MAKERERE UNIVERSITY

COLLEGE OF ENGINEERING, DESIGN, ART AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Deep Learning for Cervical Cancer Screening

submitted by
Raymond Nuwagaba (17/U/9292/PS) BSTE

Main Supervisor: Dr. Andrew Katumba
Co supervisor: Dr. Hanifa Nabuuma

A report Submitted to the Department of Electrical and Computer Engineering in partial fulfillment of the requirement of award for BSc. Telecommunications Engineering at Makerere University.

Declaration

No portion of the work in this document has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author. I have abided by the Makerere University academic integrity policy on this assignment.

Signature: .....

Date: 3rd March 2022

Approval

This report has been submitted with approval of the following supervisors:

Main supervisor
Dr. Andrew Katumba
Lecturer,
Department of Electrical and Computer Engineering,
College of Engineering, Design, Art and Design,
Makerere University.
Signature: .....
Date: 3rd March 2022

Co-supervisor
Dr. Hanifa Nabuuma
Lecturer,
Department of Electrical and Computer Engineering,
College of Engineering, Design, Art and Design,
Makerere University.
Signature: HANIFA NABUUMA
Date: 28th February 2022

Acknowledgement

I would like to thank the Almighty God for the gift of life and ability to complete the project. There are so many individuals and institutions to whom I am indebted in the writing of this report.

Firstly, I owe particular thanks to both my supervisors, Dr. Andrew Katumba and Dr. Hanifa Nabuuma who offered this area of study and granted me an opportunity to carry it on. Your guidance is all for this milestone. This analysis benefited greatly from thoughtful comments and several suggestions from colleagues and I wish to single out my project partner Mr. Kaddu Joshua. My family has been very supportive, cheerfully endured my uncommunicativeness and isolation during the time of this project tenure.

I also thank the Makerere university staff and management most especially those under the department of Electrical and Computer Engineering for coming up with such an educational program. Lastly, I wish to convey my sincere gratitude to Google Technology company for their services that have enabled us do this project with ease.

Abstract

Traditional screening of cervical cancer type classification majorly depends on the pathologist's experience, which also has less accuracy. Colposcopy is a critical component of cervical cancer prevention. In conjunction with precancer screening and treatment, colposcopy has played an essential role in lowering the incidence and mortality from cervical cancer over the last 50 years. However, due to the increase in workload, vision screening causes misdiagnosis and low diagnostic efficiency. Medical image processing using the convolutional neural network (CNN) model shows its superiority for the classification of cervical cancer type in the field of deep learning.

This project proposes deep learning CNN architectures to detect cervical cancer using the colposcopy images. The project was divided into two tasks: task one where we have a dataset containing images of three types of cervixes that is type 1, type 2 and type 3 and task two were we have a dataset containing images of cancerous (positive) and non-cancerous cervixes (negative). The datasets where obtained using a colposcope.

In task one the dataset was trained using deep learning CNN architectures; Yolo V5 and Yolo V4. In dataset one the Yolo V4 gave a better performance with a mAP of 0.646 the compared to the Yolo V5 which gave mAP of 0.283. In task two we trained the classification model and the object detection model. Under the classification model the xception model had a better performance with a training accuracy of 97.13%, validation accuracy of 89.01% and test accuracy of 91.3% compared to the Inception V3 model which gave a training accuracy of 88.21%, validation accuracy of 78.1% and testing accuracy of 75.9%. under the object detection model, we only trained using the Yolo V4 which gave mAP of 0.879424. In conclusion, with more datasets more GPU time the accuracy level of our model can be improved.

CONTENTS

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Networks
COCO	Common Objects in Context
FN	False Negative
FP	False Positive
FPS	Frames per second
COCO	Common Objects in Context
GPU	Graphics Processing Unit
HCME	Hierarchical convolutional Mixture of Experts
LED	Light Emitting Diode
mAP	Mean Average Precision
ONNX	Open Neural Network Exchange
PAN	Path Aggregation Network
ReLU	Rectified linear unit
SSP	Spatial pyramid pooling
TFlite	Tensorflow lite
TP	True Positive
TTA	Test Time Augmentation
VIA	Visual Inspection with Acetic Acid
VILI	Visual Inspection with Lugol's Iodine
WHO	World Health Organisation
YOLO	You Only Look Once

1. Introduction

This chapter provides explanation of the project background, problem statement, justification and objectives.

1.1 Background

Cervical cancer is a disease in which the cells of the cervix become abnormal and start to grow uncontrollably. This caused by the Human papillomavirus. Cervical cancer is the second most common cancer among women in the developing world and is responsible for 230,200 deaths and 444,500 cases annually [1, 2, 3]. It is a major cause of morbidity and mortality in resource-poor settings where access to cervical cancer screening and vaccination is limited. [4]. East Africa has the highest age-standardised incidence rates for cervical cancer at 42.7 per 100,000 women per year [2].

In Uganda, an estimated 33.6% of women in the general population harbor human papillomavirus, a necessary cause of cervical cancer, and 44 per 100,000 women develop the disease every year [5]. With 3,915 women diagnosed with cervical cancer annually, Uganda ranks 14th among countries with the highest incidence rates [5]. Amongst Ugandan women of reproductive age, the risk of developing cancer is high. Statistics by the World Health Organization (WHO) revealed that cervical cancer is the fourth most prevalent cancer globally, with a reporting rate of 5,70,000 new cases in 2018, accounting for 7.5% of all women cancer deaths [6]. Over 3,11,000 cervical cancer deaths per year were reported at around 85% in low- and intermediate-income countries, and the early diagnosis of cervical cancer offers a way of saving a life. [6]. Despite several medical and science advancements, this disease is not completely curable, mainly if diagnosed in a developing stages. Prevention and screening services, therefore, play a crucial role in the fight against cervical cancer.

The screening of cervical cancer follows a typical workflow: HPV testing, cytology or PAP smear testing, colposcopy, and biopsy. Several tools support the workflow which have been created to make it more effective, practical, and inexpensive. The PAP smear image screening is mostly employed for the treatment of cervical cancer, but it requires a greater number of microscopic examinations to diagnosis of cancer and noncancer patients, and also it is time consuming and requires trained professionals, but there is a chance of missing the positive cases by using the conventional screening method. The PAP smear and HPV testing are very costly treatment, and it also provides lower sensitivity. On the other side, the colposcopy treatment is widely used in the developing countries. To overcome the shortcomings in PAP smear images

and HPV testing, the colposcopy screening is used. Cervical cancer deaths can be avoided by successful screening schemes and can lead to lowered sickness and impermanence. [7]. In low- and middle-income nations, cervical cancer screening facilities are very sparse because of a shortage of qualified and educated health care staff and insufficient healthcare funding to fund screening systems [8]. Several works have taken various approaches for collecting details from images in digital colposcopy. These studies' key aim is to provide health practitioners with tools during colposcopy exams irrespective of their level of competence.

Previous studies have been developed in diagnosis using computer-aided systems for a range of tasks, including improvement and evaluation of image quality, regional segmentation, picture identification, identification of unstable regions and patterns, transition zone type classification (TZ) type, and cancer risk classification [9]. CAD instruments help improve the picture of cervical colposcopy and areas of concern segments and identify certain anomalies. These methods help clinicians to make diagnostic choices, but they should have adequate experience and expertise to make an appropriate diagnosis. The appearance of pathological regions may indicate such neoplasms; so in a colposcopy analysis, the detection of these lesions may be very critical. These abnormal areas include acetowhite, abnormal vascularization, mosaic areas, and punctures [10].

Deep learning has made significant advances in different applications such as computer vision, natural language processing, forecasting, and battery health monitoring [11]. Medical image processing, including classification, identification, segmentation, and registration, plays an essential role in disease diagnosis. Medical images such as MRI, CT, and ultrasound images and blood smear images, make up the vast majority of the image data processed [12]. Deep learning's multilayer neural network perception mechanism can learn more abstract features in images and is expected to address the issues that plague conventional medical CAD systems. However, the deep learning techniques should be supported with an extensive database, especially for positive cases. The convolution neural network (CNN) is used to identify MI signals in an efficient computer-aided diagnosis (CAD) framework for urban healthcare in smart cities [13].

1.2 Problem Statement

Incorrect diagnosis of cervical cancer due to the limited number of clinical oncologists in developing countries and sub-optimal performance of cervical cancer screening methods which leads to errors in prescriptions. Hence a need for a higher performance cervical cancer screening method to support the existing screening methods.

1.3 Justification

Cervical cancer is the second most common cancer among women in the developing world and is responsible for 230,200 deaths and 444,500 cases annually [1]. Studies in 27 developing countries show that a single clinical oncologist is catering to more than 1000 incident cancers.

As a result, delay and error rates attributed to available scientific screening methodology that is, pap smear test, Visual Inspection with Acetic Acid, Visual Inspection with Lugol's Iodine and colposcopy is relatively high. This creates a need to build a higher performance model to address this issue so as to effectively manage resources allocated to cervical cancer cases. Added advantages of developing a higher perform computer vision-based classification model are, the achievement of higher accuracy levels and quick delivery of in relation to available options.

1.4 Project Objectives

1.4.1 Main objective

To develop a deep learning model for cervical cancer screening.

1.4.2 Specific objectives

- To obtain and prepare the dataset.
- To train a classification and an object detection model for cervical cancer screening.
- To evaluate the performance of the model.
- To deploy the model on a smartphone.

2. Literature Review

This section contains literature of researches and works of various topics used in the process of our project

2.1 Cervical Cancer

Cervical cancer is a type of cancer that occurs in the cells of the cervix — the lower part of the uterus that connects to the vagina. Various strains of the human papillomavirus (HPV), a sexually transmitted infection, play a role in causing most cervical cancer. When exposed to HPV, the body's immune system typically prevents the virus from doing harm. In a small percentage of people, however, the virus survives for years, contributing to the process that causes some cervical cells to become cancer cells. You can reduce your risk of developing cervical cancer by having screening tests and receiving a vaccine that protects against HPV infection [14].

2.2 Artificial Intelligence

AI is based on four approaches and these are; the ability of a computer to acting humanly, thinking humanly, thinking rationally and acting rationally [15].

2.2.1 Applications of Artificial Intelligence

Among the applications of Artificial Intelligence, we are going to focus more on computer vision, since it's the application used in this project.

1. **Speech recognition.**
   It is also known as automatic speech recognition (ASR), computer speech recognition, or speech-to-text, and it is a capability which uses natural language processing (NLP) to process human speech into a written format. Many mobile devices incorporate speech recognition into their systems to conduct voice search e.g. Siri—or provide more accessibility around texting.

2. **Customer service**
   Online virtual agents are replacing human agents along the customer journey. They

answer frequently asked questions (FAQs) around topics, like shipping, or provide personalized advice, cross-selling products or suggesting sizes for users, changing the way we think about customer engagement across websites and social media platforms. Examples include messaging bots on e-commerce sites with virtual agents, messaging apps, such as Slack and Facebook Messenger, and tasks usually done by virtual assistants and voice assistants.

3. **Computer Vision**
   Computer vision is a field of artificial intelligence that trains computers to interpret and understand the visual world. Using digital images from cameras and videos and deep learning models, machines can accurately identify and classify objects — and then react to what they "see."

2.2.2 Computer Vision

Computer vision is a field in AI technology which enables computers and systems to derive meaningful information from digital images, videos and other visual inputs, and based on those inputs, it can take action. This ability to provide recommendations distinguishes it from image recognition tasks. Powered by convolutional neural networks, computer vision has applications within photo tagging in social media, radiology imaging in healthcare, and self-driving cars within the automotive industry [16].

2.2.2.1 Brief History of Computer Vision

Early experiments in computer vision took place in the 1950s, using some of the first neural networks to detect the edges of an object and to sort simple objects into categories like circles and squares. In the 1970s, the first commercial use of computer vision interpreted typed or handwritten text using optical character recognition. This advancement was used to interpret written text for the blind.
As the internet matured in the 1990s, making large sets of images available online for analysis, facial recognition programs flourished. These growing data sets helped make it possible for machines to identify specific people in photos and videos.

2.2.2.2 How Computer Vision Works

Computer vision needs lots of data. It runs analyses of data over and over until it discerns distinctions and ultimately recognize images. For example, to train a computer to recognize automobile tires, it needs to be fed vast quantities of tire images and tire-related items to learn the differences and recognize a tire, especially one with no defects.Two essential technologies are used to accomplish this: a type of machine learning called deep learning and a convolutional neural network (CNN).

Machine learning uses algorithmic models that enable a computer to teach itself about the context of visual data. If enough data is fed through the model, the computer will "look" at the data and teach itself to tell one image from another. Algorithms enable the machine to

learn by itself, rather than someone programming it to recognize an image.A CNN helps a machine learning or deep learning model "look" by breaking images down into pixels that are given tags or labels. It uses the labels to perform convolutions (a mathematical operation on two functions to produce a third function) and makes predictions about what it is "seeing." The neural network runs convolutions and checks the accuracy of its predictions in a series of iterations until the predictions start to come true. It is then recognizing or seeing images in a way similar to humans.

Much like a human making out an image at a distance, a CNN first discerns hard edges and simple shapes, then fills in information as it runs iterations of its predictions. A CNN is used to understand single images. A recurrent neural network (RNN) is used in a similar way for video applications to help computers understand how pictures in a series of frames are related to one another [17].

2.2.2.3 Techniques Used in computer Vision

1. **Image Segmentation**
   Image segmentation partitions an image into multiple regions or pieces to be examined separately [18]

2. **Object Detection**
   Object detection identifies a specific object in an image. Advanced object detection recognizes many objects in a single image: a football field, an offensive player, a defensive player, a ball and so on. These models use an X,Y coordinate to create a bounding box and identify everything inside the box [18]

3. **Facial Recognition**
   Facial recognition is an advanced type of object detection that not only recognizes a human face in an image, but identifies a specific individual [18].

4. **Edge Detection**
   Edge detection is a technique used to identify the outside edge of an object or landscape to better identify what is in the image [18].

5. **Pattern Detection**
   Pattern detection is a process of recognizing repeated shapes, colors and other visual indicators in images.

6. **Image Classification**
   Image classification groups images into different categories [18]

7. **Feature Matching**
   Feature matching is a type of pattern detection that matches similarities in images to help classify them [18].

.

2.2.2.4 Computer vision in healthcare

The goal of computer vision in healthcare is to make a faster and more accurate diagnosis that a physician could make. Currently, the most widespread use cases for computer vision and healthcare are related to the field of radiology and imaging.

Application of computer vision in health Radiology and oncology. Computer vision has broad application in healthcare but especially in the fields of radiology and oncology. The potential use cases include monitoring of tumor progression, bone fractures detection, and the search for metastases in the tissues. Breast cancer, lung cancer, leukemia, prostate cancer, and others are all malignancies that can be detected through computer-aided diagnosis. AI-powered solutions are designed to augment radiologists and make the medical image interpretation cheaper, faster, and more accurate [19].

1. **Dermatology**. Computer vision algorithms have a broad application in dermatology as well. Dermatology is mostly about a visual inspection of the patient's skin. And artificial intelligence has the power to enhance care. Specifically, computerized skin image analysis is leveraged for delivering personalized skincare (including skin treatments, makeup, facial creams and gels, humidity, etc.) for people based on their photos. It also has the potential to be used for the early detection of skin conditions, such as the diagnosis of skin cancer [19].

2. **Lab tests automation**.Cloud computing technology is also used for blood count, tissue cell analysis, change tracking, and other lab tests. Computer-vision powered blood analyzers either take images of blood samples or receive comprehensible input in the form of a picture of the already prepared slide containing a film of blood. As a rule, trained professionals take such images from a custom-designed camera attached to an ordinary microscope. Then, based on image processing and computer vision technologies, the system processes the input and automatically detects specific abnormalities in blood samples [19].

2.3 Machine Learning and Deep Learning

In the era of data science, artificial intelligence is trying to provide human kind intelligence to computers and for this, machine learning and deep learning are the technologies which are helping artificial intelligence to do it. Deep learning is confined version of machine learning. Illustration of the AI, machine learning and deep learning paradigm.

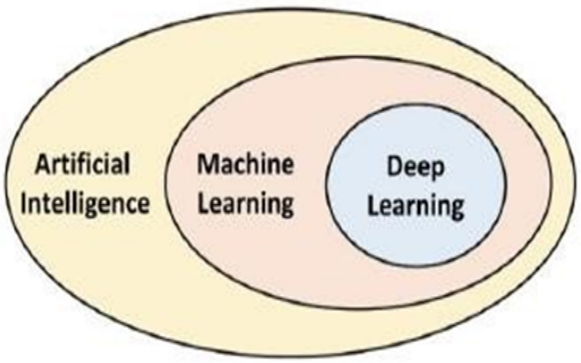

Figure 2.1: Illustration of the AI, machine learning and deep learning paradigm

2.4 Machine Learning

Machine learning is based on the idea that system can learn from data, identify the patterns and make decision with minimum human intervention. Machine learning algorithms build mathematical model based on sample data and then make the decision [20]. Machine learning procedures.
Machine learning incorporates four steps, given below;

- **Google colab**. This is a python development environment integrated with a Tesla K80 GPU with Random Access Memory of 12GB. It enabled us to train and test our machine learning models.
- Feature extraction
- Selection of corresponding machine learning algorithm
- Training and evaluating the data model's efficiency
- Using trained model for prediction Machine learning approaches.

Basically, Machine learning methods are broadly categorized in two categories that is; Shallow learning and deep learning [20]

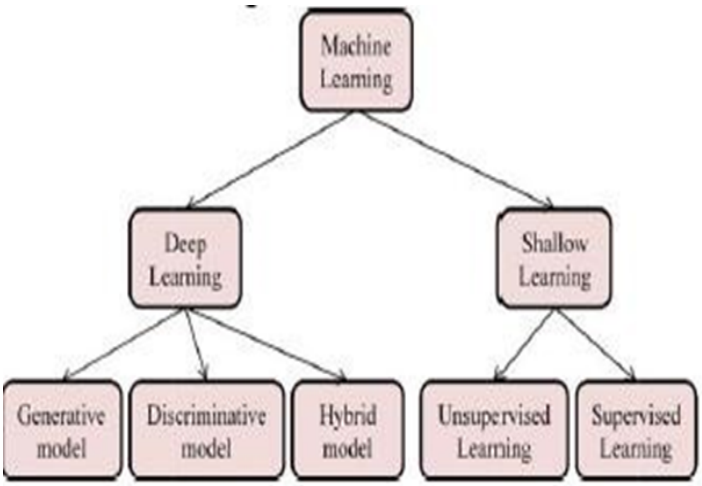

Figure 2.2: Illustration of Machine learning approaches

2.4.1 Shallow Learning

Shallow learning uses neural networks with single layers or Support Vector Machines. Shallow learning is broadly divided into two categories: Supervised and Unsupervised Learning. Though, there are also other methods of machine learning [20].

2.4.1.1 Supervised Learning

In supervised learning, algorithm builds a mathematical model from a set of data that contains both the input and desired outputs. This implies that algorithms are trained using labeled examples. An algorithm learns by comparing its actual output with correct outputs to find out errors and then, the model is modified accordingly.
Examples of Supervised learning are classification, regression, prediction and gradient boosting. Some examples of supervised machine learning are Nearest neighbour, Naïve Bayes, Decision Tree, Regression Tree etc. [20]

2.4.1.2 Unsupervised Learning

In unsupervised learning, a mathematical model is to be built from a set of data which contains only inputs. Unsupervised learning is used against that data which doesn't consists historical label. The objective in such a case would be to study the patterns in the input dataset to get better understanding and identify similar patterns that can be grouped into specific classes or events [21]. Examples of unsupervised learning are; clustering, dimension reduction, anomaly detection [21].

2.4.1.3 Semi-supervised learning

Semi-supervised algorithms are used to develop mathematical models from incomplete training data, where a portion of the sample input doesn't consist labels. This learning is useful when cost of labelling is too high to allow for fully labelled training process. In semi-supervised training methods, the model is initially trained on the part of dataset which has labels and then this model is used to estimate labels for the unlabeled data, which will be used to refine the model [20].

2.4.1.4 Reinforcement learning

The basic objective of reinforcement learning algorithms is to map situations to actions that yield the maximum final reward [21]. Examples of reinforcement learning techniques are; Q-learning, Markov decision process, temporal Difference methods, Monte-Carlo methods.

2.4.2 Deep Learning

Deep learning is a set of algorithms of machine learning which uses multiple layers that corresponds to different level of abstraction to each level. It consists of input layer, output layer and several hidden layer. It is used for voice synthesis, image processing, handwriting recognition, object detection, prediction analytics and decision making. Deep learning can

be broadly classified into three types that is Generative models, discriminative models and hybrid models [20].

2.4.2.1 Generative models

This is a powerful way of learning any kind of data distribution using unsupervised. All generative models aim at learning the true data distribution of the training set so to generate new data points with some variations. Since it is not possible to learn the exact distribution of the data either explicitly or implicitly, the power of the neural networks can be leveraged to learn a function which can approximate the model distribution to the true distribution. The most commonly used and efficient approaches are Variational Autoencoders and Generative Adversarial Networks [20]. Generative models use probability estimates and likelihood to model datapoints and distinguish between different class labels in a dataset. A generative model learns the joint probability distribution p (x, y). It predicts the conditional probability with the Bayes Theorem. Discriminative models tend to learn the boundary between classes or labels in a dataset that is; the goal here is to find the decision boundary separating one class from another.

2.4.2.2 Discriminative models

Are supervised machine learning models which learn the conditional probability distribution P(X/Y).

2.4.3 Deep learning architecture

Deep Learning architectures consist of multiple processing layers. Each layer is able to produce non-linear responses based on the data from its input layer.
The functionality of Deep Learning is imitated from the mechanisms of human brain and neurons for processing of signals [20]. A Deep Neural Network consists of an input layer, several hidden layers, and an output layer. Each layer includes several units called neurons. A neuron receives several inputs, performs a weighted summation over its inputs, then the resulting sum goes through an activation function to produce an output. Each neuron has a vector of weights associated to its input size as well as a bias that should be optimized during the training process [20].

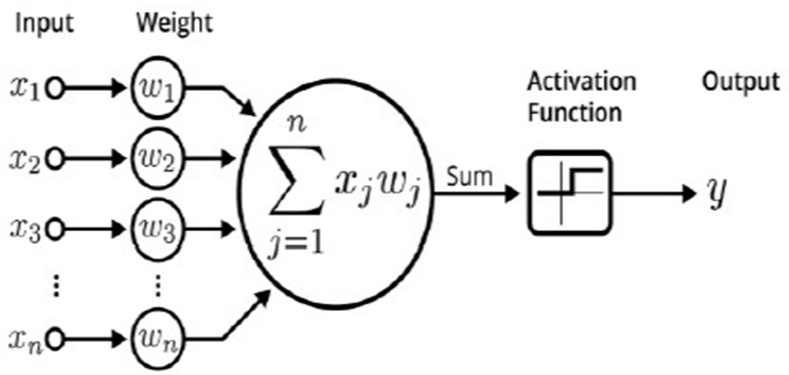

Figure 2.3: Illustration of a neural network architecture.

These artificial neurons are assigned sequentially which makes a chain as one neurons output becomes input of next neuron, and this process goes on over and over which makes an Artificial Neural Network. Deep Learning Neural Networks consists of more than one hidden layer [20]

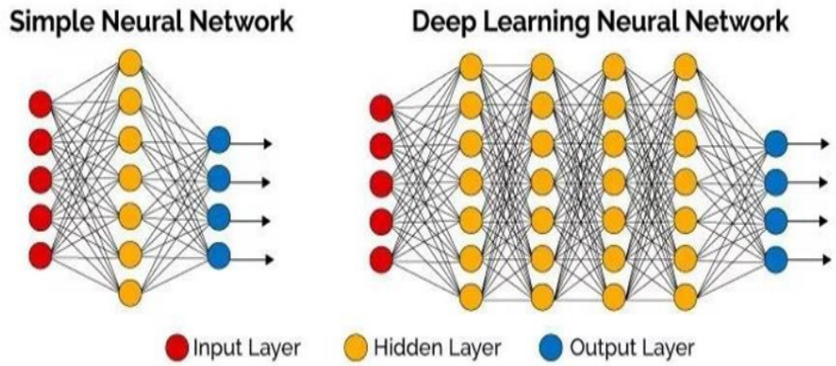

Figure 2.4: Shallow and deep neural networks.

2.5 Activation functions

Activation functions are functions used in neural networks to computes the weighted sum of inputs and biases, of which is used to decide if a neuron can be fired or not. It manipulates the presented data through some gradient processing usually gradient descent and afterwards produce an output for the neural network, that contains the parameters in the data [22]. Activation function can be either linear or non-linear depending on the function it represents. For a linear model, a linear mapping of an input function to an output, is performed in the hidden layers before the final prediction of class score. The input vectors x transformation is given by;

$$f(x) = wTx + b, \tag{2.1}$$

where x = input, w = weights, b = biases. Furthermore, the neural networks produce linear results from the mappings from equation (1).
The output of these models is given by;

$$y = w1x1 + w2x2 + ... + wnxnb, \tag{2.2}$$

However, since the output are linear in nature, the nonlinear activation functions are required to convert these linear inputs to non-linear outputs as this aids the learning of high order polynomials beyond one degree for deeper networks. The non-linear output after the application of the Activation Function is given by [22]

$$y = \alpha(w1x1 + w2x2 + ... + wnxnb), \tag{2.3}$$

where α is the activation function.
A special property of the non-linear activation functions is that they are differentiable else they cannot work during backpropagation of the deep neural networks.

2.5.1 Types of activation functions

2.5.1.1 Sigmoid function

The Sigmoid is a non-linear Activation Function used mostly in feedforward neural networks. The Sigmoid function is given by the [22];

$$\frac{1}{1+e^{-x}} \tag{2.4}$$

The sigmoid function appears in the output layers of the Deep Learning architectures, and they are used for predicting probability-based output and has been applied successfully in binary classification problems, modelling logistic regression tasks as well as other tasks. These are used mostly in shallow networks. The Sigmoid AF suffers major drawbacks which include sharp damp gradients during backpropagation from deeper hidden layers to the input layers, gradient saturation, slow convergence and non-zero centered output thereby causing the gradient updates to propagate in different directions [22].

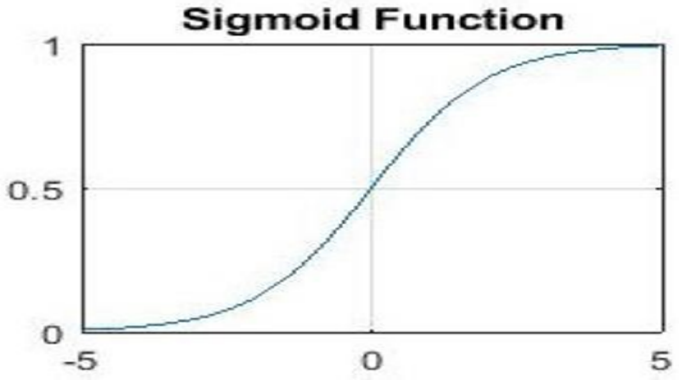

Figure 2.5: sigmoid function

2.5.1.2 Hyperbolic Tangent Function

The hyperbolic tangent function known as tanh function, zero-centered function whose range lies between -1 to 1, thus the output of the tanh function is given by;

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.5}$$

The tanh function became the preferred function compared to the sigmoid function in that it gives better training performance for multi-layer neural networks [22]. However, the tanh function could not solve the vanishing gradient problem suffered by the sigmoid functions as well. The main advantage provided by the function is that it produces zero centered output thereby aiding the back-propagation process.
A property of the tanh function is that it can only attain a gradient of 1, only when the value of the input is 0, that is when x is zero. This makes the tanh function produce some dead neurons during computation [22].

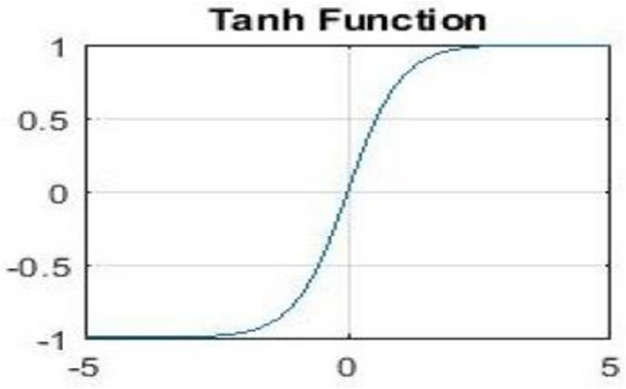

Figure 2.6: Hyperbolic Tangent Function

This is also known as the threshold Activation Function and this implies that after a certain threshold, a neuron is activated and below the that threshold, the neuron is deactivated. This is used for binary classifications [22].
The step Activation function has zero gradient thus causing a hindrance during the backpropagation implying that the weights and biases don't update.

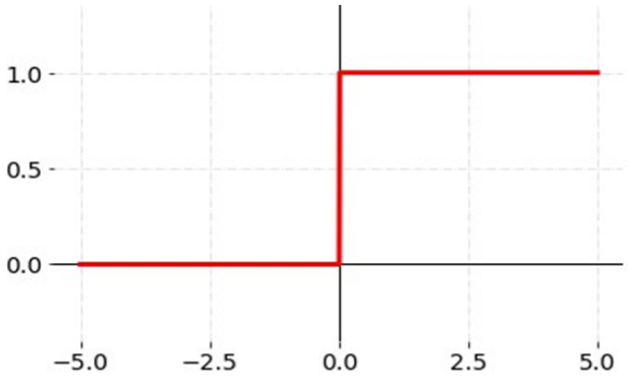

Figure 2.7: Step function

$$f(x) = 1, for x >= 0 \tag{2.6}$$

2.6 Convolutional Neural Networks

A convolutional neural network (CNN) is a type of artificial neural network used in image recognition and processing that is, specifically designed to process pixel data. CNN is a particular type of feed-forward neural network in AI. CNN is widely used for image recognition. CNN represents the input data in the form of multidimensional arrays. It works well for a large number of labeled data. CNN extract the each and every portion of input image, which is known as receptive field. It assigns weights for each neuron based on the significant role of the receptive field. So that it can discriminate the importance of neurons from one another.

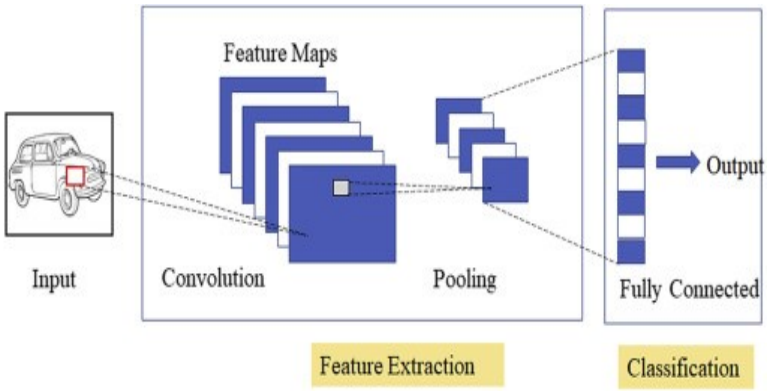

Figure 2.8: Convolutional Neural Network Architecture

2.6.1 How CNN work

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

- Convolutional layer
- Pooling layer
- Fully-connected (FC) layer

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object [23].

2.6.1.1 Convolutional Layer

The convolutional layer is the core building block of a CNN, and it is where the majority of computation occurs. It requires a few components, which are input data, a filter, and a feature map. Let's assume that the input will be a color image, which is made up of a matrix of pixels in 3D. This means that the input will have three dimensions a height, width, and depth which correspond to RGB in an image. We also have a feature detector, also known as a kernel or a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.
The feature detector is a two-dimensional (2-D) array of weights, which represents part of the image. While they can vary in size, the filter size is typically a 3x3 matrix; this also determines the size of the receptive field. The filter is then applied to an area of the image, and a dot product is calculated between the input pixels and the filter. This dot product is then fed into an output array. Afterwards, the filter shifts by a stride, repeating the process until the kernel has swept across the entire image. The final output from the series of dot products from the input and the filter is known as a feature map, activation map, or a convolved feature.

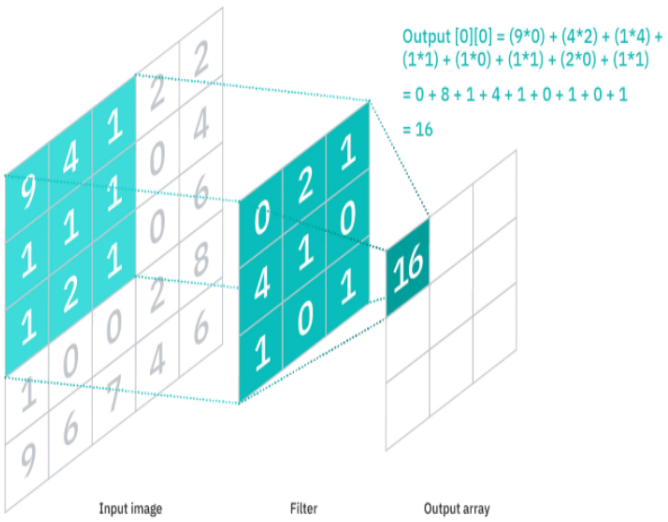

Figure 2.9: Convolutional Layer

As you can see in the image above, each output value in the feature map does not have to connect to each pixel value in the input image. It only needs to connect to the receptive field, where the filter is being applied. Since the output array does not need to map directly to each input value, convolutional (and pooling) layers are commonly referred to as "partially connected" layers. However, this characteristic can also be described as local connectivity. Note that the weights in the feature detector remain fixed as it moves across the image, which is also known as parameter sharing. Some parameters, like the weight values, adjust during training through the process of backpropagation and gradient descent. However, there are three hyperparameters which affect the volume size of the output that need to be set before the training of the neural network begins. These include:

1. The number of filters affects the depth of the output. For example, three distinct filters would yield three different feature maps, creating a depth of three.

2. Stride is the distance, or number of pixels, that the kernel moves over the input matrix. While stride values of two or greater is rare, a larger stride yields a smaller output.

3. Zero-padding is usually used when the filters do not fit the input image. This sets all elements that fall outside of the input matrix to zero, producing a larger or equally sized output. There are three types of padding:

    - Valid padding: This is also known as no padding. In this case, the last convolution is dropped if dimensions do not align.
    - Same padding: This padding ensures that the output layer has the same size as the input layer.
    - Full padding: This type of padding increases the size of the output by adding zeros to the border of the input.

After each convolution operation, a CNN applies a Rectified Linear Unit (ReLU) transformation to the feature map, introducing nonlinearity to the model.

As we mentioned earlier, another convolution layer can follow the initial convolution layer. When this happens, the structure of the CNN can become hierarchical as the later layers can see the pixels within the receptive fields of prior layers. As an example, let's assume that we're trying to determine if an image contains a bicycle. You can think of the bicycle as a sum of parts. It is comprised of a frame, handlebars, wheels, pedals, et cetera. Each individual part of the bicycle makes up a lower-level pattern in the neural net, and the combination of its parts represents a higher-level pattern, creating a feature hierarchy within the CNN. Ultimately, the convolutional layer converts the image into numerical values, allowing the neural network to interpret and extract relevant patterns.

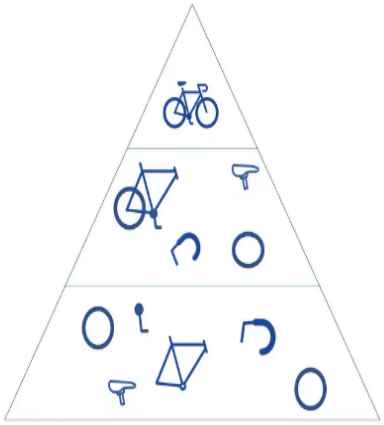

Figure 2.10: Feature hierarchy within the CNN

2.6.1.2 Pooling Layer

Pooling layers, also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input. Similar to the convolutional layer, the pooling operation sweeps a filter across the entire input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array. There are two main types of pooling:

- Max pooling: As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.
- Average pooling: As the filter moves across the input, it calculates the average value within the receptive field to send to the output array. While a lot of information is lost in the pooling layer, it also has a number of benefits to the CNN. They help to reduce complexity, improve efficiency, and limit risk of overfitting. Fully-Connected Layer

2.6.1.3 Fully-Connected Layer

The name of the full-connected layer aptly describes itself. As mentioned earlier, the pixel values of the input image are not directly connected to the output layer in partially connected layers. However, in the fully-connected layer, each node in the output layer connects directly to a node in the previous layer.
This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLu functions, FC layers usually leverage a softmax activation function to classify inputs appropriately, producing a probability from 0 to 1.

2.6.2 Architectures of Convolutional Neural Network

- LeNet-5
- Alex Net
- VGG
- Inception and Google Net
- Residual Network or ResNet

2.6.3 How to optimize CNN

Optimization is the process of changing the attributes of the neural network such as weights and learning rate to reduce the losses [24].
The different types of optimization algorithms used so as to minimize the loss function are;

- Gradient Descent
- Stochastic Gradient Descent (SGD)
- Mini Batch Stochastic Gradient Descent (MB-SGD)
- SGD with momentum
- Nesterov Accelerated Gradient (NAG)
- Adaptive Gradient (AdaGrad)
- AdaDelta
- RMSprop
- Adam

2.6.3.1 Gradient Descent

It's used heavily in linear regression and classification algorithms. Backpropagation in neural networks also uses a gradient descent algorithm. Gradient descent is a first-order optimization algorithm which is dependent on the first order derivative of a loss function. It calculates that which way the weights should be altered so that the function can reach a minimal. Through backpropagation, the loss is transferred from one layer to another and the model's parameters also known as weights are modified depending on the losses so that the loss can be minimized [24].

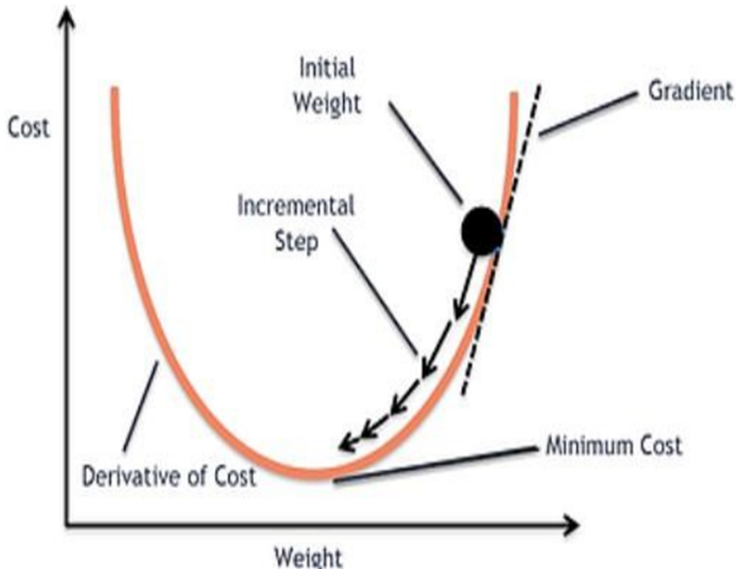

Figure 2.11: Gradient descent illustration

2.6.3.2 Stochastic Gradient Descent

It's a variant of Gradient Descent. It tries to update the model's parameters more frequently. In this, the model parameters are altered after computation of loss on each training example. So, if the dataset contains 1000 rows SGD will update the model parameters 1000 times in one cycle of dataset instead of one time as in Gradient Descent [24].
Adaptive Gradient Descent (AdaGrad) The key idea of AdaGrad is to have an adaptive learning rate for each of the weights. It performs smaller updates for parameters associated with frequently occurring features, and larger updates for parameters associated with infrequently occurring features.

2.6.3.3 AdaDelta

The problem with the previous algorithm AdaGrad is learning rate becomes very small with a large number of iterations which leads to slow convergence. To avoid this, the AdaDelta algorithm has an idea to take an exponentially decaying average. Adadelta is a more robust extension of Adagrad that adapts learning rates based on a moving window of gradient updates, instead of accumulating all past gradients. This way, Adadelta continues learning even when many updates have been done. Compared to Adagrad, in the original version of Adadelta,

you don't have to set an initial learning rate.
Adam (Adaptive Moment Estimation) Adam computes adaptive learning rates for each parameter. In addition to storing an exponentially decaying average of past squared gradients like Adadelta and RMSprop, Adam also keeps an exponentially decaying average of past gradients, similar to momentum. Whereas momentum can be seen as a ball running down a slope, Adam behaves like a heavy ball with friction, which thus prefers flat minima in the error surface. Adam provides better accuracy than other optimization models for neural network optimization [24].

2.7 Hyper parameters

Hyper parameters are parameters whose values are used to control the learning process and these are divided into two categories that is; Optimization. hyper parameters and Model Hyper parameters [25].

2.7.1 Optimization Hyper parameters

These are related to optimization and training process. Examples of these hyper parameters include;

2.7.1.1 Learning rate

The learning rate should be set to an optimal value so as to prevent longer time of epochs for the model converge or to stop the model from overshooting the ideal state for the case of learning rates that are smaller and larger than the optimal value respectively [25].

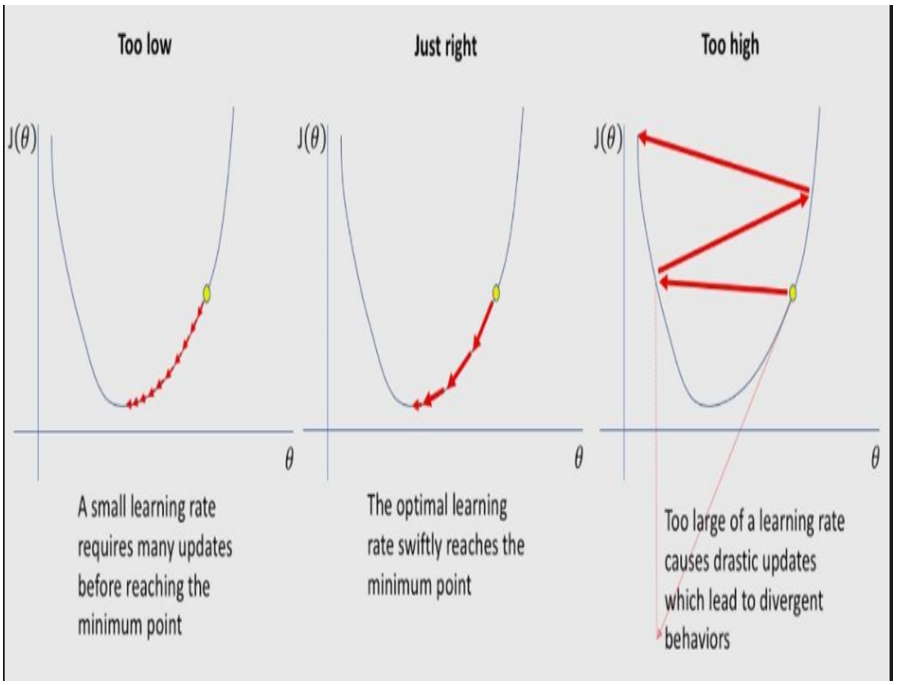

Figure 2.12: Learning rate illustration

2.7.1.2 Mini-Batch size

Batch size has an effect on the resource requirements of the training process, speed and number of iterations in a non-trivial way. A larger mini-batch size allows computational boost that utilize matrix multiplication in the training calculations however, it requires more memory for the training process. On the other hand, smaller mini-batch size induces more noise in error calculations [25].

2.7.1.3 Number of epochs

The number of epochs for training a model is determined by the validation error that is model is trained for as many number of iterations as long as the validation error keeps decreasing [25].

2.7.2 Model hyper parameters

These are more involved in the structure of the model. Examples of these hyper parameters are;

2.7.2.1 Number of hidden units

The number of units is the main measure of the model's learning capacity. Simple functions require a fewer number of hidden units. This implies the more complex the function, the more learning capacity the model will need. According to empirical observation, setting the number of hidden units larger than the number of inputs tends to enable better results in number of tasks. However, much larger number of units than the required lead to overfitting.

2.7.2.2 Number of layers

The deeper neural networks are, the better the performance.

2.8 Performance evaluation metrics

2.8.1 Accuracy

Accuracy is calculated as the fraction of predictions that are correct [26] and is defined by equation 2.7.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{2.7}$$

2.8.2 Precision

Precision score is calculated as the fraction of examples that are correctly predicted to be positives from among all the examples that have been predicted positive (correctly or incorrectly)

[26]. It is defined by equation 2.8

$$Precision = \frac{TP}{TP + FP}, \tag{2.8}$$

where; TP - True Positive FP - False Positive

2.8.3 Recall

Recall is a metric that quantifies the number of correct positive predictions made out of all positive predictions that could have been made [26] and this is defined by equation 2.9.

$$Recall = \frac{TP}{TP + FN}, \tag{2.9}$$

where; FN - False Negative

2.8.3 Recall

The mean average precision (mAP) is a metric used to measure the performance of models doing object detection tasks [26]. mAP is defined by equation 2.10.

$$mAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}, \tag{2.10}$$

where Q is the number of queries in the set and AveP(q) is the average precision (AP) for a given query, q

2.9 Related Work

Several algorithms were utilized for machine learning, and their segmentation refining was matched to a cervical cancer classifier in which random forests showed the best output [27]. Also, robust refinement methods have been used to manage, and unattended learning approaches to the different image or superpixel patches from extracted objects methods include Adaboost detectors [28]. SVM supports [29] , or Gaussian mixture models [30]. A novel Markov random field segmentation based on superpixels was proposed and implemented for nonoverlapping cells [31]. The multifilter SVM is executed, and the parameters were set for the identification of cervical cells [32]. It was suggested that cervical cell classification using artificial neural networks (ANN) was built and tested with a precision of 78% [33]. Unbalanced medical evidence for the variety of cervical cancer without any parameter change was addressed using an unsupervised approach [34]. The particle swarm optimization (PSO) with KNN membership values outperformed all other fundamental classification models [35]. The cervical cancer cell is classified using shape and texture characteristics of the segmentation and classification method and Gabor characteristics. It was found that a greater accuracy of 89% was obtained for both normal and cancer cell classification [36]. The extracted features from CNN were classified using the least square support vector machine (LSSVM)

and produced more remarkable results, one of the suggested model's reference components [37]. Radial basis function- (RBF-) SVM also obtained a strong outcome and outperformed logistic regression and random forest methods [38]. Based on the features, it was found that the accuracies were ranged from 90 to 95%.

New deep architectures such as ResNet, Inception, and tree models [39] have recently shown promising results in many applications and detect cancer cells. As one of the deep learning methods, the convolutional neural networks is the commonly used technique to identify and recognize cervical cancer [40]. Early cervical cancer cell identification and classification method based on CNN's was developed to extract deep learned features from the cervical images [41]. The extreme learning machine (ELM) was used to categorize the input images. The CNN paradigm was used for fine-tuning and transfer learning. Alternatives to classifiers based on the ELM, the multilayered perceptron (MLP), and the automotive encoder (AE) were also studied. It was reported that the stacked soft-max autoencoder reported a 97.25% precision on the cervical cancer dataset [42]. It was concluded that a tentative effort was made to tackle the issue of patient risk prediction using the applications for machine learning to grow cervical cancer. The machine learning software with cervical screening was used to tackle the problem of predicting the patient's risk [43]. They concentrated on the transition of information between linear classifiers to related activities to predict the patient's risk. Since the related risk factors in the population are highly sparsely influenced, the techniques for reducing dimensionality can boost the power of predictive machine learning models [44]. However, several projects benefit from reducing dimensionality and classification by using suboptimal methods in which each part is learned separately [45]. For the efficient collection and classification of cell properties in cervical smeared images [46], a quantum hybrid- (QH-) innovative approach was combined with adaptive search capability of the quantum-behaved particle swarm optimization (QPSO) method with the intuitionist reasonableness of the standard fuzzy K-nearest neighboring (fuzzy k-NN) algorithm (known simply as Q-fuzzy approach).

A model was suggested for the cervical cancer prediction model (CCPM) that produces an early prediction of cervical cancer with input risk factors [47]. CCPM eliminates outliers first by employing outlier identification methods such as Density-Based Spatial Noise Cluster (DBSCAN) and isolation Forest (iForest) by balancing the number of cases in the dataset. This approach has shown greater accuracy in cervical cancer forecasting. To design an integrated cervical cell diagnostic and screening device, the authors have developed a new Regionally Growing Extraction Function (RGBFE) to extract diagnostic features from the images [48]. Data from the cervical cell images with extracted features were supplied into the intelligent diagnostic component. Precancerous phases were forecasted using a new architecture called the Hybrid Multilayered Perceptron (H2MLP) network using an artificial neural network is created. The cells are classified into normal, low-quality intraepithelesis (LSIL), and high-quality intraepithelesis (HSIL). Improved screening systems are also inaccessible in developing countries, owing to the difficulty and time-consuming nature of manually screening irregular cells from a cervical cytology specimen. This system focused on transfer learning, and pretrained and densely connected convolutional networks are used to suggest a computer-aided diagnostic

(CAD) method for automated cervical image classification to assess CIN2 or higher-level lesions in the cervical imaging (ImageNet and Kaggle). The effect of various training strategies on model results, including scratch random initialization (RI), pretrained model (FT) tuning, different size of training data, and -fold cross validation, was evaluated. Experimental findings demonstrated accuracy of 73.08% for 600 test images [49]. The summary of the literature related to the screening of cervical cancer is provided in the table.

Table 2.1: summary of the literature related to the screening of cervical cancer

S.NO	Methods	Dataset	Advantages	Disadvantages
1	Inception V3 model [6]	Herley dataset	(i)High accuracy. (ii)Good Universality. (iii)Low complexity	(i)The deep learning needs further study to investigate cervical cells
2	Transfer learning, pretrained DenseNet [49]	Fujian Maternal and child health hospital Kaggle	(i)More feasibility and effective	(i) Limited data
3	CNN-extreme learning machine-(ELM-) based system [10]	Herlev dataset	(i)Fast learning (ii)Easy convergence (iii)Less randomized	(i)More complexity (ii)Need more investigation
4	Gene-assistance module, voting strategy [50]	Chinese hospital and Universitario De Caracas, Venezuela	(i)More scalable and practical	(i)Limited datasets
5	Random forest and Adaboost [26]	Radiotherapy dataset	(i)Better treatment planning	(i)Need to extract features (ii)Painful treatment
6	ColpoNet [28]	Colposcopy images	(i)Better accuracy (ii)Efficient classification	(i)Need to improve accuracy by extracting relevant information

7	CNN Model [29]	Papanicolaou-stained cervical smear dataset	(i)Better sensitivity and specificity	(i) Reported 1.8 percent false-negative images
8	Fourier transform and machine learning methods. [30]	Microscopic images	(i)Fully automatic system (ii)Saving precious time for the microscopist	(i)The level of complexity is more
9	CNN-SVM model [33]	Herlev and one private dataset	(i)Good robustness (ii)Highest accuracy	(i)Need improvement to adjust parameter (ii)Need of hand-crafted features
10	Stacked Autoencoder [39]	UCI database	(i)High accuracy (ii)Reduced data dimension	(i)Training time is very high due to reducing the dimension
11	PSO with KNN algorithm [45]	Cervical smear images	(i)Better accuracy (ii)Good feature selection	(i)Time-consuming due to two-phase feature selection
12	Ensemble model [46]	PAP smear image	(i)For 2 class problem achieves the accuracy of 96(ii)For 7 class problem achieves an accuracy of 78	(i)Overall of cells are difficult to identify
13	Multimodal deep network [51]	National Cancer Institute	(i)Good correlation (ii)High accuracy (iii)Learn better complementary features	(i)More complexity in image fusion

Owing to the millions of cells that a pathologist must examine, Pap smear screening takes longer days for analyses. Deep learning models were used to identify all cells and other materials present in the Pap smear image screening. The system is often difficult to classify since two cells overlap. To address the need for this problem, meticulously annotated data is required; developing this form of the medical field dataset is very difficult. Considering the challenges mentioned above, a novel deep learning model for cervical cancer screening via colposcopy is proposed. The significant aspects of using colposcopy images for cervical cancer screening are that it provides more focus to the patients because it is a simple and noninvasive procedure (no need to introduce instruments into the body). When compared to the other tests, the colposcopy dataset array is sparse. The automated classification of cervical cancer from colposcopy images helps mass screening for medical professionals to quickly determine whether further diagnostic checks are necessary. This paper presents the computerized system for cervical cancer prediction using colposcopy images. The critical contribution of the article is as follows:

1. This research is aimed at developing automatic cervical cancer detection from colposcopy images using the proposed deep convolutional neural network named CYENET. Unlike previous work reported in the literature, this proposed method does not require segmentation and feature engineering stages; it can also extract the discriminative features using ensemble approaches.

2. The transfer learning approach is used by fine-tuning the VGG19 model, which is widely used for medical image processing to predict accuracy. Besides the extensive experiment on the cancerous and noncancerous colposcopy images to effectively demonstrate the proposed CYENET (colposcopy ensemble network) and pretrained VGG model with recently proposed methods, and our proposed method achieves better accuracy as compared with the existing method in terms of classifying cervical cancer from colposcopy images.

3. The transfer learning approach is used by fine-tuning the VGG19 model, which is widely used for medical image processing to predict accuracy. Besides the extensive experiment on the cancerous and noncancerous colposcopy images to effectively demonstrate the proposed CYENET (colposcopy ensemble network) and pretrained VGG model with recently proposed methods, and our proposed method achieves better accuracy as compared with the existing method in terms of classifying cervical cancer from colposcopy images.

4. Intel ODT dataset is used for experimentation. The data augmentation technique is performed on the colposcopy images to prevent the trained model's overfitting problem. This technique is an efficient strategy to learn the particular features to achieve superior accuracy.

    (a) Another significant contribution of this paper is the use of occlusion sensitivity maps to visualize the picture characteristics of cervical cancers for classification purposes.

3. Methodology

This chapter comprises of the steps taken to develop a cervical cancer detection model and to deploy it on a mobile colposcopy.

3.1 Introduction

The main objective of this project was to develop a deep learning model for cervical cancer screening. The project was divided into task one and task two. Task one involved training an object detection model to detect various types of cervixes that were divided into type 1, type 2, and type 3. Task two involved training a classification and an object detection model to detect cancerous and non-cancerous cervixes.

3.2 Tools Used

The tools used for this project were;

3.2.1 Google Drive

Google drive is a cloud-based storage solution that allows you to save files online and access them anywhere from any smartphone, tablet, or computer. You can use Drive on your computer or mobile device to securely upload files and edit them online. Drive also makes it easy for others to edit and collaborate on files. In this project google drive was used to store and share datasets, google drive was also enabled us to use google colab which allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.

3.2.2 Google Colab

Google colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupiter notebook service that requires no setup to use, while providing free access to computing resources including GPUs. Colab notebooks are stored in Google Drive. Generally, Colab was being used to access GPUs from google in order to train our machine learning model.

3.2.3 Smartphone

A smartphone is a portable device that combines mobile telephone and computing functions into one unit. In this case a smartphone was used to install an app that was developed for deployment of our model.

3.2.4 Anaconda

Anaconda is a distribution of the python and R programming languages for scientific computing (data science, machine learning applications, predictive analytics, and other applications). We used the anaconda environment to convert the weights of the trained models into Tensorflow Lite (tflite) format which is compatible with the Android environment.

3.2.5 Android Studio

This is a software environment where one can build applications for android phones, tablets, televisions and other android platforms. We used the android studio to integrate the Tensorflowlite model with an android application so as to deploy and use of the model on a cell based mobile colposcopy.

3.3 Task One

Task one consisted of images with cervical cancer types that is 226 Type 1 images, 425 Type 2 images, and 231 Type 3 as shown below.

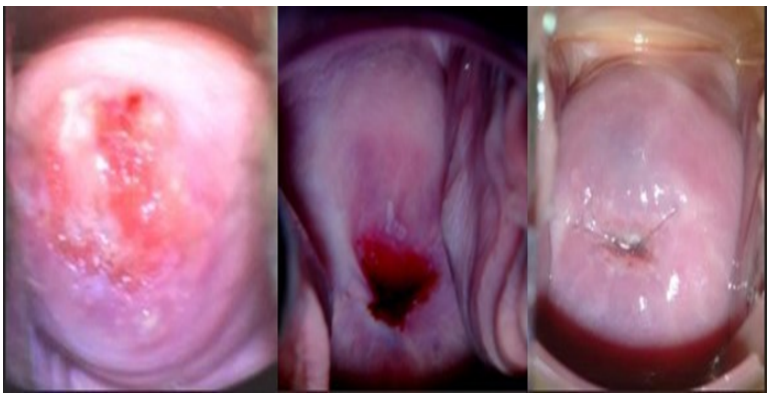

Figure 3.1: Type 1, Type 2 and Type 3 cervical cancer images.

3.3.1 Data collection

Dataset of task one was obtained from Kaggle under intel. This dataset was under MobileODT cervical cancer screening competition. This dataset was already annotated by gynecologist. Kaggle is a website where you can find competitions to solve data science problems. Kaggle allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers.

3.3.2 Data Processing

We used a web-based platform called roboflow to carry out our data processing. The images were resized to 640 x 640 (stretch) and the following augumentation techniques were applied;

- Auto-orientation of pixel data
- The hue was adjusted between -20 and +20
- We adjusted the random exposure between -20 and +20 percent
- We adjusted the random brightness between -35 and +35 percent

After the data had been annotated, it was converted to the YOLO format.

3.3.3 Data Division

In task one, the dataset was divided into training, validation and test sets. The training set had 1455 images, the validation set had 265 images and the test set had 132 images.

3.3.4 Training Process

In this stage, the dataset was trained several times using different object detection models in order to obtain the best suitable model. We trained our dataset with a yolov5 but faced a challenge when we attempted to deploy it as the model would crush after doing the necessary conversions aimed at making it compatible with the android environment. This was the same case with other models like SSD ResNet, Fast-RCC etc. This made us narrow down to a yolov4 darknet model as there are clear procedures on how it can be deployed and its performance is relatively good. We trained the model for 6000 steps using a batch size of 16 and an image size of 416 by 416. Yolo V4 was also giving us better results than the Yolo V5 after training. Generally, the training process involved;

1. Setting up a GPU environment and in this case we used google colab.
2. Cloning a yolo darknet github repository
3. We downloaded pre-trained weights which were going to train the model train the model on the images.
4. Setting up various parameter to be used such as number of steps in this case we used 6000 steps and the batch size which was 16.
   The steps above were carried out for both the Yolo v4 and the Yolo V5 model

3.3.5 Deployment

The yolo v4 model trained was converted to a Tensor Flow Lite file. Tensor flow Lite (flite) machine learning framework enables the deployment of machine learning models on multiple devices and surfaces such as mobile phones (iOS and Android), desktops and other edge

devices. The model was then deployed on an Android platform using Android studio. We converted the yolov4 weights to tflite format which is compatible with the android format and used android studio to create an android application. The inference time was about 7000 to 9000ms.

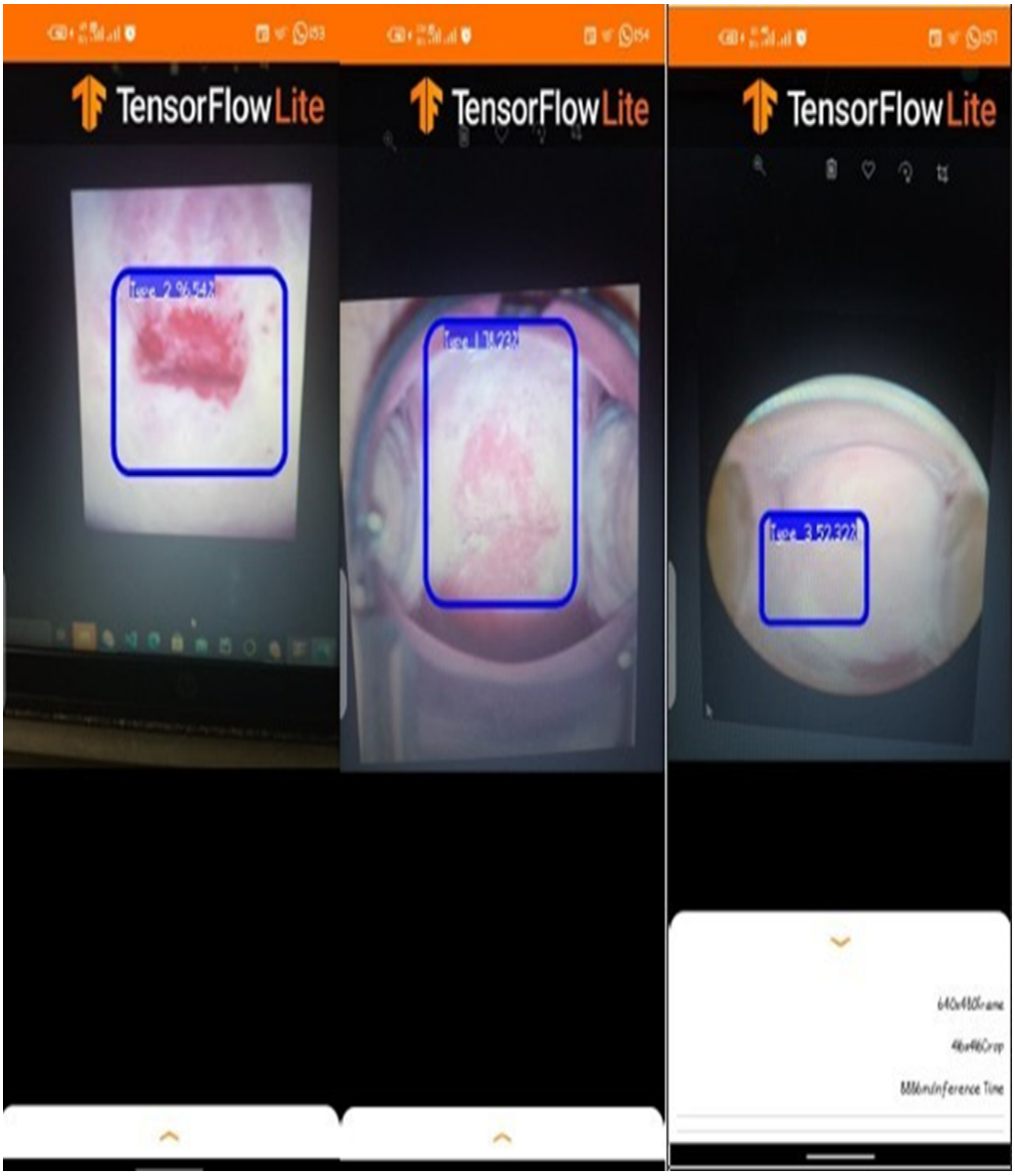

Figure 3.2: Android application showing cervix types

3.4 Task Two

Task two consisted of two classes of images that is; 383 positive images implying the cervices were cancerous and 1886 negative images implying non-cancerous cervices.

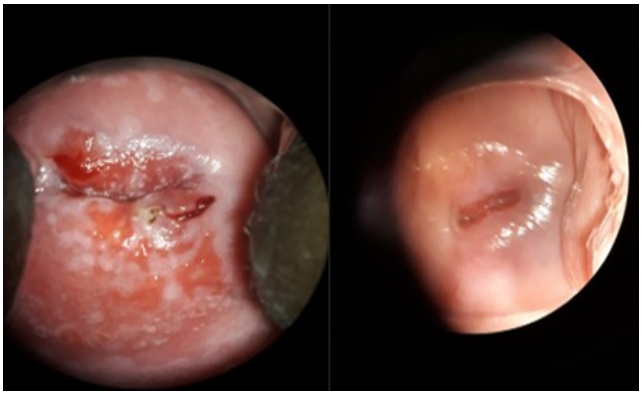

Figure 3.3: Cancerous (positive) and non-cancerous (negative) cervix images

3.4.1 Data Collection

Task two dataset was collected from different sites around Uganda such as Mbarara, Arua, Uganda Cancer Institute, Jinja and Mayuge by different clinicians. The dataset images were captured by a colposcope and cases were determined using VIA. The dataset was uploaded on a server from which it was made accessible to a gynecologist for ground truth labelling.

3.4.2 Data Processing

We used a web-based platform called roboflow to carry out our data processing. The images were resized to 640 x 640 (stretch). Due to a large class imbalance of the dataset, the images in the positive class were augmented to increase the image number for both classification and object detection using augmentation types namely; Random exposure adjustment of between -20 and +20 percent Hue adjusted between -20 and +20 The effect of image augmentation on the positive class of the dataset is shown in table 3.1

Table 3.1: Augmentation effect on the number of images in the positive class.

TASK	BEFORE AUGMENTATION	AFTER AUGMENTATION
OBJECT DETECTION	383	968
CLASSIFICATION	383	968

3.4.3 Data Division

In task two, the dataset was divided into training, validation and test sets. The training set had 2240 images, the validation set had 512 images and the test set had 102 images.

3.4.4 Training Process

Task two dataset was divided into training, validation and test set in a ratio of 0.7:0.2:0.1. The dataset was annotated and the annotation file was converted from COCO PASCAL format to the YOLO format. The training process involved the following steps;

1. Setting up a GPU environment and in this case we used google colab.
2. Installing the required dependencies for the GPU
3. Cloning a yolo darknet github repository
4. We downloaded pre-trained weights which were going to train the model
5. Setting up various parameter to be used such as number of step in this case we used 6000 steps and the batch size which was 16.

The yolov4 model was trained for 6000 steps and a batch size of 16 was used. The yolov5 was trained for 6000 steps and a batch size of 16 was used.

3.4.5 Deployment

The yolo v4 model trained was converted to a Tensor Flow Lite file. Tensor flow Lite (flite) machine learning framework enables the deployment of machine learning models on multiple devices and surfaces such as mobile phones (iOS and Android), desktops and other edge devices. The model was then deployed on an Android platform using Android studio. We converted the yolov4 weights to flite format which is compatible with the android format and used android studio to create an android application. The inference time was about 7000 to 9000ms.

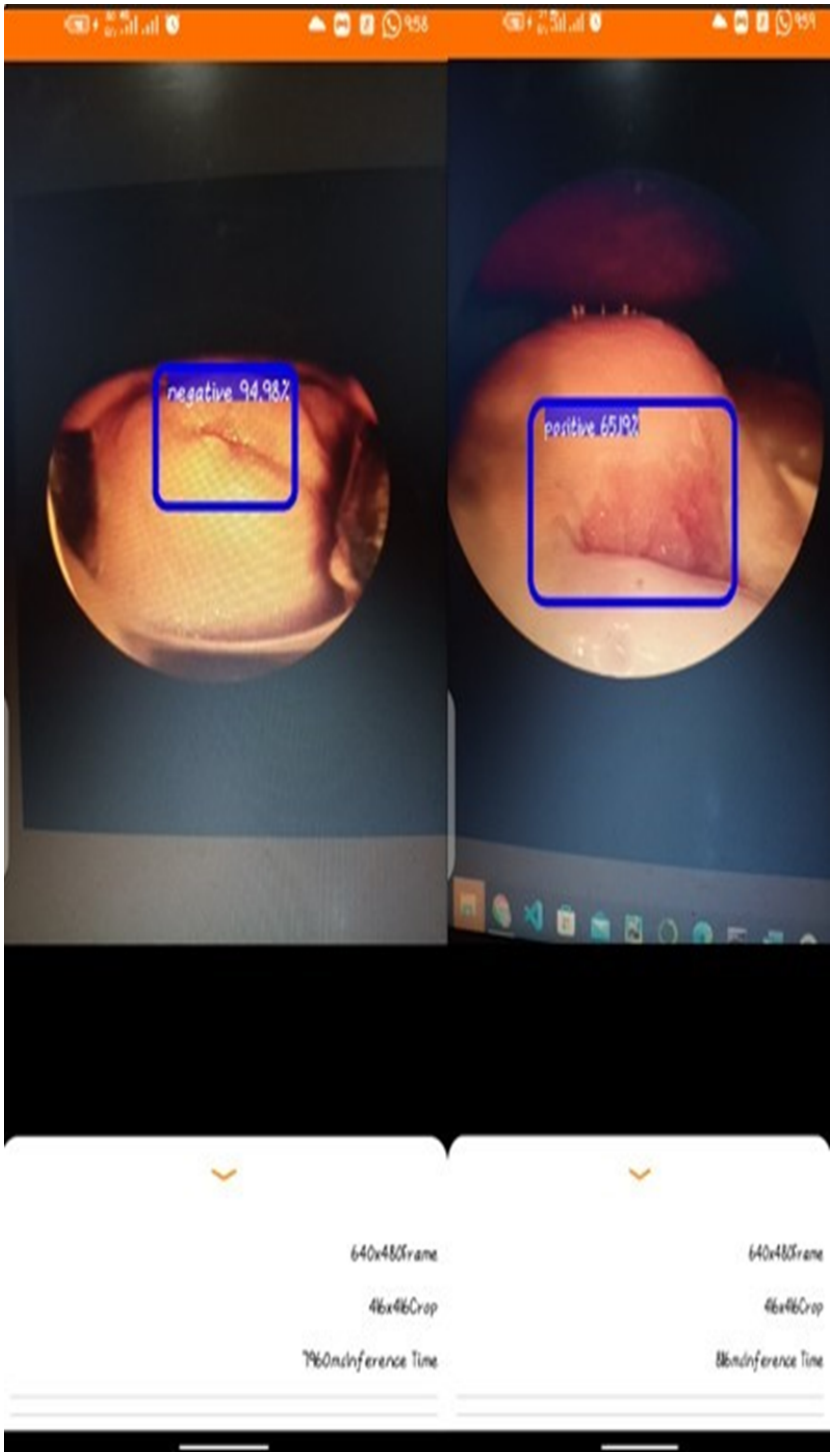

Figure 3.4: Android application showing deployment of task two

4. Results

This chapter shows various results obtained from the training process of task one and task two.

4.1 Task One Results

In task one, the object detection models were trained that was Yolo V5 and the Yolo V4 model.

4.1.1 Yolo V5 Model Results

The Yolo V5 model results were obtained by evaluating the validation and the test sets as shown below:

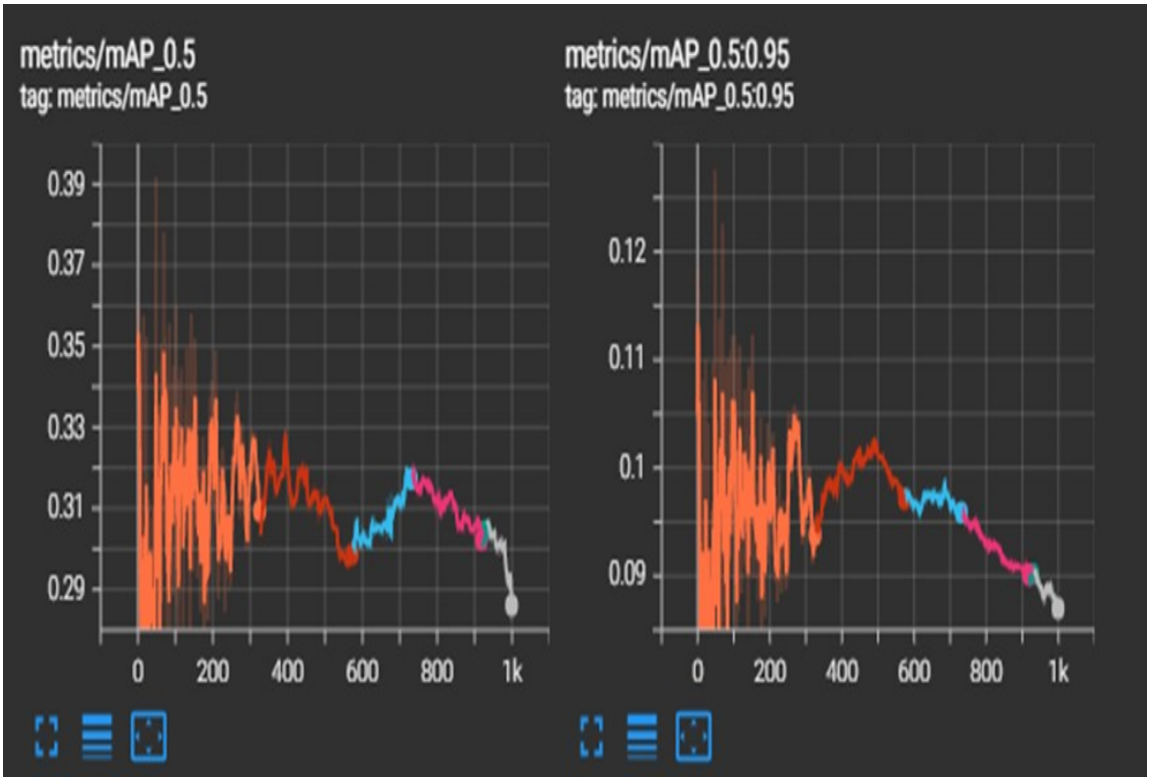

Figure 4.1: Mean Average Precision

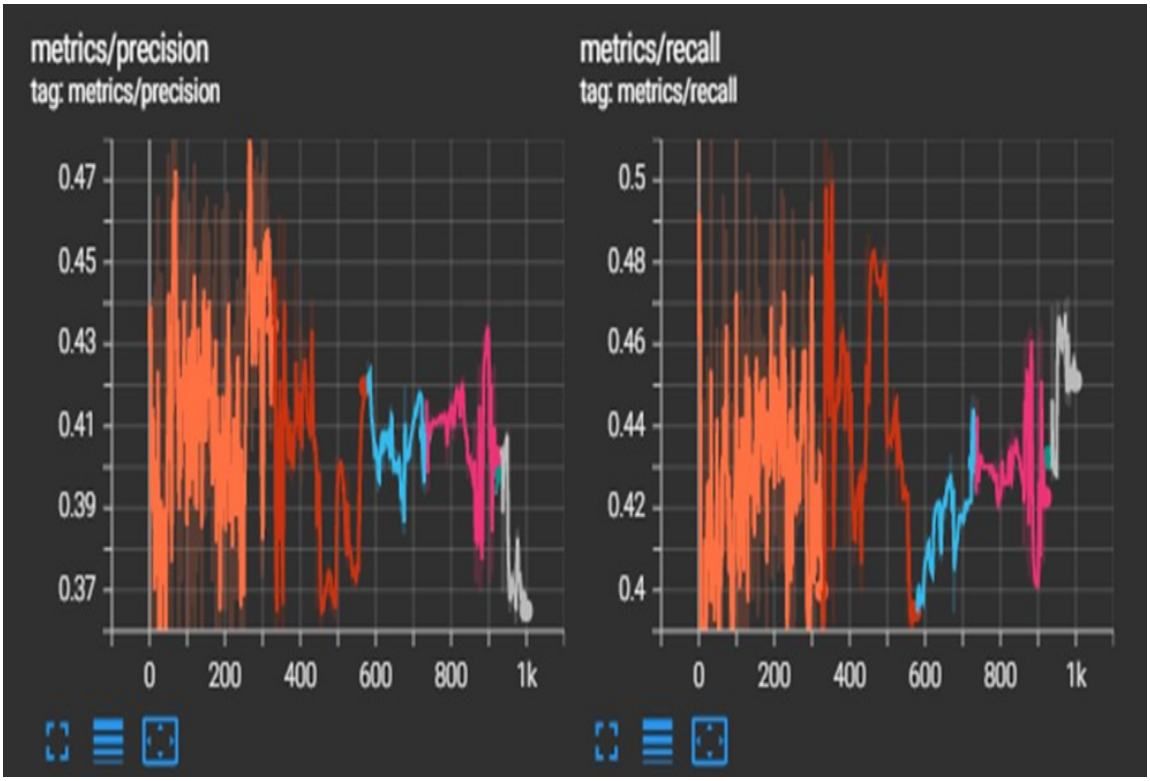

Figure 4.2: The Precision and Recall Graph

The Yolo V5 model results are summarized in the table:

Table 4.1: The Yolo V5 model results for task one

Dataset	Precision	Recall	mAP
Results	0.363	0.449	0.283

4.1.2 Yolo V4 Model Results

The Yolo V4 model results were obtained by evaluating the validation and the test sets as shown below:

```
 detections_count = 541, unique_truth_count = 265
class_id = 0, name = Type_1, ap = 80.43%         (TP = 50, FP = 18)
class_id = 1, name = Type_2, ap = 71.65%         (TP = 97, FP = 22)
class_id = 2, name = Type_3, ap = 41.86%         (TP = 40, FP = 44)

 for conf_thresh = 0.25, precision = 0.69, recall = 0.71, F1-score = 0.70
 for conf_thresh = 0.25, TP = 187, FP = 84, FN = 78, average IoU = 48.56 %

 IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
 mean average precision (mAP@0.50) = 0.646473, or 64.65 %
Total Detection Time: 22 Seconds

Set -points flag:
 `-points 101` for MS COCO
 `-points 11` for PascalVOC 2007 (uncomment `difficult` in voc.data)
 `-points 0` (AUC) for ImageNet, PascalVOC 2010-2012, your custom dataset

 mean_average_precision (mAP@0.5) = 0.646473
```

Figure 4.3: Yolo V4 Results for Task One

The Yolo V4 model results are summarized in the table:

Table 4.2: Yolo V4 model results for task one

Parameters	Precision	Recall	mAP	F1-Score	TP	FP	FN
Results	0.69	0.71	0.646	0.70	187	84	78

Where;
TP - True Positive
FP - False Positive
FN - False Negative

4.2 Task Two Results

In task two, the classification and the object detection models were trained. Under the classification model, we used the inception v3 and the xception model whereas under the object detection model we used the Yolo V4 model.

4.2.1 Classification Model Results

4.2.1.1 Xception Model Results

This classification model had the best better performance with a training accuracy of 97.13%, validation accuracy of 89.01% and test accuracy of 91.3%

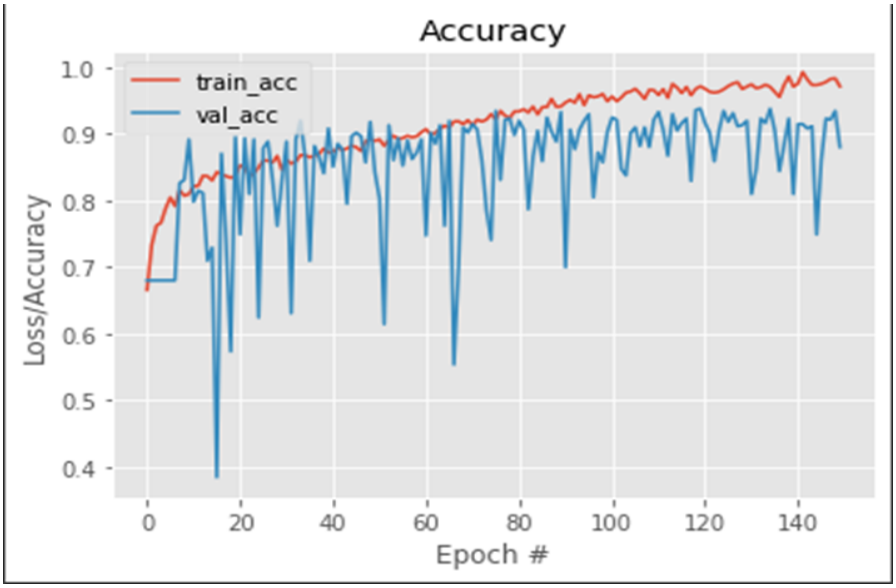

Figure 4.4: Training and validation accuracy for the Xception.

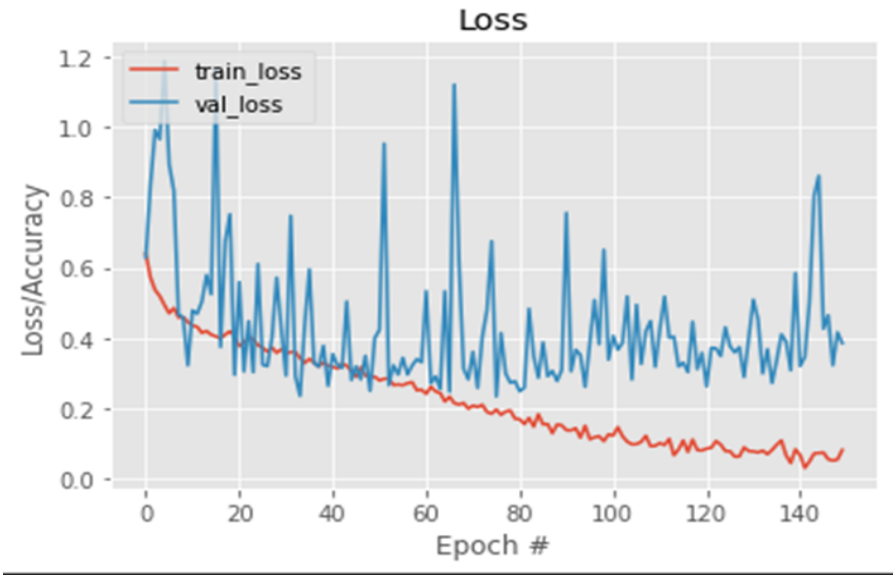

Figure 4.5: Training and validation loss for the Xception Model.

4.2.1.2 The Inception V3 Model Results

This model gave a training accuracy of 88.21%, validation accuracy of 78.1% and testing accuracy of 75.9%.

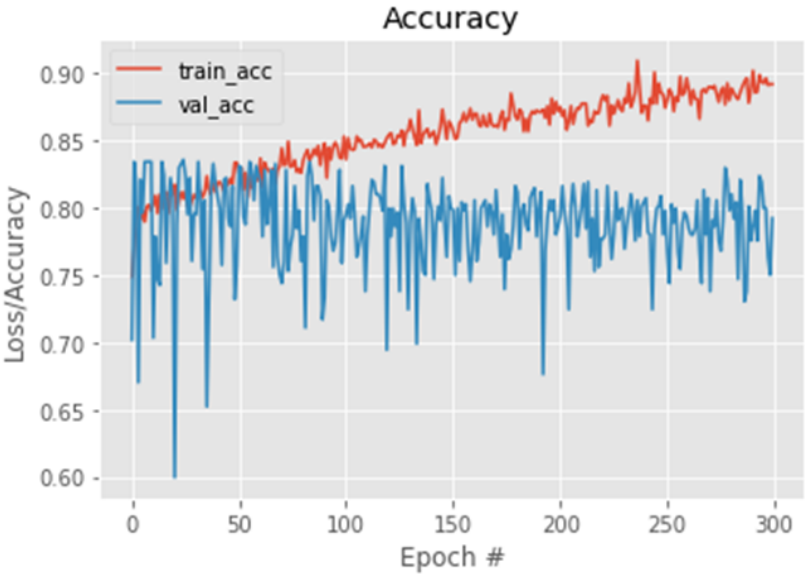

Figure 4.6: Training and validation accuracy for the inception V3 Model.

Figure 4.7: Training and validation loss for the inception V3 Model

4.2.2 The Object Detection Model Results

In task two, the object detection model was trained using the Yolo V4 model. The results are shown in the table.

Table 4.3: Yolo V4 results for task two

Parameters	Precision	Recall	mAP	F1-Score	TP	FP	FN
Results	0.88	0.89	0.879424	0.88	472	65	60

Where;

TP - True Positive
FP - False Positive
FN - False Negative

```
 calculation mAP (mean average precision)...
532
 detections_count = 687, unique_truth_count = 532
class_id = 0, name = negative, ap = 80.76%       (TP = 241, FP = 51)
class_id = 1, name = positive, ap = 95.13%       (TP = 231, FP = 14)

 for conf_thresh = 0.25, precision = 0.88, recall = 0.89, F1-score = 0.88
 for conf_thresh = 0.25, TP = 472, FP = 65, FN = 60, average IoU = 67.18 %

 IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
 mean average precision (mAP@0.50) = 0.879424, or 87.94 %
Total Detection Time: 45 Seconds

Set -points flag:
 `-points 101` for MS COCO
 `-points 11` for PascalVOC 2007 (uncomment `difficult` in voc.data)
 `-points 0` (AUC) for ImageNet, PascalVOC 2010-2012, your custom dataset

 mean_average_precision (mAP@0.5) = 0.879424
```

Figure 4.8: Yolov4 results trained on dataset of task two.

4.3 Discussion of results

In task one we only trained the object detection model using Yolo V5 and Yolo V4. The Yolo V4 provided better results with a mAP of 0.646 the compared to the Yolo V5 which gave mAP of 0.283. In task two we trained the classification model and the object detection model. Under the classification model the xception model had a better performance with a training accuracy of 97.13%, validation accuracy of 89.01% and test accuracy of 91.3% compared to the Inception V3 model which gave a training accuracy of 88.21%, validation accuracy of 78.1% and testing accuracy of 75.9%. under the object detection model, we only trained using the Yolo V4 which gave mAP of 0.879424.

5. Conclusion and Recommendations

Convolutional neural networks (CNNs) have accomplished astonishing achievements across a variety of domains, including medical research, and an increasing interest has emerged in oncology. Deep learning has become a dominant method in a variety of complex tasks such as image classification and object detection. In this study, we implemented semi-supervised learning to achieve a fully labelled dataset and generative model specifically Variational Auto Encoders to generate more training data and classify the dataset.

5.2 Challenges

The heavy class imbalances in the dataset affected the model performance in a way for instance the positive class having 383 images and the negative class having 1886 images.
We also faced a challenge of GPU run time disconnections due to usage of free google GPU which is time limited. This has made graphical visualization of yolov4 results tricky.

5.3 Recommendations

More data should be collected more especially for the positive class so as to reduce on the class imbalance. Data generative models may also be used.
Provision of a free GPU so as to smoothen the training process as well as to reduce on the time spent training a single model.

References

[1] B. F. S. R. F. J. L. J. J. A. Torre LA, " Global cancer statistics, 2012. CA Cancer J Clin. 2015;65(2):87–108.".

[2] S. I. D. R. E. S. M. C. R. M. P. D. F. D. B. F. C. i. a. m. w. s. m. a. m. p. i. G. 2. I. J. C. 2. Ferlay J and 136(5):E359–86., " Cancer incidence and mortality worldwide: sources, methods and major patterns in GLOBOCAN 2012. Int J Cancer. 2015;136(5):E359–86.".

[3] W. C. P. Stewart Bernard W, "World Cancer Report 2014. Lyon: International Agency for Research on Cancer/World Health Organization; 2014.".

[4] B. A. C. P. W. J. B. J. D. I. A. I. Bingham A, "Factors affecting utilization of cervical cancer prevention services in low-resource settings. Salud Publica Mex. 2003;45 Suppl 3:S408–16.".

[5] B.-R. L. S. B. B. M. C. R. M. J. B. F. d. S. S. C. X. Bruni L, "Human papillomavirus and related diseases in Uganda. In: Summary report 2014. HPV Information Centre; 2014.," [Online]. Available: http://www.hpvcentre.net/statistics/reports/UGA.pdf..

[6] L. Z. C. H. W. a. J. F. C. N. Dong, "Inception v3 based cervical cell classification combined with artificially extracted features Applied Soft Computing, vol. 93, p. 106311, 2020".

[7] T. X. X. J. e. a. W. Hua, " "Lymph-vascular space invasion prediction in cervical cancer: exploring radiomics and deep learning multilevel features of tumor and peritumor tissue on multiparametric MRI," Biomedical Signal Processing and Control.

[8] K. K. A. S. e. a. .. T. I. Yusufaly, " "A knowledge-based organ dose prediction tool for brachytherapy treatment planning of patients with cervical cancer," Brachytherapy, vol. 19, no. 5, pp. 624–634, 2020".

[9] Z. Z. H. L. e. a. J. Shao, " "DCE-MRI pharmacokinetic parameter maps for cervical carcinoma prediction," Computers in Biology and Medicine, vol. 118, article 103634, 2020".

[10] G. M. a. M. S. H. .. A. Ghoneim, " "Cervical cancer classification using convolutional neural networks and extreme learning machines," Future Generation Computer Systems, vol. 102, pp. 643–649, 2020".

[11] C. K. P. A. K. D. G. R. R. a. A. G. .. V. Chandran, " "State of charge estimation of lithium - ion battery for electric vehicles using machine learning algorithms," World Electric Vehicle Journal, vol. 12, no. 1, p. 38, 2021".

[12] V. C. a. M. G. S. ". M. Suriya, "Enhanced deep convolutional neural network for malarial parasite classification," International Journal of Computers and Applications, pp. 1–10, 2019".

[13] M. H. H. U. e. a. A. Alghamdi, " "Detection of myocardial infarction based on novel deep transfer learning methods for urban healthcare in smart cities," Multimedia Tools and Applications, 2020.".

[14] B. M. C. Staff, "Mayo Clinic," [Online]. Available: https://www.mayoclinic.org/diseases-conditions/cervical-cancer/symptoms-causes/syc-20352501.

[15] S. R. a. P. Norvig, "Artficial Intelligence A modern Approach Third Edition," Artficial Intelligence A modern Approach Third Edition, New Jersey: Pearson Education, 2010. .

[16] IBM, "What is Computer Vision," [Online]. Available: https://www.ibm.com/topics/computer-vision.

[17] S. Insights, "SAS Insights," [Online]. Available: https://www.sas.com/en$_u$s/*insights/analytics/comp*
vision.html : : *text* = *Computer*

[18] J. Le, ""The 5 Computer Vision Techniques That Will Change How You See The World," Heartbeat, 2018 April 2018. [Online]. Available: https://heartbeat.fritz.ai/the-5-computer-vision-techniques-that-will-change-how-you-see-the-world-1ee19
.

[19] A. C. a. P. Gulia, ""Machine Learning and Deep Learning," International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8 , no. 12, pp. 1-5, 2019.," . .

[20] M. Swamynathan, " Master Machine Learning with Python in Six Steps, Bangalore, Karnataka, India: Apress, 2017.".

[21] V. Kakkaraparthi, ""Activation Functions in Neural Networks,"," "Activation Functions in Neural Networks," Medium, 9 February 2019 [Online]. Available: https://prateekvishnu.medium.com/activation-functions-in-neural-networksbf5c542d5fec: :text=Disadvan

[22] IBM, "Convolutional Neural Networks," [Online]. Available: https://www.ibm.com/cloud/learn/convolutional-neural-networkstoc-how-do-con–z4UwR2M.

[23] KDnuggets, ""Optimization Algorithms in Neural Networks," KDnuggets, December 2020.," [Online]. Available: https://www.kdnuggets.com/2020/12/optimization-algorithms-neuralnetworks.html:

[24] J. Leonel, ""Hyperparameters in Machine /Deep Learning," [Online].

[25] R. J. Tan, "Towards Data Science," [Online]. Available: https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52.

[26] .. Q. Meng, " "Machine learning to predict local recurrence and distant metastasis of cervical cancer after definitive radiotherapy," International Journal of Radiation Oncology • Biology • Physics, vol. 108, no. 3, article e767,," 2020.

[27] R. J. X. C. e. a. J. Shan, ” “Machine learning predicts lymph node metastasis in early-stage oral tongue squamous cell carcinoma,” Journal of Oral and Maxillofacial Surgery, vol. 78, no. 12, pp. 2208–2218, 2020.”.

[28] V. B. R. K. a. M. J. S. K. Saini, ” “ColpoNet for automated cervical cancer screening using colposcopy images,” Machine Vision and Applications,” vol. 31, 2020.

[29] P. G. a. S. B. P. Sanyal, ” “Performance characteristics of an artificial intelligence based on convolutional neural network for screening conventional Papanicolaou-stained cervical smears,” Medical Journal, Armed Forces India, vol. 76, no. 4, pp”.

[30] A. J. a. F. K. B. R. Jany, ” “Automatic microscopic image analysis by moving window local Fourier transform and machine learning,” Micron, vol. 130, article 102800, 2020”.

[31] V. N. S. M. M. J. e. a. V. Karunakaran, ” “Diagnostic spectro-cytology revealing differential recognition of cervical cancer lesions by label-free surface enhanced Raman fingerprints and chemometrics,” Biologie et Médecine,” vol. 29, p. 102276, 2020.

[32] S. B. Z. a. T. C. C. Z. F. A. D. Jia, ” “CNN-SVM network abstract,” Neurocomputing, 2020.”.

[33] B. Z. L. a. C. C. Z. .. A. Dongyao Jia, ” “Detection of cervical cancer cells based on strong feature CNN-SVM network,” Neurocomputing, vol. 411, pp. 112–127, 2020”.

[34] A. Z. a. X. W. “. u. n. l. P. r. J. Ren, ”” Pharmacological Research, no. article 104743, 2020.”.

[35] M. A. a. Y. S. “. M. F. Ijaz, ”Data-driven cervical cancer prediction model with outlier detection and over-sampling methods,” Sensors, vol. 20, no. 10, pp. 2809–2822, 2020”.

[36] T. N. M. Z. e. a. .. E. M. L. Ruiz, ” “A novel gene panel for prediction of lymph-node metastasis and recurrence in patients with thyroid cancer,” Surgery, vol. 167, no. 1, pp. 73–79, 2020”.

[37] L. F. T. K. K. L. e. a. .. D. Stelzle, ” “Estimates of the global burden of cervical cancer associated with HIV,” The Lancet Global Health, vol. 9, no. 2, pp. e161–e169, 2021”.

[38] G. Z. S. K. e. a. D. Goksuluk, ”MLSeq: machine learning interface for RNA-sequencing data,” Computer Methods and Programs in Biomedicine, vol. 175, pp. 223–231, 2019.”.

[39] S. K. a. O. C. .. K. Adem, ” “Classification and diagnosis of cervical cancer with stacked autoencoder and softmax classification,” Expert Systems with Applications, vol. 115, pp. 557–564, 2019”.

[40] P. R. B. S. e. a. A. Goncalves, ” “Bayesian multitask learning regression for heterogeneous patient cohorts,” Journal of Biomedical Informatics: X, vol. 4, article 100059, 2019.”.

[41] B. Deepa and M. G. Sumithra, ” “An intensity factorized thresholding based segmentation technique with gradient discrete wavelet fusion for diagnosing stroke and tumor in brain MRI,” Multidimensional Systems and Signal Processing, vol. 30, no. 4, pp. 2081–2”.

[42] A. W. A. H. B.-E. a. J. O. W. William, ” “A review of image analysis and machine learning techniques for automated cervical cancer screening from pap-smear images,” Computer Methods and Programs in Biomedicine, vol. 164, pp. 15–22, 2018”.

[43] W. L. C. S. e. a. .. C. Zhang, ” “Urine proteome profiling predicts lung cancer from control cases and other tumors,” eBioMedicine, vol. 30, pp. 120–128, 2018”.

[44] S. P. A. M. e. a. K. Matsuo, ” “A pilot study in using deep learning to predict limited life expectancy in women with recurrent cervical cancer,” American Journal of Obstetrics and Gynecology, vol. 217, no. 6, pp. 703–705”.

[45] A. M. Iliyasu and C. Fatichah, ” “A quantum hybrid PSO combined with fuzzy k-NN approach to feature selection and cell classification in cervical cancer detection,” Sensors, vol. 17, no. 12, p. 2935, 2017”.

[46] V. S. a. R. G. A. Sarwar, ”Hybrid ensemble learning technique for screening of cervical cancer using Papanicolaou smear image analysis,” Personalized Medicine Universe, vol. 4, pp. 54–62, 2015.”.

[47] N. Karjane and D. Chelmow, ” “New cervical cancer screening guidelines, again,” Obstetrics and Gynecology Clinics of North America, vol. 40, no. 2, pp. 211–223, 2013”.

[48] M. A. S. L. a. R. R. “. C. Schwaiger, ”Current guidelines for cervical cancer screening,” Journal of the American Academy of Nurse Practitioners, vol. 24, no. 7, pp. 417–424, 2012”.

[49] Y. M. L. P. L. e. a. T. Zhang, ” “Cervical precancerous lesions classification using pre-trained densely connected convolutional networks with colposcopy images,” Biomedical Signal Processing and Control, vol. 55, p. 101566, 2020”.

[50] E. S. A. G. a. M. A. .. J. Lu, ” “Machine learning for assisting cervical cancer diagnosis: an ensemble approach,” Future Generation Computer Systems, vol. 106, pp. 199–205, 2020”.

[51] H. Z. X. H. S. Z. a. D. N. M. T. Xu, ” “Multimodal deep learning for cervical dysplasia diagnosis,” in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016, Lecture Notes in Computer Science, pp. 115–123, Springer, Cham, 2016.”