called squamous cells, which line various organs including the skin, lungs, and digestive tract. Squamous cell carcinoma of the lung is a type of non-small cell lung cancer, accounting for about 25-30% of

cases. Non-small cell lung cancer encompasses several types, including large cell carcinoma, which can arise in any region of the lung. This form of cancer is identified by the presence of unusually large, abnormal cells that exhibit rapid growth and division, and have the ability to rapidly metastasize to other organs.

Whenever a patient is suspected of having lung cancer, a chest X-ray is frequently the first diagnostic procedure that is carried out. While this imaging modality can provide useful preliminary information about the presence and severity of any lung abnormalities, it is typically insufficient for accurately identifying the cancer. Additional imaging techniques, such as computed tomography (CT) scans, magnetic resonance imaging (MRI), or positron emission tomography (PET) scans, may be required to assess the tumor's size, form, and location, as well as its potential spread to surrounding tissues or organs. These advanced imaging techniques can also aid in treatment planning and monitoring the efficacy of any medicines delivered.

Compared to the other cancers, lung cancers don’t show obvious symptoms in the early stages. But when the patients have their first symptoms, doctors should do a more precise analysis by using chest X-ray or low dose high resolution lung CT screening or Magnetic Resonance Imaging (MRI). In comparison to conventional radiography, Computed Tomography (CT) has proven to be a superior tool for lung screening due to its ability to produce highly detailed, high-resolution images. By utilizing CT, it is possible to detect early-stage lesions that are too small to be visualized through traditional X-ray methods. This advanced imaging technique provides clinicians with a more accurate and comprehensive assessment of the lungs, enabling earlier diagnosis and more effective treatment of lung cancer.

CT scan images have been widely utilized to diagnose a variety of lung disorders that consist of pneumoconiosis, pneumonia, pulmonary edema, and lung cancer. The type of lung cancer must be determined in order to plan treatment. However, because of the massive amount of data collected by CT scans, radiologists find it extremely difficult to identify the accurate type of cancer. Difficulty in accurately identifying cancerous cells can arise from a variety of factors, such as variations in intensity on CT scan images and potential misinterpretation of anatomical structures by doctors and radiologists [8].Since different radiologists have different experience level, they made inconsistent judgement. Recently, computer aided diagnosis has become a promising tool, which assist medical experts to detect the cancer accurately [9]. Most of the systems don’t detect the type of the cancer using CT scan images. Therefore, it is critical to develop new practices to accurately detect the specific type of lung cancer that is closest to the patient's condition when a CT scan is provided.

Deep learning, machine learning, and transfer learning have advanced medical picture analysis to a remarkable level. Because these techniques have become equivalent to humans at categorizing images and identifying objects. For example, a newly introduced deep learning model from Google has demonstrated exceptional effectiveness in diagnosing diabetic retinopathy, a condition that is induced by diabetes and can result in severe vision impairment.

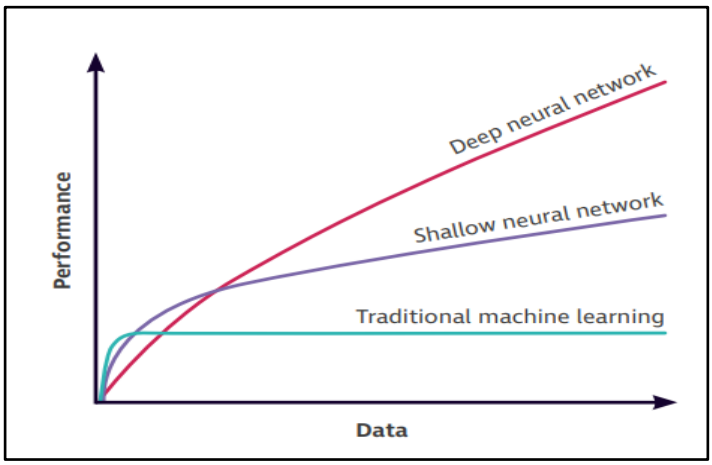


Figure 2.1 A comparison of the performance of traditional machine learning and neural networks when training data is limited. Adapted from [10]

It already got the attention from several medical communities [10].In practice, however, deep learning techniques should have large amount of labeled data to get accurate results. When the amount of data is limited, deep learning models tend to overfit. Due to the overfitting, the model will not be able to generalize unseen examples, resulting in poor performance.

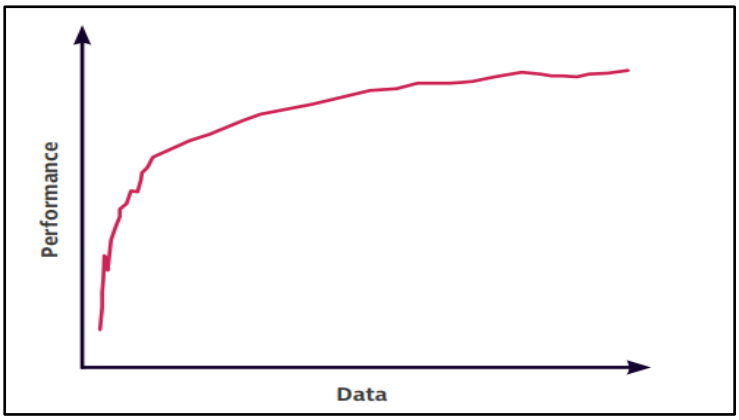


Figure 2.2 The performance of a deep neural network when training data is limited. Adapted from [10]

Motivate researchers has been able to accomplish some achievements with medial image classification using deep learning techniques for lung cancers [11]. It's obvious that same technology can be applied to identify each type of the lung cancer using CT scan images.

But there's a common occurrence of having limited number of datasets in the area of medical image analysis. Because patient health data is well protected by data laws. In order to apply to a deep learning technique, there should be high number of CT scan images which belongs to each type of the cancer. It's really challenging to find out large number of labelled data for each type. Hence, focus on deep learning methods for ease diagnosis of early stage of a lung cancer with satisfactory accuracy. In order to address this issue, we will concentrate on a few-shot learning method, which is a classification approach that deals with situations where there are only a limited number of examples available per class.

3 RESEARCH OBJECTIVES

- Exploring the use of few shots learning method for medical image analysis.
- Implementing a supporting tool for medical professional to identify lung cancer types using CT scans.
- Identifying proper feature extraction methods for lung CT scan images.
- To contribute to the studies on few shots learning for lung cancer type detection.

4 LITRETURE REVIEW

4.1 Detection of Lung Cancers

Several research studies have proposed multiple techniques for identifying lung cancer and classifying lung nodules by applying various image processing and machine learning and image processing technologies. Uren, Prasad and Abeers proposed a model [6] that detect lung cancers using CT scan images. In their suggested model, images were pre-processed using gaussian and median filters, and watershed segmentation was the algorithm used for segmentation. The next stage was to extract the classifier's features, such as area, perimeter, centroid, diameter, and mean intensity. The classifier used to determine whether a nodule is cancerous or benign is a support vector machine. The accuracy of proposed model is 92% and classification learner app taken 53.2 seconds to train.

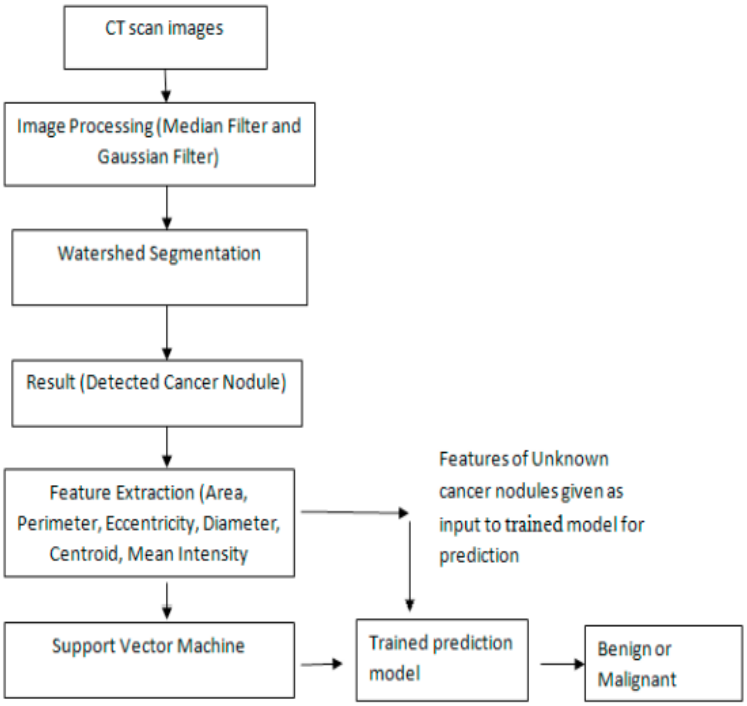


Figure 4.4.1 A proposed model for lung cancer detection. Figure taken from [6]

Noor, Shahid Farid, Saira and Hassan Khan has developed an automated system for detect lung nodule in CT images [12].Their method has two main phase which first one segment the CT scans and in the second phase lung nodules will be detected. The proposed framework employs an innovative approach that is fully automated and requires no user input. The segmentation technique constructs a histogram of the input

image and employs an automatic threshold selection process to identify and remove the outer region of the image. The remaining region is then processed to extract the lungs. The process entails morphological operations and connected component analysis to identify nodules from the segmented lungs in the second phase. To accomplish this, a statistical and shape-based feature training is conducted for the nodule, followed by the application of a support vector machine as the classifier.

Xiaodan Chen S. Feng, and D. Pan. [13], have introduced another version of watershed segmentation. The adaptive threshold algorithm, mathematical morphology, and watershed algorithm have been applied by the author of this paper to enhance the process of lung disease diagnosis via CAD system. The noise in the original Computed Tomography image (CT) was originally decreased using a Gaussian filter and gradient enhancement in this process. The major bronchus and trachea have been eliminated from CT lung scans after the pictures have been segmented in the second step using the OTSU approach. The watershed transform is completed as the last phase, and testing is carried out using a series of CT scans.

Fuzzy local information cluster means automatic segmentation algorithm and back propagation network is another method proposed by Lavanya and Muthu Kannan, in order to detect lung lesion [14].The foremost step of this step was enhancing the CT image quality by removing the noise and as the next step pre-processed images was segmented using FLICM algorithm. Once the segmentation process is completed, the segmented nodule of the lung lesion is employed for feature extraction, which plays a crucial role in obtaining more comprehensive information about the image. Finally, a BPN network is utilized to conduct classification as the final step in detecting the lung lesion.

A group of researchers from India were motivated to develop a system for detecting lung cancer nodules using an active contour model and fuzzy interference system [15]. In this system, the preprocessing phase involves utilizing gray transformation for enhancing image contrast and binarization to prepare the image for segmentation using an active contour model. Various features such as area, mean, entropy, correlation, major and minor axis lengths are extracted and applied to a classification model, which uses the fuzzy inference method. With an accuracy of 94%, the system recognizes the location of lung nodules in CT scan images. However, it is limited in that it does not provide classification of the relevant stage of the cancer.

Main method that we are going apply to our scenario is few-shot learning. A method was proposed to early diagnosis of an eye disease called glaucoma using few shots learning and deep learning by Mijung Kim, Jasper Zuallaert and Wesley De Neve [16].

Applying deep learning techniques to datasets that are quite small and the preservation of image quality throughout the analysis process are two major difficulties that the authors of this study have prioritized addressing in their design. The authors achieved this by first computing the cosine distance between each image in a support set and a target image, and then feeding the support set into an embedding function composed of CNN layers and a bidirectional LSTM layer. In the end, the writers were successful in incorporating 1-shot, 5-shot, 10-shot, and 20-shot learning into their strategy.

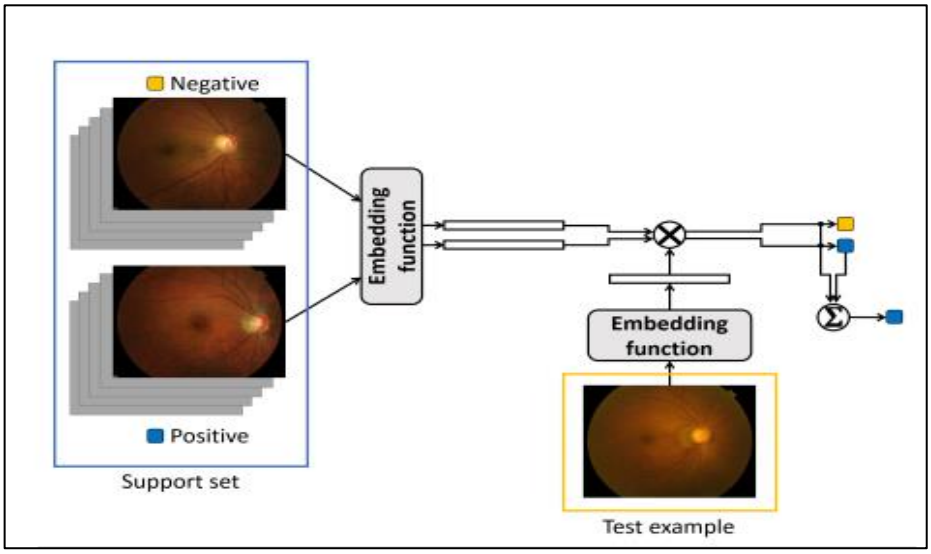


Figure 4.2 Few shot learning with embedding function Figure taken from [16]

4.2 Few Shot Learning

Most of the AI technologies cannot generalize from few examples. In order to learn from limited number of data, “Few shot learning” is proposed [17].Few shot learning approaches can be categorized into meta learning based FSL and non-meta learning based FSL. Furthermore there are more sub methods under meta learning and non-meta learning methods [18].

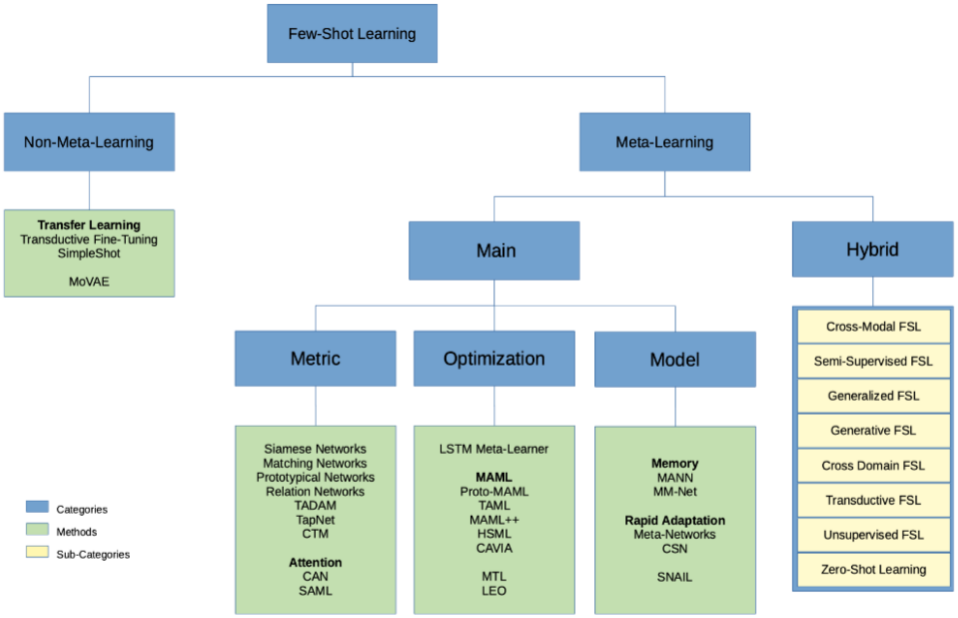


Figure 4.3 Approaches in Few Shot Learning. Figure taken from [18]

The topic is typically described using the terminology of "N-way K-shot learning" in few-shot learning (FSL). During the learning process, this relates to the number of classes and training data samples available. For example, if the job is to categorize dogs and cats, the classification setup is a 2-way classification setup because there are two classes to be focused. Furthermore, if the model is trained on only 5 samples of cats and 5 samples of dogs, the learning configuration is a 2-way 5-shot. N-way denotes the number of classes, whereas K-shot is the number of possible training samples per class.

4.2.1 Metric Based Learning

4.2.1.1 Prototypical Networks

The Prototypical Network is a popular meta-learning approach for few-shot learning that employs a nearest neighbor method, eliminating the need for hyper-parameters in the meta-test phase and resulting in practically insignificant inference time. The effectiveness of the Prototypical Network is assessed by the ratio of between-class variation to within-class variation of features in a support set generated by the meta-trained model. Likewise, no model learning is required during the prototype classifier's testing phase, allowing it to be used with any learned feature extractor. As a consequence, the prototype classifier is a useful and effective first step in resolving a few-shot learning problem.

The paper titled "A Closer Look at Prototype Classifier for Few-shot Image Classification" by Mingcheng Hou and Issei Sato presents a thorough analysis of prototype classifiers for image classification based on FSL [19]. The authors have

presented their empirical analysis of the prototype classifier, which includes a series of experiments on benchmark few-shot image classification datasets. They investigate various aspects of the prototype classifier, such as the impact of different prototype representations, the effect of the number of support examples, and the importance of distance metrics. Their findings demonstrate that modifying the prototype classifier might significantly boost its performance on few-shot picture classification challenges.

One shot learning for radar image recognition with few images has been implemented by a research team at the National University of Defense Technology in Hunan, China. [20]. The initial stage in adjusting radar images to the input of the succeeding network is pre-processing them. The second step is to train a model that maps images to linearly separable feature spaces using more data. In the last stage of the suggested approach, the mean value of homogeneous vectors in the feature space is used as a prototype, and the model is trained to group the feature vectors in the image into their relevant prototypes. The clustering effect can be applied to the target data by training the mapping with auxiliary and target data. The prototypes for each category are matched based on their Euclidean distance to forecast the class of unknown photos. The proposed approach was put to the test using the MSTAR dataset.

In order to create the output of a hidden layer in their algorithm linearly separable, they eliminated the hidden layer's non-linear activation function. The image extraction module in their implementation uses the convolutional layer parameters of the VGG-16 network trained on the ImageNet dataset. They compared the deep transfer learning and proposed methods' accuracy by gradually increasing the number of photos in each class. The relationship between them is depicted in the figure below.

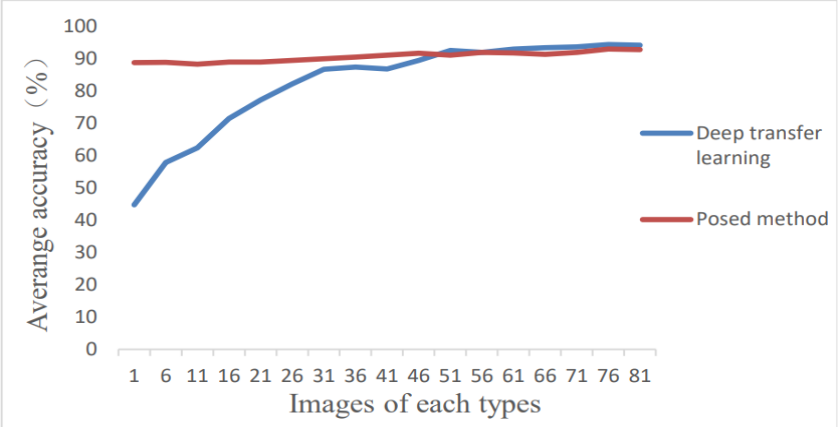


Figure 4.4 Average accuracy based on number of images. Figure taken from [20]

Recently, researchers from the United States and China have proposed a novel approach to few-shot learning, termed the Location-Sensitive Local Prototype Network, which considers critical parameters such as grid size. Their method consists of two main steps. Extraction of location-sensitive local prototypes is the initial stage, and it is accomplished by running support images through a feature encoder to create the relevant feature maps. Then, local prototypes are computed by combining the

support image and masked averages. In the second stage, if the support and query photos have a similar spatial layout, the local prototypes that were extracted from the support set can be matched to their respective locations in the query images. This enables grid-based few-shot segmentation.

For all their experiments they have used VISCERAL dataset [21] which include contrast-enhanced CT scans. For six organ classes—liver, spleen, left kidney, right kidney, left psoas, and right psoas—they have conducted 1-way, 1-shot trials. As the feature encoder, they utilized ResNet-50, which was trained beforehand on ImageNet. In the segment visualizations displayed in the Blow Figure, their algorithm has successfully transferred semantic information from the support to the query images to yield high-quality segmentation results despite variations in organ sizes and shapes.

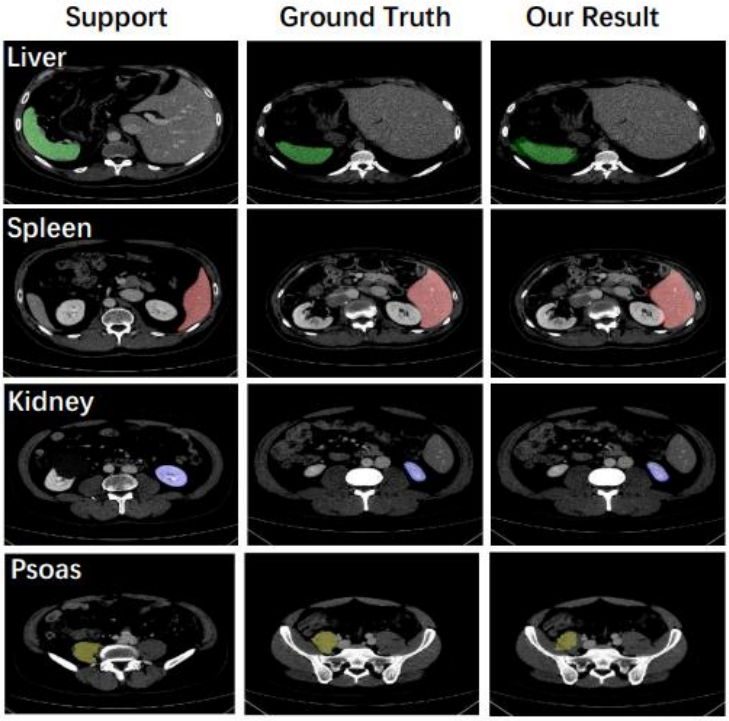


Figure 4.5 Visual results of medical images

Visual results of proposed segmentation method based on FSL. Figure taken from [21]

Aihua Cai, Wenxin Hu, and Jun Zheng have explored the use of few-shot learning techniques for medical image classification tasks, specifically for identifying liver cancer in CT images [22]. The difficulties of classifying medical images are covered in the first section of the paper, along with the high dimensionality and limited supply of labeled data. The authors point out that standard convolutional networks have difficulty extracting useful information from medical images since their characteristics are frequently chaotic and varied. The neural network made up of four convolutional blocks that serves as the embedding function f(x) in the proposed method is a neural network. The authors include an attention mechanism with spatial and channel attention modules between the third and fourth convolutional blocks to increase the

model's capacity for representation. These modules help extract the image's spatial and channel information in turn.

Additionally, the authors add 1x1 convolution kernels to the convolutional blocks to deepen the network without sacrificing resolution and enhance non-linear properties. The suggested method is assessed using two medical imaging datasets and contrasted with other cutting-edge techniques. The experimental results demonstrate the efficiency of the attention mechanism in enhancing the Prototypical Network for classifying medical images, since the proposed method accomplish more classification accuracy than the baseline methods. The study offers a potential method for classifying medical images that makes use of both Prototypical Networks and attention mechanisms. The authors show that their method outperforms other cutting-edge techniques in terms of classification accuracy by successfully handling the noisy and diverse aspects of medical images. The model's architecture is depicted in the figure below.

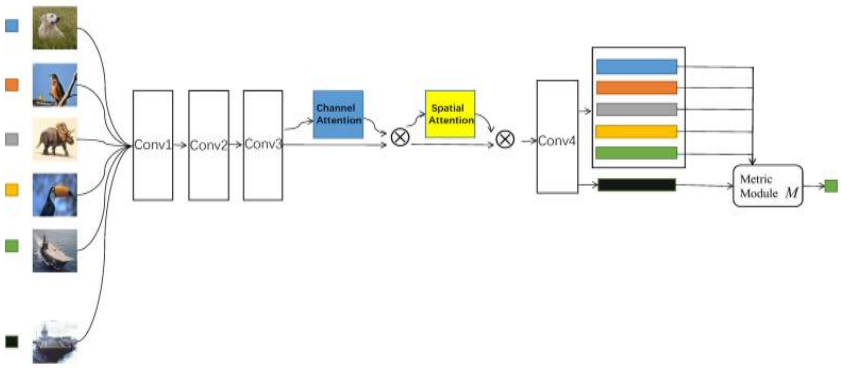


Figure 4.6 Proposed architecture for few shot learning by Ailua and team. Figure taken from [22]

Finally, they evaluated the accuracy of their model to pre-trained models with changed classification layers. The ImageNet dataset, which contains real images, is generally used to train the parameters of the pre-trained models. However, when used for medical imaging applications, these models perform less well than they should since medical images differ greatly from natural photos on ImageNet.

Method	Accuracy
AlexNet	79.80%
VGG16	82.08%
ResNet101	82.57%
DenseNet169	83.06%
Dual Path Network	82.74%
New Method	92.44%

Table 4.1 Accuracies based feature extraction method. [22]

4.2.1.2 Siamese Networks

A Siamese Neural Network belongs to a category of neural network architectures that incorporate two or more identical subnetworks. These subnetworks are characterized by the same configurations, parameters, and weights. Initially introduced by Bromley in 1993, this network architecture was primarily devised for signature verification purposes. [23]. The Siamese network is a useful tool for enhancing data in few-shot learning. The Siamese network can create additional training instances by comparing pairs of training data, increasing FSL's accuracy and resilience. When compared to merely training the model on the original examples, this method is especially advantageous since it gives the model more varied data.

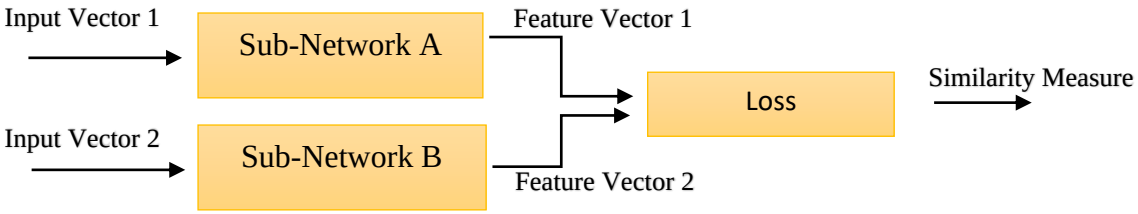


Figure 4.7 Conceptual diagram of Siamese Neural Network

Due to inadequate amounts of training data, a model can frequently become overfitted within a few learning attempts. By adding training data in the form of creating similar pairs and dissimilar pairs, the Siamese network solves this issue. The Siamese network can be viewed as a representation learning process that seeks to identify low dimensional discriminative characteristics. By doing this, it can produce effective representations and remove extraneous or redundant data from the training set. To put it another way, the Siamese network aids in selecting the most pertinent and significant traits while ignoring the unnecessary ones, resulting in more effective and efficient training [24].

Gregory, Richard and Ruslan from the University of Toronto have introduced a novel approach for one-shot image recognition using Siamese neural networks. The authors highlight that for these methods to work, they often depend on manually created features and a substantial amount of labeled data. They contend that a more adaptable strategy is required, one that can learn new information from a small sample size and automatically identify relevant aspects in the data. The authors suggest a Siamese neural network architecture to overcome these problems. Siamese networks are made up of two identical neural networks, each of which generates a low-dimensional embedding from an input image. To assess if the two input images are similar or different, the two embeddings are then compared using a distance measure [25].

They have demonstrated the effectiveness of their approach on a number of one-shot recognition tasks, including handwritten character recognition and face verification. They show that their Siamese network can achieve state-of-the-art results with just one or a few examples per class.

5 METHODOLOGY

This section covers the implementation details of the research.

5.1 System Architecture

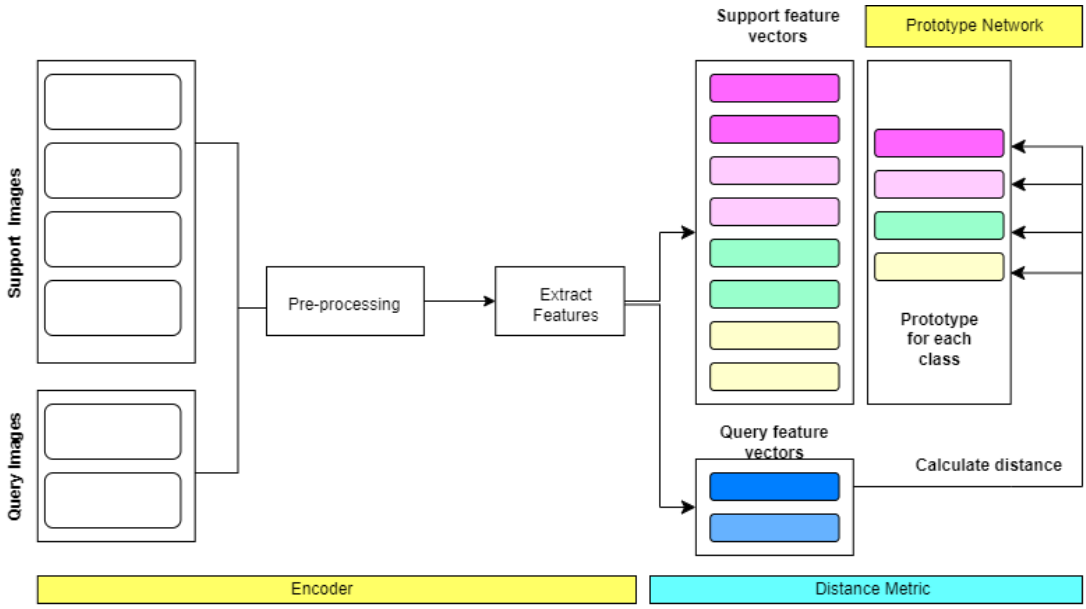


Figure 5.1 System Architecture of the proposed model

A prototypical network architecture is composed of three primary components: an encoder network, a prototype network, and a distance metric. During the encoder phase, input data is encoded into a feature space. The prototype network then calculates class prototypes by averaging the embeddings of the support examples that belong to the same class. These support examples consist of a small set of labeled examples that are provided during training, and the prototypes function as a condensed representation of each class. Finally, the distance metric measures the distance between the query example and the class prototypes within the feature space.

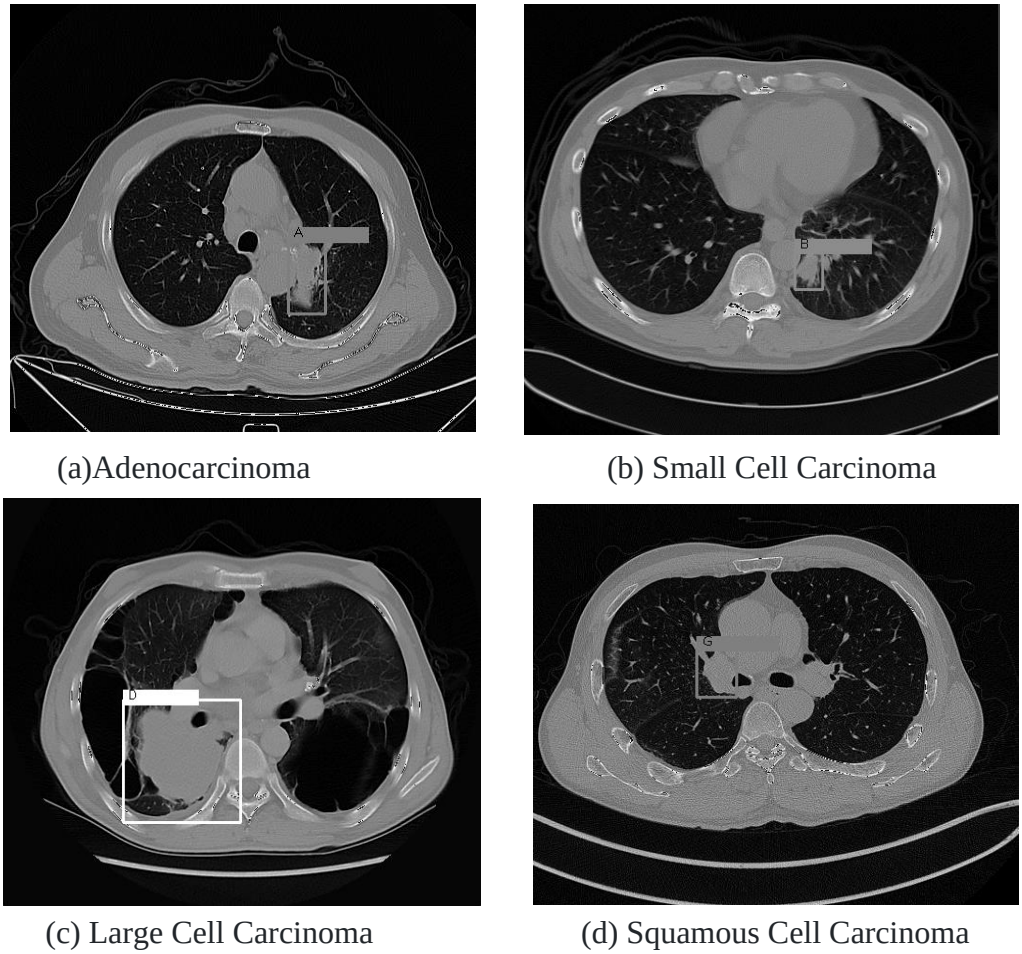
Figure 5.2 Flow of preprocessing

One essential component of this system is image preprocessing. The pre-processing procedures are depicted in the diagram above. Future chapters will discuss each component.

5.2 Data set

All CT scan images used for implementation were obtained from the TCIA. TCIA stands for The Cancer Imaging Archive. This open-access platform is built by the National Cancer Institute in order to support research developments and educational projects that are based on medical imaging of cancer. [26]

For this study's objective, a dataset from TCIA that comprises of CT and PET-CT DICOM images of lung cancer patients with XML Annotation files indicating tumor location with bounding boxes is employed. The images were obtained in hindsight from individuals who were suspected to have lung cancer and had received both a standard-of-care lung biopsy and PET/CT. The CT scans of patients with adenocarcinoma, small cell carcinoma, large cell carcinoma, and squamous cell carcinoma are included in this collection. The mediastinum (window width: 350 HU; level: 40 HU) and lung (window width: 1,400 HU; level: -700 HU) settings were used for the examination of the images. The reconstructions, which concentrated on lung settings, were made with a 2mm slice thickness. The spacing between CT slices varied from 0.625mm to 5mm. Plain, contrast, and 3D recons were the scanning modes that were employed.



(a)Adenocarcinoma (b) Small Cell Carcinoma

(c) Large Cell Carcinoma (d) Squamous Cell Carcinoma

Figure 5.3 CT scan images each lung cancer type.
Figures taken from [26]

5.3 Image Representation

Images can be expressed as a grid of numbers, where each number corresponds to a pixel's intensity. The basic building blocks of a digital image are pixels, which reflect the brightness of the actual scene. To obtain the pixel values from CT scan images, we use the `` `pydicom.dcmread()` `` function, which provides a `` `DicomObject` `` as output. This object comprises two main components: metadata information such as the patient's name, study date, and modality, among others; and the actual image data, which is stored as a 2D array in the `pixel_array` attribute of the `` `DicomObject` ``. The element of the array that corresponds to each pixel in the picture represents the intensity value of each pixel at that location in the image.

5.4 Image Preprocessing

Few shot learning is defined as N-way-K-shot-classification. There are two main datasets in FSL, which are called support sets and query sets. The support set should have images with categories or classes. The number of categories is referred to as N classes. The number of images belongs to each class considered "K-shots. Assume we have a dataset that has 4 categories, and each category has 2 images. We can call it the 4-way-2-shot classification. Query images are used to find the classes they really belong to.

To handles the DICOM format images, custom data set class which inherits properties from torch. utils.data.Dataset is created. This custom dataset class can be loaded in to a model for training or testing and it has the __len__() method which returns the length of the input data and the __getitem__() method takes an index as input, retrieves the corresponding sample and target, and applies any transformations if specified before returning the sample and target as a tuple.The customized dataset class will be used in a PyTorch Data Loader for further model implementation.

Once __getitem__() method reads the DIOCM file located ,it extract the pixel array data from the file and then it will converted to a Numpy array for next preprocessing.

5.4.1 Transform to Hounsfield Units

In this phase the pixel representation of the DICOM image will converts to Hounsfield Units (HU). Converting pixel values to HU, images can be more easily compared and analyzed, allowing for more accurate and reliable diagnosis and treatment planning.

RescaleIntercept: As the first step the rescale intercept value from the DICOM header of the medical image. The rescale intercept is a DICOM attribute that is used to adjust the pixel values to account for any offset in the measurement.

RescaleSlope: This line retrieves the rescale slope value from the DICOM header of the medical image. The rescale slope is a DICOM attribute that is used to adjust the pixel values to account for any scaling factor in the measurement.

Next rescale slope and intercept values to the pixel array of the medical image to convert it to Hounsfield Units. Hounsfield Units are a standardized scale used in medical imaging that represent the radiodensity of a particular tissue or material.

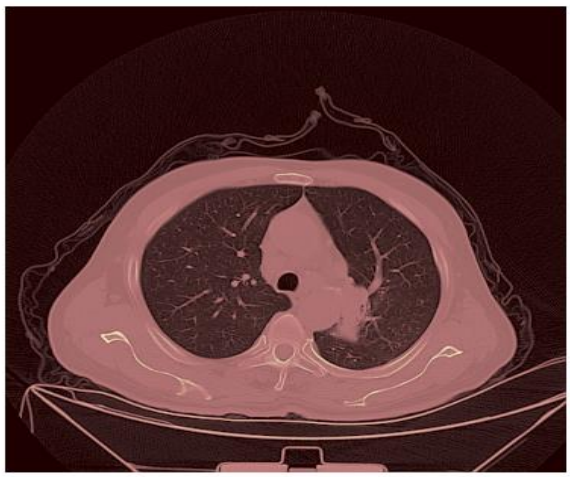


Figure 5.4 A CT scan after applied HU transformation.
Figure taken from [26] for pre-processing

5.4.2 Image enhancement

The main goal of image enhancement is to make the image's quality, on a subjective level, higher than the original image. Image enhancement aims to improve the clarity of image features while preserving the natural image structure. This is done to facilitate image analysis tasks such as edge and boundary detection. Noises are already embedded in the CT scan images and the noise can be cause to false detection of stage of the cancer. Therefore, noises have to be removed to get accurate results. In this implementation median filter is applied.

Median filtering is a nonlinear process which is used for removing impulsive or salt and pepper noise from the digital images. When utilizing this particular filter, each individual pixel within the image is assessed in sequence, with a focus on evaluating its immediate neighbors to determine whether or not it is truly indicative of its surrounding area. Unlike other filters that may simply replace a pixel value with the

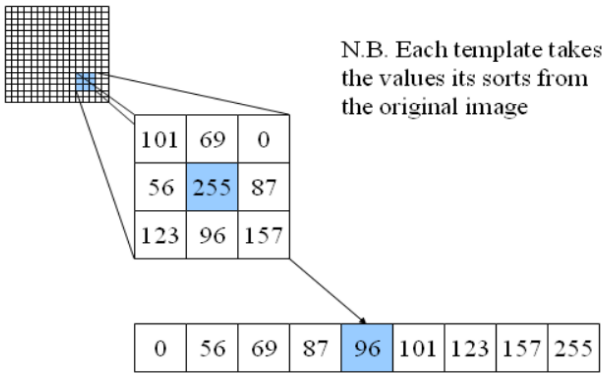
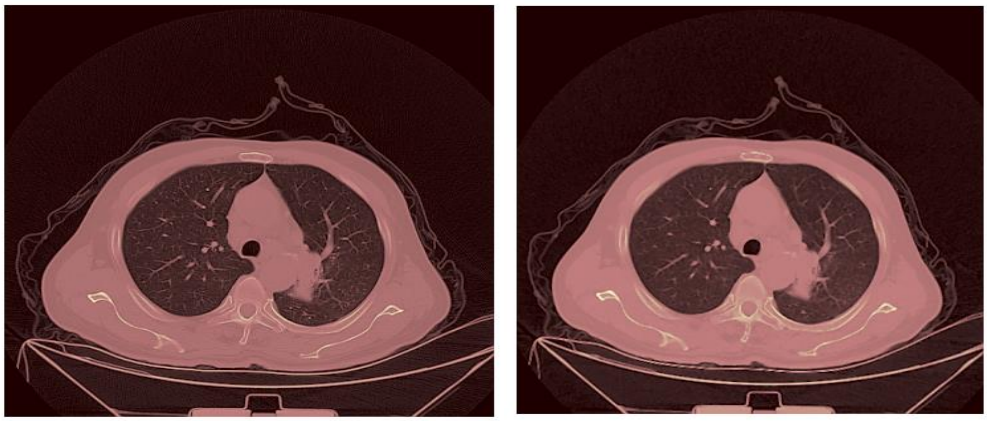


Figure 5.5 Median filtering.

Figure is taken from computer vision demonstration website in University of Southampton

mean average of its neighboring values, this filter replaces it with the median average of nearest values.

The median is computed by first arranging all of the neighborhood's pixel values in numerical order, and then substituting the middle pixel value for the one being evaluated.



(a) Before noise removal (b) After noise removal

Figure 5.6 Enhanced CT scan Images. CT scan image taken from [26]

5.5 Image Segmentation

In medical image analysis, automated segmentation of anatomical structures plays a pivotal role. The goal of image segmentation is to divide an image into distinct regions or classes, where each subset is internally homogeneous in terms of one or more characteristics. As such, numerous segmentation techniques have been proposed by researchers for CT scans. Johannes, Florian, Jeanny, and Sebastian conducted a comparative analysis of deep learning approaches using various lung segmentation algorithms. Four generic semantic segmentation models, including U-net, ResU-net, Dilated Residual Network, and Deeplab v3+, were considered in their study. Based on the results, it was observed that the U-net model provided coverage over a wide range of visual variability [27].

5.5.1 U-Net Architecture

U-Net Architecture is a CNN architecture that was proposed for biomedical image segmentation tasks in 2015 by Olaf Ronneberger, Philipp Fischer, and Thomas Brox. The architecture is named after its U-shaped design, which consists of a contracting path (down sampling) followed by an expansive path (up sampling) [28].

The architecture is made up of three primary components: the bottleneck layer, the expanding path, and the contracting path. The input image's spatial resolution is steadily decreased as the number of feature mappings is increased by the several convolutional and pooling layers that make up the contracting route. A single convolutional layer that joins the expanding and contracting channels creates the bottleneck layer. The expanding path, which is the mirror image of the contracting path, has numerous deconvolutional layers that gradually boost the feature maps' spatial resolution while lowering their channel count. To maintain spatial information, matching layers in the contracting and expanding routes are connected via skip connections.

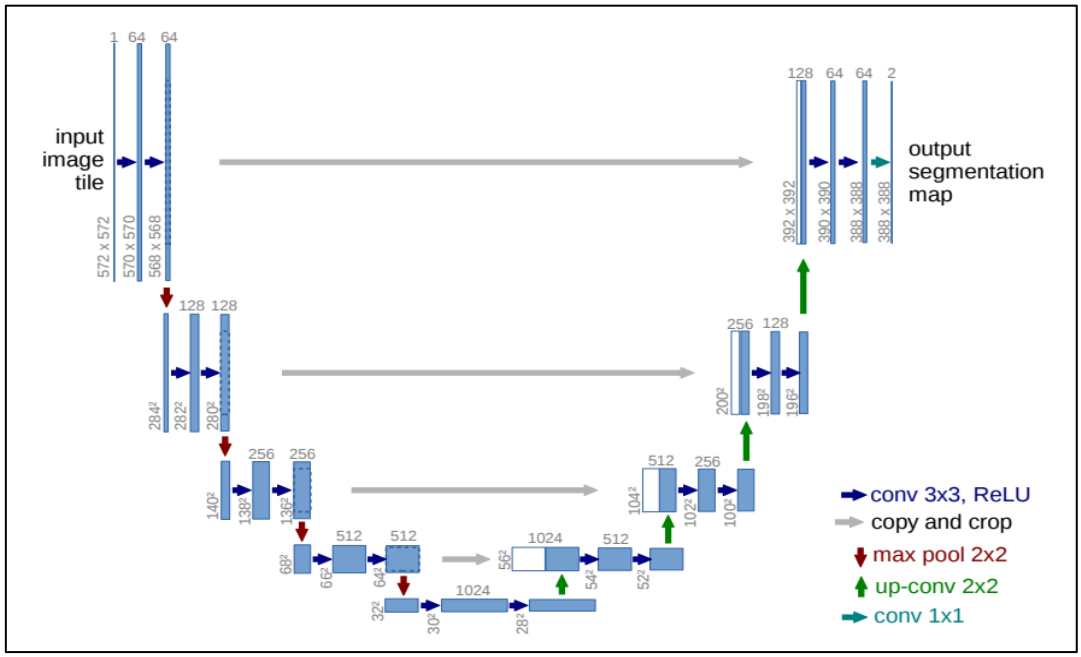
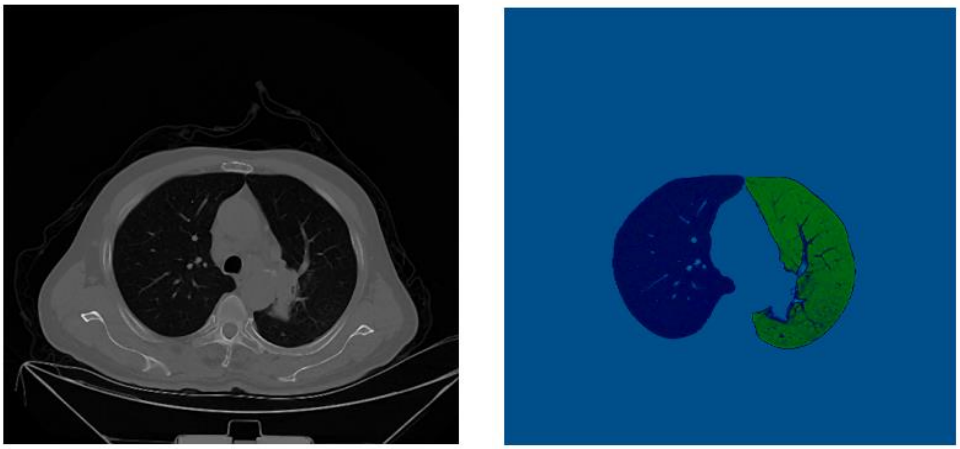


Figure 5.7 U-net architecture (example for 32x32 pixels in the lowest resolution).
Figure is taken from [28]

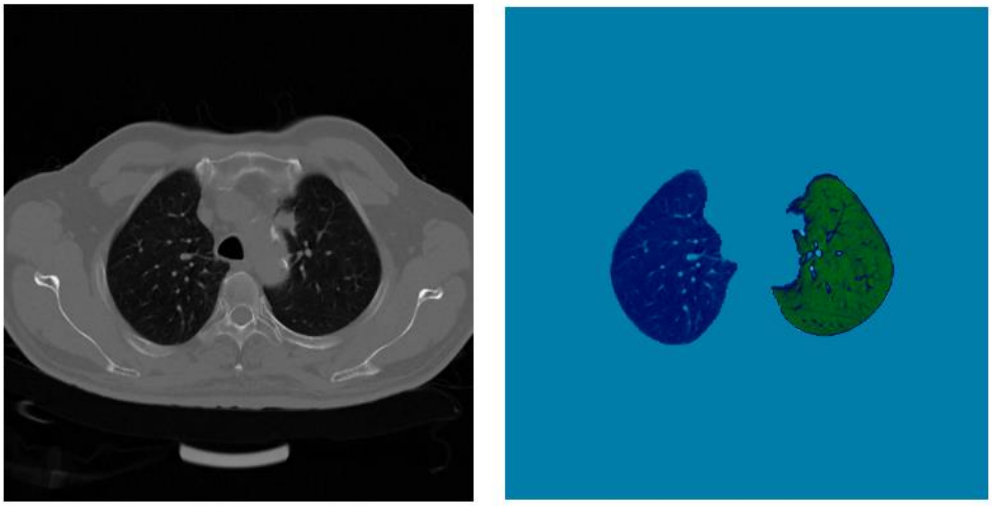
R231 refers to a specific variant of the U-Net architecture, which was proposed in a research paper titled "The Importance of Skip Connections in Biomedical Image Segmentation" by Hui Ding, Xiaofei Zhang, and Ming Ma in 2021 [29].The R231 variant of U-Net extends the original U-Net architecture by adding more layers to each of the contracting and expansive paths. Specifically, the R231 architecture has 23 convolutional layers in the contracting path and 23 deconvolutional layers in the expansive path. This model separates the right and left lungs into distinct slices and removes air pockets, tumors, and effusions.

In this implementation the same U-net model was used to segment the lung from pre-processed CT images. To apply this model programmatically, Python package called “lungmask” is imported ,which provide U-Net(R231) variant from the architecture.



(a) Before segmentation *(b) After segmentation*

Figure 5.8 U-Net (R321) applied for CT scan image with Adenocarcinoma. CT scan images taken from [26]



(a) Before segmentation *(b) After segmentation*

Figure 5.9 U-Net (R321) applied for ct scan image with Squamous Cell Carcinoma. CT scan images taken from [26]

5.6 Feature Extraction

In this phase, meaningful features from input image data are extracted. In order to achieve this step pre-trained models were used. The input images are passed through the feature extractor and output feature maps are flattened to form a feature vector for each image. In this experiment ResNet50, DenseNet, VGG16 and CNN were used for the better analysis.

5.6.1 VGG16

VGG16 is a one pre-trained model that used for feature extraction of lung CT images. VGG16 is a convolutional neural network architecture designed for image classification tasks, first introduced in the 2014 paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" by Simonyan and Zisserman [30]. VGG16, an architecture developed by the Visual Geometry Group (VGG) at the University of Oxford, features a composition of 16 convolutional layers followed by 3 fully connected layers. The convolutional layers possess a consistent filter size of 3x3, a stride of 1, and a padding of 1, ensuring the preservation of the input image's spatial resolution. As we progress deeper into the network, the number of filters in each layer progressively rises, starting from 64 in the initial layer and culminating at 512 in the final layer.

After each convolutional layer, a rectified linear unit (ReLU) activation function is applied to introduce non-linearity into the model. Max pooling is used after every two convolutional layers to reduce the spatial resolution of the feature maps while retaining the most important information. Finally, the output of the convolutional layers is flattened and passed through 3 fully connected layers, with the final layer producing the class scores for the input image.

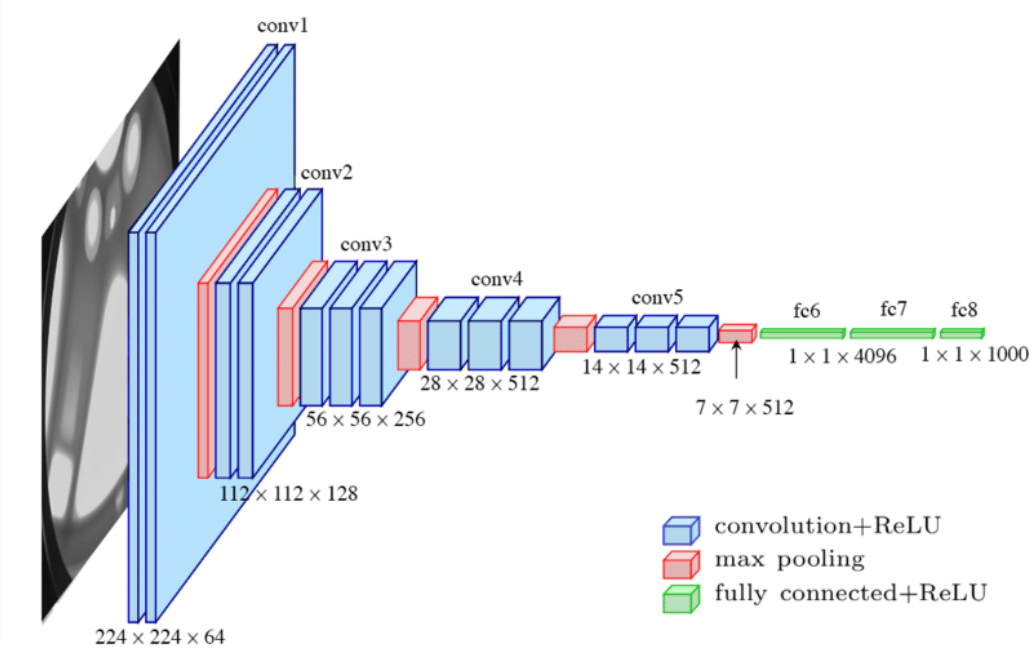


Figure 5.10 The standard -16 architecture

Adapted from "Simonyan and Zisserman (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition VGG *1-14".*

For this implementation, the model was changed to accept grayscale images as input rather than the RGB images that the original VGG16 architecture was designed for. The pretrained parameter is used to load the weights of the pre-trained VGG16 model, which are used as initial values for the model's parameters. The last fully connected layer of the VGG16 model is removed by creating a new Sequential module containing all the layers of the VGG16 features module except for the last layer. But the model will keep the output channel size as 64 and the kernel size, stride and padding values of the convolution layer remain the same as in the original VGG16 model.

5.6.2 ResNet50

ResNet50 is a convolutional neural network architecture that was introduced in the paper "Deep Residual Learning for Image Recognition" by He et al. in 2016 [31]. It is a variant of the ResNet family of models, which are known for their ability to train very deep neural networks with thousands of layers. The ResNet50 architecture comprises a total of 50 layers, encompassing convolutional layers, pooling layers, fully connected layers, and shortcut connections that enable the bypassing of one or more layers. By incorporating these shortcut connections, the network effectively addresses the issue of vanishing gradients that can arise during the training of deep neural networks.

There are four phases in the ResNet50 design, and each one includes several residual blocks. Two or three convolutional layers with batch normalization and ReLU activation are included in each residual block, along with shortcut connections that combine the input and output of the block. The first stage has 64 filters, the second stage has 128 filters, the third stage has 256 filters, and the fourth stage has 512 filters. The output of the last residual block is fed into a global average pooling layer, which averages the feature maps over the spatial dimensions of the input. The resulting feature vector is then passed through a fully connected layer with SoftMax activation to produce the final class probabilities. This can be used as a feature extractor for a wide range of computer vision tasks. When used for feature extraction, the pre-trained ResNet50 model is used as a fixed feature extractor, and the output of one of its intermediate layers is used as the input to a classifier or another downstream task-specific model. In order to apply ResNet50 as a feature extractor for this experiment, the model is modified to accept single channel grayscale images as input.

5.6.3 Convolution Neural Network

The CNN architecture is widely used for feature extraction in image-based machine learning applications, including few-shot learning. The CNN model consists of several convolutional layers followed by pooling layers and fully connected layers. In the context of few-shot learning, the CNN model is trained on a large dataset to learn the

high-level feature representations of the images. Then, the trained CNN model is used to extract the features from the support and query images in the few-shot learning scenario.

In this methodology, the CNN model used has three convolutional layers, each followed by a ReLU activation function and a pooling layer. The convolutional layers perform the operation of convolving a set of learned filters with the input image to extract features. The pooling layers perform a down sampling operation by selecting the maximum value within a given region of the feature map. The fully connected layers at the end of the network take the flattened feature maps and perform classification using the extracted features.

The CNN model can be fine-tuned on a limited number of labeled support pictures for each new classification assignment in order to adapt it for few-shot learning. The CNN model may learn to extract characteristics that are particular to the new classification job by fine-tuning on the support set. The distance between the features of the query picture and the class prototypes generated from the support images may then be utilized to categorize the query images using the extracted features.

5.6.4 DenseNet

DenseNet is a deep learning architecture for image classification and object detection tasks. DenseNet is unique in that it incorporates the outputs of all preceding layers as input to each subsequent layer [32]. This dense connection structure enables feature reuse and lowers the number of parameters needed to learn many features by giving each layer direct access to the feature maps of all previous layers.

In terms of feature extraction, DenseNet has been shown to be very effective in learning features from images. By utilizing dense connectivity, the network can learn a diverse range of features across different scales and levels of abstraction. This is particularly important for object detection tasks, where objects can appear at different scales and orientations. DenseNet can be used as a feature extractor in the prototypical network architecture. The pre-trained DenseNet model can be used to extract features from the images in the support and query sets. The output of the DenseNet model is typically a high-dimensional feature vector that represents the image.

Layers	Output Size	DenseNet-121($k = 32$)	DenseNet-169($k = 32$)	DenseNet-201($k = 32$)	DenseNet-161($k = 48$)
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	[1 × 1 conv; 3 × 3 conv] × 24	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 48	[1 × 1 conv; 3 × 3 conv] × 36
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	[1 × 1 conv; 3 × 3 conv] × 16	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 24
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

Figure 5.11 DenseNet architectures for ImageNet. Figure taken from [32].

5.7 Prototype Representation

In a few shot classification, small support set of N labeled exampled will be given. D-dimensional feature vector of each example will be output by VGG16 model. In order to create a prototype for each class ,average of all feature vectors in a class label will be calculated. This prototype serves as a representative for the label and is used to compare with the features of new examples during inference.

If we consider a small support set of N labeled examples S= $\{(X_1,y_1)\ldots\ldots(X_N,y_N)\}$ where each $X_i \in \mathbb{R}^{\mathcal{D}}$ is the D-dimensional feature vector of an example and $y_i \in \{1, \ldots\ldots, K\}$ is the corresponding label. S_k denotes the set of examples labeled with class k.

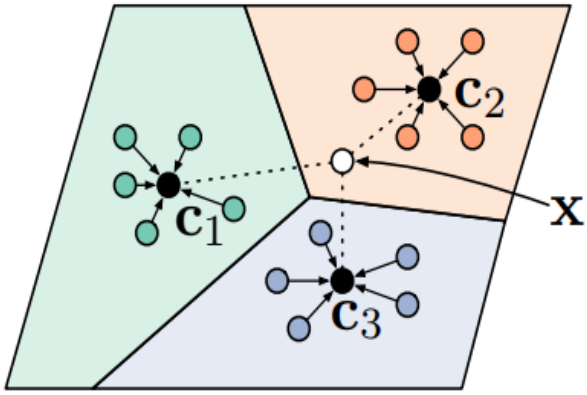


Figure 5.12 Few shot prototypes as the mean of embedded support features.
Figure taken from [33]

5.8 Calculate Distance Metric

One key component in prototypical network is calculation of distance matrix. The distance matrix is used to measure the similarity between support se and the query set. It's shown that Euclidean distance is perform better than cosine similarity when calculating distance [33].Below shows the formula for Euclidean distance.

$$d(z, z') = \| z - z' \|^2$$

when Euclidean distance is applied, the model is equivalent to linear model with a particular parameterization.

5.9 Meta Learning

Meta-learning is a popular machine learning technique often applied to Few-Shot Learning (FSL). The objective of meta-learning algorithms is to rapidly acquire knowledge about a novel task using a limited amount of new data. Meta-learning, also known as "learning to learn". In particular, it selects training samples from the base classes for few-shot classification tasks and optimizes the model to perform well on these tasks. An N-way and K-shot task typically consists of N classes, each of which has K support samples and Q query samples.

The objective is to group these N x Q query samples into N classes using the N x K support samples. In this approach, few shot classification tasks are directly tuned for the model. The primary benefit of meta-learning is thought to be the alignment of training and testing objectives. In this approach, few shot classification tasks are directly tuned for the model. The primary benefit of meta-learning is thought to be the alignment of training and testing objectives. Usually, the Dataset D which take for

standard supervised machine learning, split in to two separable and unique datasets, which are called train and test sets denoted by D_{train} and D_{test} respectively. One iterative cycle, during the conventional neural network, is called “Epoch”. During each epoch, the whole D_{train} is feeding forward and backward and the it will be partitioned in to several parts which is called “Batch”. Consequently, the number of iterations per epoch are the all batches that feed in to model.

In the meta learning process, the dataset D split in to two meta sets without any overlapping and it can be named as D_{meta_train} and D_{meta_test}. D_{meta_train} and D_{meta_test} are then further separated into train and test sets. To distinguish these two types of training and testing sets, the inner train and test sets are referred to as "Support set" and "Query set," respectively. The support set is in the form of N-way k-shot random samples, and the query set consist of q random samples for each of the N support set classes and these two data sets would be created during each episode. Support set in the D_{meta_train} will be taken by model to learn and then applied on the query set in the in the $D_{meta\_train.}$ Then the prediction will be made. The prediction error over episodes is used to update the meta-learner. The meta-learner learns to learn from the limited dataset throughout a series of episodes. This stage is known as meta-learning. After that support set in the D_{meta_test} is used to create a classifier. In this case, the model focus on fast learning task-specific parameters. Subsequently, the performance of the classifier is assessed using query set in the D_{meta_test} to enter the adaption phase. In other words, the model utilizes the prior knowledge acquired during the meta-learning phase to rapidly adjust to a new task during the adaption phase.

For this development, the CrossEntropyLoss() function is used to define the loss criterion for the classification task. This function computes the cross-entropy loss between the predicted class scores and the actual labels for each training example. The optim.Adam() function is used to define the optimizer for updating the model's parameters during training. Adam is a popular optimization algorithm that computes adaptive learning rates for each parameter, allowing the model to converge faster and more reliably.

The learning rate wat set to 0.001, which determines the step size taken by the optimizer in the parameter space. Low learning rates can lead the model to converge slowly, while excessive learning rates can cause the model to diverge and exceed the ideal parameters. The optimal learning rate depends on the specific problem and dataset being used.

5.10 Tools & Technologies

Python was chosen as the implementation language due to its reputation as a high-level, interpreted programming language that excels in simplicity, readability, and user-friendliness, particularly in the domains of data science and web development. The following list enumerates the libraries and frameworks employed in the project:

- PyTorch
- Numpy
- PyDicom
- OpenCV
- Tensorflow
- Matplot
- Lungmask

As the IDE ,Jupyter Notebook Extension in Visual Studio Code is used. The Jupyter Notebook extension allows users to run Jupyter notebooks within VS Code, providing a more integrated and streamlined development experience. Jupyter notebooks are interactive documents that combine code, text, and visualizations. With the Jupyter Notebook extension, users can create, edit, and run Jupyter notebooks directly within VS Code. The extension provides syntax highlighting, code completion, and other features that make it easier to work with Jupiter notebooks. It also includes a built-in Jupyter server that can run notebooks in the background and provide access to interactive widgets.

6 EVALUATION

The experiment encompassed the utilization of four different feature extractors. To assess the efficacy of each extractor, the accuracy of each one was computed. This computation involved adjusting the number of images in each class within the support

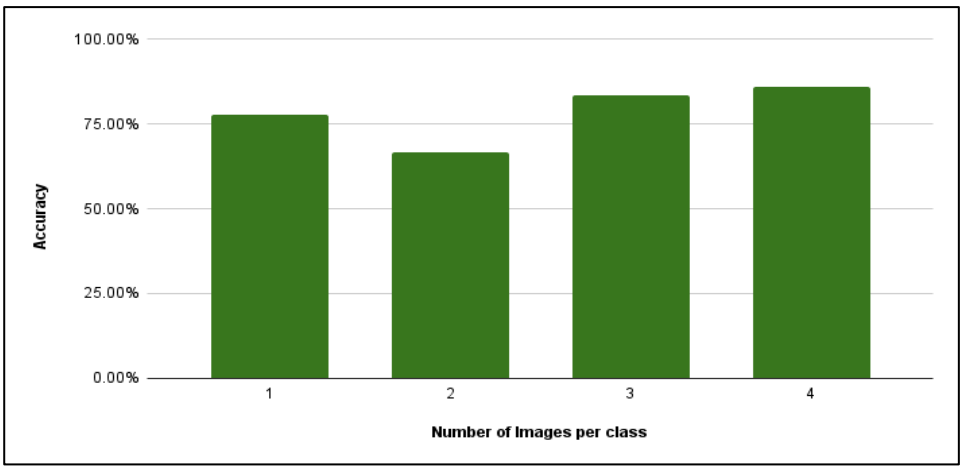


Figure 6.1 Accuracy for VGG16 model

set to gauge performance.

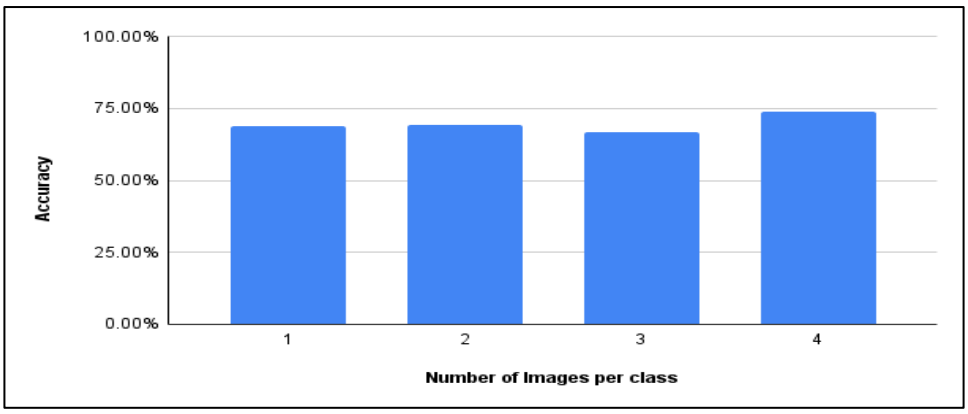


Figure 6.2 Accuracy for CNN

According to figures 6.1 and 6.2, the highest levels of accuracy were achieved when each class had 3 or 4 shots. Conversely, when a class had only one or two shots, the accuracy for VGG16 and CNN models was significantly lower. The following two graphs depict the accuracies achieved by using ResNet50 and DenseNet as feature extractors for the prototypical network. It can be observed that all accuracies are lower than those achieved by the VGG16 model. The highest accuracy was achieved with the DenseNet model, as shown in Figure 6.4, which was 76%. However, when each class had 4 shots, the VGG16 model achieved a higher accuracy of 86%.

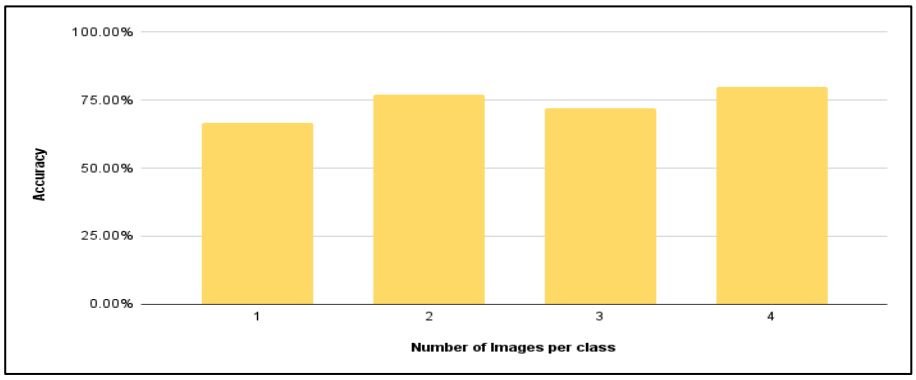


Figure 6.3 Accuracy for DenseNet

Figure 6.4 Accuracy for ResNet50

For further analysis inter class and intra class variation in the feature space were calculated. A higher ratio indicates that the classes are well-separated and easily distinguishable in the feature space, whereas a lower ratio indicates that the classes are more difficult to distinguish. This measure can be useful in evaluating the effectiveness of the prototypical network in learning a high-quality feature space that can accurately distinguish between different classes. The following four figures depict the feature spaces of the support dataset using four different feature extraction models. Each color represents a unique class.

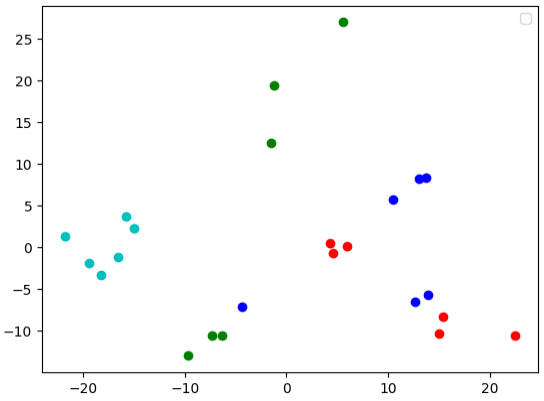


Figure 6.6 Feature space extracted using CNN

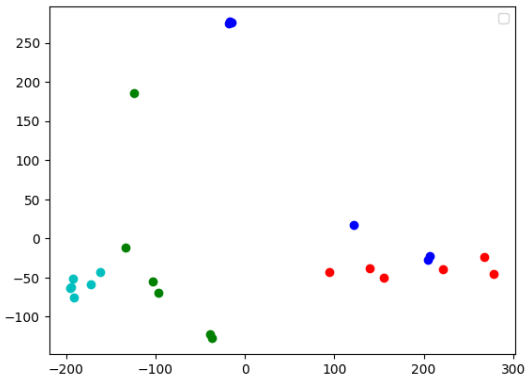


Figure 6.5 Feature space extracted using DenseNet

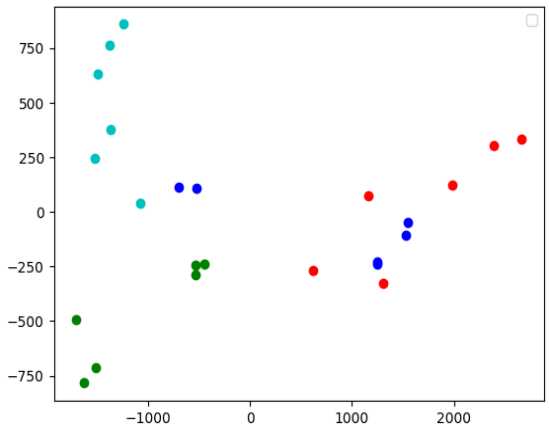


Figure 6.8 Feature space extracted using ResNet50

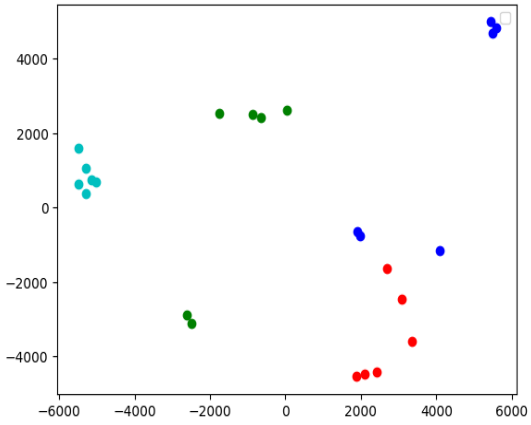


Figure 6.7 Feature space extracted using VGG16

Upon analyzing the ratio of intra-class distance to inter-class distance for each feature extraction model, it was observed that the VGG16 model had the highest ratio. This indicates that the classes were well separated in the feature space generated by the VGG16 model. In other words, the VGG16 model was able to create a feature space where the intra-class distances were relatively smaller compared to the inter-class distances. This is a positive indicator of the quality of the feature space and suggests

that the VGG16 model may be better suited for the task of few-shot learning when compared to the other feature extraction models.

Feature Extractor Model	Ratio of inter-class to intra-class distances
CNN	16
VGG16	295358
ResNet50	4659
DenseNet	215255

Table 6.1 Ration of inter class to intra-class based on the feature extraction model.

Based on the analysis conducted, it was determined that VGG16 performed the best as a feature extractor when compared to other models. In other words, the VGG16 architecture was found to be the most effective at extracting useful features from lung CT scan images for this implementation, outperforming other architectures that were tested.

Next step was to do episodic learning, also known as meta-learning, is a machine learning technique that involves training a model to learn how to learn. In the context of few-shot learning, meta-learning involves training a model on many "episodes," each of which consists of a few examples of a new task. During each episode, the model is represented with 4 shots per class and model is trained to minimize the distance between the feature vectors of each example and its class prototype. With only 3 episodes, the model achieved an accuracy of 80%. To further improvement of the accuracy the model is trained for 50 training episodes. The depicted figure illustrates the relationship between the number of episodes and the validation accuracy of the model. In other words, the graph demonstrates how the performance of the model, as measured by accuracy on a validation dataset, changes as the number of training episodes increases.

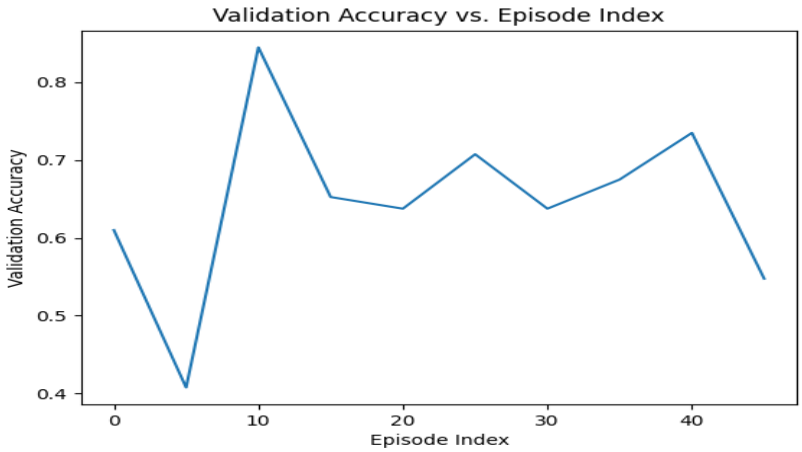


Figure 6.9 Validation accuracy based on episodes.

The figure above shows that the highest level of accuracy was achieved when the number of episodes was set to 10, which is 86%.

Siamese neural network is a another few shot learning method which belongs to meta learning approach. These methods are designed to learn similarity or distance metrics between pairs of inputs. They consist of two identical subnetworks with the same weights and parameters. For this observation, a convolutional neural network with three convolutional layers followed by fully connected layers is used.

After passing the two preprocessed DICOM images for the network, the loss is calculated. Contrastive loss is used as the loss function for learning similarity or dissimilarity between pairs of samples. For this implementation a training loop with 5 episodes was used. Through the training, a pair of images will be passed to the neural network and two out vectors from the network will be fed in to the define loss function. In the loss function, the loss of the output images will be calculated and then loss will pass to the back propagation. Below graph shows the loss over time.

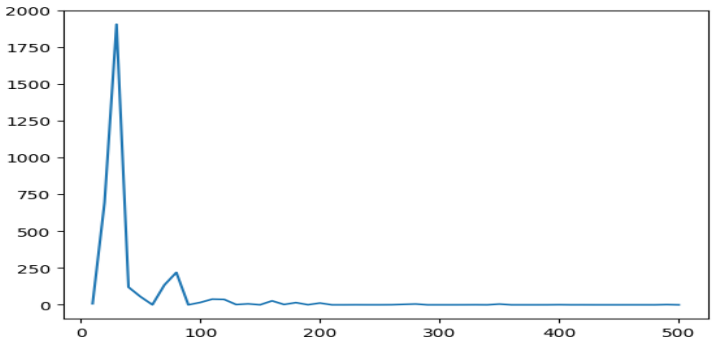


Figure 6.11 Loss over time in contrastive loss learning

As indicated in the graph, the loss started around 1800 and ended at close 0. After that the trained model was tested using test dataset. Two images from test set is passed in to the and obtain two vectors and then Euclidean distance between two vectors has calculated using F.pairwise_distance() function in functional library in PyTorch module. Below figures shows that the pairs from same lung cancer type have small dissimilarity number which is close to 0. The dissimilarity number is large when the DICOM images from different lung cancer types.

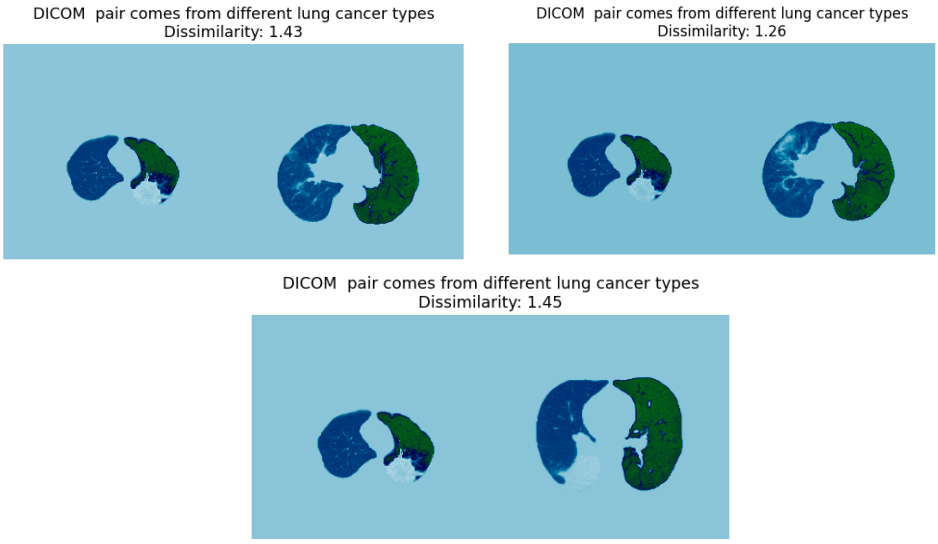


Figure 6.10 Lung image pairs with different lung cancer types.

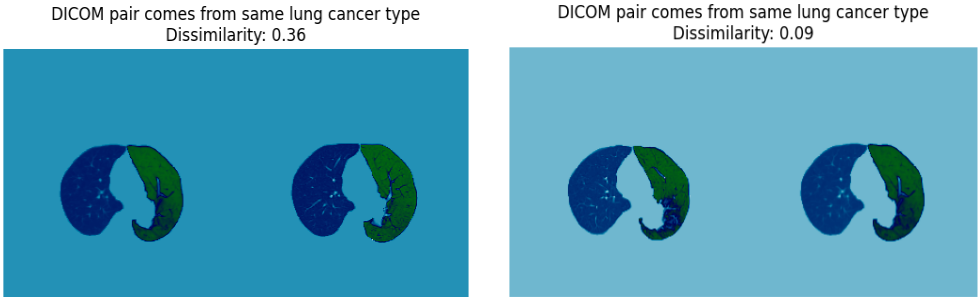


Figure 6.12 Lung image pairs with same lung cancer type

Siamese neural networks and prototypical networks are two different approaches used in few-shot learning tasks. Below table illustrates the comparison between two network types.

	Prototypical Network	Siamese Neural Network
Architecture	Consists of an encoder network, prototypical network and prototype class	Consist of two identical subnetworks with same weights.
Architecture of the network used	VGG16. Multiple convolutional layers followed by ReLu activation function.	Three convolutional layers followed by ReLu activation and max pooling. Three fully connected layers with ReLU activations.
Training Procedure	Minimize the distance between the support images and the respective class prototype, while maximizing the distance from prototypes of other classes. Meta Learning	Minimize the distance between embeddings of similar instances and maximize the distance between embeddings of dissimilar instances. Contrastive Loss Learning

Table 6.2 Comparison of Prototypical Network and Siamese Neural Network

7 CONCLUSION

The central objective of the thesis project revolved around the identification of lung cancer types using CT scan images, with a primary focus on ascertaining the most appropriate image processing methodology for medical images in situations where the availability of training data is limited. In the study, PET-CT DICOM images from TCIA platform is used. The project utilized the prototypical network, a well-known few-shot learning model, which creates a prototype of all feature vectors from images. To achieve a better feature space, four different feature extraction methods were explored, and it was found that the VGG16 model provided the best feature performance, achieving an average accuracy of 86% in the 4-shot 4-way setting. In addition, the project involved segmentation of lung images, and the use of image enhancement techniques proved to be beneficial in achieving improved results.

To further improve the accuracy of the model, Meta Learning was applied. The model was trained for 60 episodes and the best accuracy is recorded as 88% through the training. The project highlights the effectiveness of using the prototypical network in few-shot learning, as well as the importance of selecting appropriate feature extraction methods to achieve accurate results in medical image processing, especially when data is limited. The application of Meta Learning also emphasizes the potential for further improvement in accuracy using advanced machine learning techniques.

8 REFERENCES

[1] R. Siegel, K. Miller and A. Jemal, "Cancer statistics," *CA: a cancer journal for clinicians,* vol. 69 (1), pp. 7-34, 2019.

[2] "American Cancer Society," [Online]. Available: https://www.cancer.org/cancer/lung-cancer.

[3] N. Wu, J. Phang, . J. Par and Y. Shen, "Deep Neural Networks Improve Radiologists' Performance in Breast Cancer Screening," *IEEE Transactions on Medical Imaging,* vol. 39, pp. 1184--1194, April 2020.

[4] Y. Tang, Y. Tang, J. Xiao and R. M. Summers, "XLSor: A Robust and Accurate Lung Segmentor on Chest X-Rays Using Criss-Cross Attention and Customized Radiorealistic Abnormalities Generation," in *MIDL*, 2019.

[5] P. Chathurvedi, A. Jhamp, M. Vanani and V. Nemade, "Prediction and Classification of Lung Cancer Using Machine Learning Techniques," *{IOP} Conference Series: Materials Science and Engineering,* vol. 1099, March 2021.

[6] S. Makaju, P. Prasad, A. Alsadoon, A. Singh and A. Elchouemi, "Lung Cancer Detection using CT Scan Images," *Procedia Computer Science,* vol. 125, pp. 107-114, 2018.

[7] B. G. and F. D. , "Lung cancer: clinical presentation and specialist referral time.," pp. 898-904, Dec 2004.

[8] S. K. K. M. and W. S. , "Radiologic classification of small adenocarcinoma of the lung: radiologic-pathologic correlation and its prognostic impact," February 2006.

[9] G. Xiuhua , S. Tao, W. Huan and L. Zhigang, "Prediction Models for Malignant Pulmonary Nodules Based-on Texture Features of CT Image 1," 2012.

[10] W. TY and B. NM, "Artificial Intelligence With Deep Learning Technology Looks Into Diabetic Retinopathy Screening," pp. 316(22):2366-2367, 13 December 2016.

[11] G. Zhang , L. Lin and J. Wang, "Lung Nodule Classification in CT Images Using 3D," *Journal of Physics: Conference Series,* 2021.

[12] N. Khehrah, M. S. Farid, . S. Bilal and M. K. Hassan, "Lung Nodule Detection in CT Images Using Statistical and Shape-Based Features," *Journal of Imaging,* vol. 6, 2020.

[13] C. S. Feng and D. Pan, "An improved approach of lungs image segmentations based on watershed algorithms," in *7th International Conference on Internet Multimedia Computing & Service*, 2015.

[14] M. Lavanaya and M. Kanan, "Lung Lesion Detection in CT Scan Images Using the Fuzzy Local Information Cluster Means (FLICM) Automatic Segmentation Algorithm and Back Propagation Network Classification," *Asian Pacific Journal of Cancer Prevention,* p. 3395–3399., 2017.

[15] T. S. Roy, N. Sirohi and A. Patle, "Classification of lung image and nodule detection using fuzzy inference system," in *International Conference on Computing, Communication Automation*, 2015.

[16] K. Mijung, Z. Jasper and W. De Neve, "Few-shot Learning Using a Small-Sized Dataset of High-Resolution FUNDUS Images for Glaucoma Diagnosis," 2017.

[17] M. Fink, "Object Classification from a Single Example Utilizing Class Relevance Metrics," in *Advances in Neural Information Processing Systems*, MIT Press, 2004, p. 449–456.

[18] A. Parnami and M. Lee, "Learning from Few Examples: A Summary of Approaches to Few-Shot Learning," 2022.

[19] M. H. and I. S. , "A Closer Look at Prototype Classifier for Few-shot Image Classification," 2022.

[20] D. Liu, X. Gao and Q. Shen, "Prototypical Network for Radar Image Recognition with Few Samples," *Journal of Physics: Conference Series,* vol. 1634, p. 012116, Sep 2020.

[21] J.-d.-T. O. a. M. and H. a. K. , "Cloud-Based Evaluation of Anatomical Structure Segmentation and Landmark Detection Algorithms: VISCERAL Anatomy Benchmarks," *IEEE Transactions on Medical Imaging,* vol. 35, pp. 1-1, 2016.

[22] J. Kotia, A. Kotwal, B. Rishika and R. Mangrulkar, "Few Shot Learning for Medical Imaging," *Studies in Computational Intelligence,* pp. 107-132, 01 2021.

[23] J. Bromley, J. Bentz, L. Bottou, I. Guyon and Y. Lecun, "Signature Verification using a "Siamese" Time Delay Neural Network," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 7, p. 25, 08 1993.

[24] Y. Yi, C. Chen and T. Zhang, "A Survey on Siamese Network: Methodologies, Applications and Opportunities," *IEEE Transactions on Artificial Intelligence,* vol. PP, pp. 1-21, 12 2022.

[25] G. R. Koch, "Siamese Neural Networks for One-Shot Image Recognition," 2015.

[26] K. Clark, B. Vendt, K. Smith, J. Freyman, J. Kirby, P. Koopel, S. Moore and P. Stanely, "The Cancer Imaging Archive (TCIA): Maintaining and Operating a Public Information Repository," *Journal of digital imaging,* vol. 26, 07 2013.

[27] J. Hofamanninger, F. Prayer, J. Pan, S. Röhrich and H. Prosch, "Automatic lung segmentation in routine imaging is primarily a data diversity problem, not a methodology problem," *European Radiology Experimental,* vol. 4, p. 50, 08 2020.

[28] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *LNCS,* vol. 9351, pp. 234-241, 10 2015.

[29] M. Drozdzal, E. Vorontsov, G. Chartrand and S. Kadoury, "The Importance of Skip Connections in Biomedical Image Segmentation," 08 2016.

[30] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ICLR 2015,* 09 2014.

[31] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," pp. 770-778, 2016.

[32] G. Huang, Z. Liu and K. Q. Weinberger, "Densely Connected Convolutional Networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR),* pp. 2261.-2269, 2016.

[33] J. Snell, K. Swersky and R. Zemel, "Prototypical Networks for Few-shot Learning," 2017.

[34] M. o. Defence, Machine Learning with Limited Data, DSTL/PUB126061.