Retina Diseases Diagnosis Using Deep Learning

Abeer Abed ElKareem Fawzi Elsharif and Samy S. Abu Naser

Deprtment of Information Technology,
Faculty of Engineering and Information Technology,
Al-Azhar University, Gaza, Palestine

***Abstract**: There are many eye diseases but the most two common retinal diseases are Age-Related Macular Degeneration (AMD), which the sharp, central vision and a leading cause of vision loss among people age 50 and older, there are two types of AMD are wet AMD and DRUSEN. Diabetic Macular Edema (DME), which is a complication of diabetes caused by fluid accumulation in the macula that can affect the fovea. If it is left untreated it may cause vision loss. Therefore, early detection of diseases is a critical importance. Our main goal is to help doctors detect these diseases quickly before reaching a late stage of the disease. In ophthalmology, optical coherence tomography (OCT) is critical for evaluating retinal conditions. OCT is an imaging technique used to capture high-resolution cross-sections of the retinas of patient. In this thesis, we review ways and techniques to use deep learning classification of the optical coherence tomography images of diseases from which a Retinal is suffering .The models used to improve patient care are (VGG-16, MobileNet, ResNet-50, Inception V3, and Xception) to reduce costs and allow fast and reliable analysis in large studies. The obtained results are encouraging, since the best model ResNet-50 reaching 96.21% of testing accuracy, which is very useful for doctors, to diagnose retinal diseases.*

Keywords: Deep Learning, Optical Coherence Tomography, Age-Related Macular Degeneration, Diabetic Macular Edema.

1.1 Introduction

This section presents the basic information about artificial intelligence, machine learning, deep learning, retinal, and the research objective. Section 1.2 briefly explains the definition and the concept of artificial intelligence. Section 1.3 briefly explains the definition and the concept of machine learning. Section 1.4 briefly explains deep learning and its applications. Section 1.5 gives explanations of retinal diseases. Section 1.6 explains the problem statement. Section 1.7 states the research objectives, and section.

1.2 Artificial intelligence

Artificial intelligence is relating to more than one branch of knowledge, which belongs to the interdisciplinary subject of natural science, technical science, and social science. Its main task is to establish intelligent information processing theory, and then design a computing system that can show certain behaviors similar to human intelligence.

Intelligence refers to the use of human intelligence-related thinking ability to study an intelligence system, logical thinking ability, mainly including image thinking ability, and creative thinking ability. Artificial intelligence (AI) capability is a kind of intelligent behavior that is usually related to human intelligence, such as judgment, reasoning, proof, recognition, perception, understanding, communication, design, thinking, learning and problem-solving[1].

1.3 Machine Learning

Machine learning is a method of data analysis that automates analytical model building. As shown in Fig.1, it is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns, and make decisions with minimal human intervention.

Over the last decade, machine learning had been applied in different applications such as; self-driving cars, recognizing speech accurately enough, effective web search, and how to reliably discriminate between the fruits by being trained on a large amount of data[2].

The process of learning begins with observations of data, such as examples, direct experience, or instruction, to look for patterns in data and make better decisions in the future. The primary aim is to allow computers to learn automatically without human intervention or assistance and adjust actions accordingly [2].

Machine learning is often categorized into several algorithms as supervised, unsupervised, Semi-supervised, and Reinforcement. Supervised machine learning algorithms can apply what has been learned in the past to new data using labeled examples such as determining customer lifetime value, forecasting sales, and forecasting supply and demand to predict future events.

In contrast, unsupervised machine learning algorithms are used when the information used to train is neither classified nor labeled [3].

Semi-supervised machine learning algorithms fall somewhere in between supervised and unsupervised learning since they use both labeled and unlabeled data for training typically a small amount of labeled data and a large amount of unlabeled data.

Reinforcement machine learning algorithms are a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance [3].

Machine learning enables the analysis of massive quantities of data. While it generally delivers faster, more accurate results in order to identify profitable opportunities or dangerous risks, it may also require additional time and resources to train it properly. Combining machine learning with AI and cognitive technologies can make it even more effective in processing large volumes of information [3].

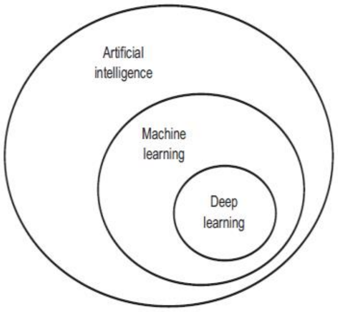


Fig 1: Deep learning exists within Machine Learning, which exists within the larger category of Artificial Intelligence. [3]

1.4 Deep Learning

As shown in Fig.1, deep learning is a subset of machine learning techniques that teach computers to do what comes naturally to humans: learn by example. Deep is a technical term, it refers to the number of layers in a neural network, as shown in Fig.2. A shallow network has one so-called hidden layer, and a deep network has more than one. Multiple hidden layers allow deep neural networks to learn features of the data in a so-called feature hierarchy, because simple features (e.g. two pixels) recombine from one layer to the next, to form more complex features (e.g. a line). Nets with many layers pass input data (features) through more mathematical operations than nets with few layers, and are therefore more computationally intensive to train. Computational intensity is one of the hallmarks of deep learning, and it is one reason why a new kind of chip called Graphics Processing Unit (GPUs) are in demand to train deep-learning models.

Deep learning allows machines to solve complex problems even when using a data set that is very diverse, unstructured, and inter-connected such as Images & Videos recognition, image analysis & classification, media recreation, recommendation systems, natural language processing, etc[3].

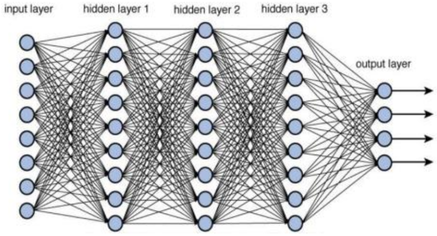


Fig 2: Deep Learning with Multiple Layers [4]

1.5 Retina

The retina is a thin layer of tissue that lines the back of the eye on the inside. It is located near the optic nerve, as shown in Fig.3. The purpose of the retina is to receive light that the lens has focused, convert the light into neural signals, and send these signals on to the brain for visual recognition [5].

The retina processes light through a layer of photoreceptor cells. These are essentially light-sensitive cells, responsible for detecting qualities such as color and light intensity. The retina processes the information gathered by the photoreceptor cells and sends this information to the brain via the optic nerve. Basically, the retina processes a picture from the focused light, and the brain is left to decide what the picture is [5].

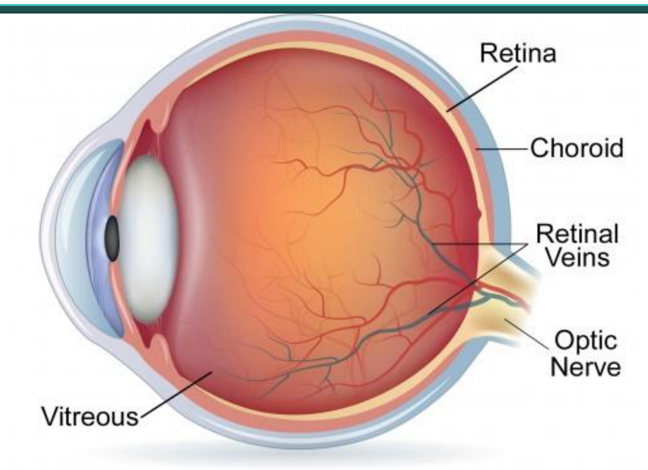


Fig 3: Schematic of the Eye [6]

In the human eye, the retina is nourished by its own blood vessels that feed the inner layers, and by the blood vessels of the choroid, that feed the outer layers of the retina. The inner blood vessels are the ones that are afflicted by diabetic retinopathy (DME). Vascular occlusions, both vein, and artery occlusion can lead to loss of normal blood supply to the retina and lead to blindness.

The choroidal blood vessels are the source of fluid and hemorrhage in wet age-related macular degeneration (AMD) also called (CNV) [6], as shown in Fig.4.

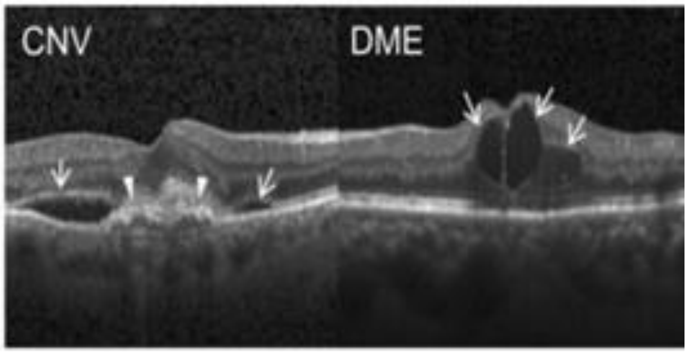


Fig 4: Intra retinal Fluid for CNV and DME

1.6 Problem Statement

The role of the retina is to turn the light rays reflected through the lens of the eye into signals that are sent to the brain to interpret them as a visible image.

In the examination of the eye, if there is a presence of blood, fluid or fat deposition, under the retina or under the retinal epithelium beneath it, the presence of abnormal blood vessels should be observed, also Choroidal neovascularization (CNV) is one of the leading causes of blindness in developed countries, especially age-related macular degeneration (AMD), and Drusen disease is a special case of (AMD). DME is a complication of Diabetic Retinopathy, an eye condition that occurs when too much blood sugar damages the blood vessels that supply the retina. So that the presence of these fluids cannot be determined under or between the retina by ophthalmologists.

OCT tests are a great way to capture accurate retinal images for the diagnosis of the retina for detecting eye disease, but the interpretation of these data, analysis, and interpretation of OCT images takes up a significant amount of time. OCT images are challenging because of high noise and motion in images especially in images with severe pathology. Furthermore, image capturing procedures such as resolution, filtering, scan patterns different between Optometrists.

Deep Learning can help doctors make faster, more accurate diagnoses. It can predict the risk of a disease in time to prevent it. Deep learning helps researchers analyze medical data to treat diseases. However, analyzing medical images can often be a difficult and time-consuming process.

In this research, we implement a deep learning model to detect early deformities within the retina through a collection of images 84,495 images (JPEG) and 4 categories (NORMAL, CNV, DME, DRUSEN), and these OCT images (Spectralis OCT, Heidelberg Engineering, Germany) were selected from retrospective cohorts of adult patients from the Shiley Eye Institute of the University of California San Diego, the California Retinal Research Foundation, Medical Center Ophthalmology Associates, the Shanghai First People's Hospital, and Beijing Tongren Eye Center between July 1, 2013 and March 1, 2017.
If we can diagnose and treat eye conditions early, it gives us a great opportunity to maintain the vision of people before losing.

1.7 Objectives

The objectives of this study are:

- Implementation of deep learning model used to detect four types of Retinal cases (NORMAL, CNV, DME, and DRUSEN).
- Avoid diagnostic errors that occur due to the manual observation of images.
- It enhances doctors' ability to analyze medical images.
- It helps Ophthalmologists, doctors make faster, and more accurate diagnoses.

2.1 Retinal Diseases

In age-related macular degeneration (AMD), there is excessive growth of abnormal blood vessels in the eye (choroidal neovascularization), eventually leading to vision loss due to detachment of retinal pigmented epithelium.
AMD is the principal cause of permanent blindness between the elderly over 50 years in industrialized countries and adversely affecting the quality of life [7]. AMD is divided into two major categories: non-neovascular (dry), also called (Drusen) and neovascular (wet), also called (CNV), as shown in Fig.5. Drusen is the feature of early and intermediate dry AMD.
AMD in terms of stages: early, intermediate, and late (advanced). The disease starts in the dry form and remains dry through the early and intermediate stages. Late stages can be either advanced dry or wet, both of them can cause significant vision loss. However, the progression of dry AMD is often insidious, whereas wet AMD can present with sudden central vision loss.

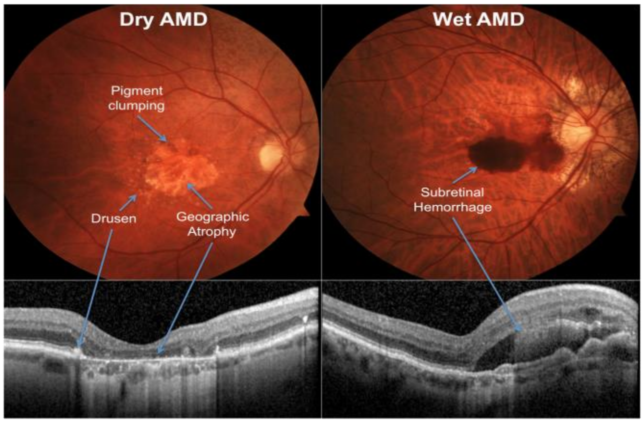


Fig 5: Dry AMD, & Wet AMD [11]

Retinal neovascularization and choroidal neovascularization (CNV), are the most common vascular diseases of the eye, as shown in Fig.6. As a result of, Retinal diseases, such as diabetic retinopathy, and age-related macular degeneration (AMD) are primarily caused by chronic or severe retinal ischemia [8].
Among AMD cases with acute visual impairment, wet AMD is responsible for approximately 90% of cases. Despite the rising prevalence of this debilitating condition, current treatment strategies for wet AMD mostly revolve around inhibitors of vascular endothelial growth factor, and photodynamic therapy. Both of them have considerable limitations such as lack of long-term improvement in visual acuity [10, 11].

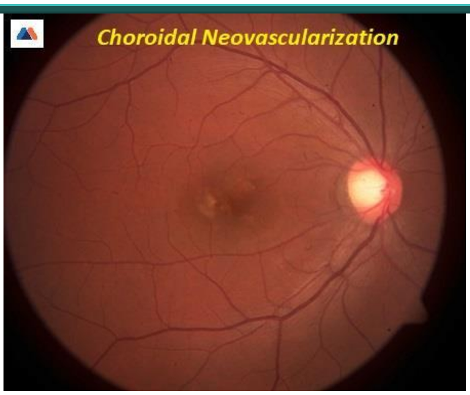


Fig 6: Choroidal Neovascularization. [9]

Diabetic Macular Edema (DME) is an accumulation of fluid in the macula part of the retina that controls our most detailed vision abilities due to leaking blood vessels, as shown in Fig.7. Diabetic retinopathy is a disease that damages the blood vessels in the retina, resulting in vision impairment. Left untreated, these blood vessels begin to build up pressure in the eye and leak fluid, causing DME [12].

Macular edema can also occur after eye surgery, in association with age-related macular degeneration, or as a consequence of inflammatory diseases that affect the eye. Any disease that damages blood vessels in the retina can cause macular edema [13].

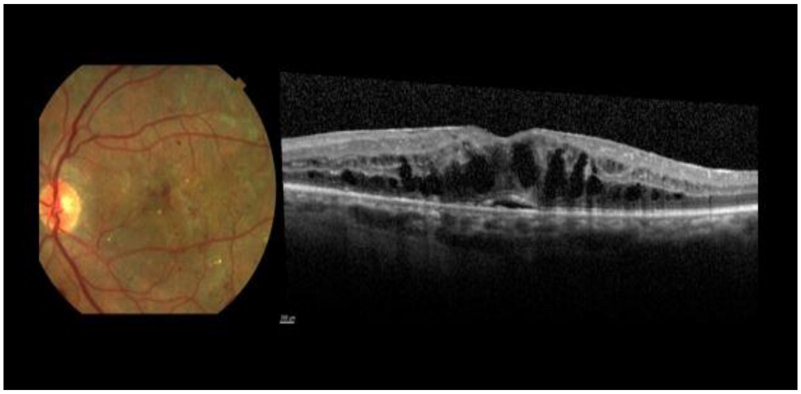


Fig 7: Diabetic Macular Edema [14]

2.2 Optical Coherence Tomography (OCT)

OCT plays an important role in disease detection, prognostication, and control many eye conditions, including (a macular hole, macular pucker, macular edema, age-related macular degeneration, glaucoma, central serous retinopathy, diabetic retinopathy, and vitreous traction). OCT relies on light waves. It cannot be used with conditions that interfere with light passing through the eye. These conditions include dense cataracts or significant bleeding in the vitreous. With OCT, your ophthalmologist can see each of the retina’s distinctive layers, as shown in Fig.8. This allows your ophthalmologist to map and measure their thickness, through scan eye without touching it. Scanning takes about 5 -10 minutes. These measurements help with diagnosis. They also provide treatment guidance for glaucoma and diseases of the retina [15, 16].

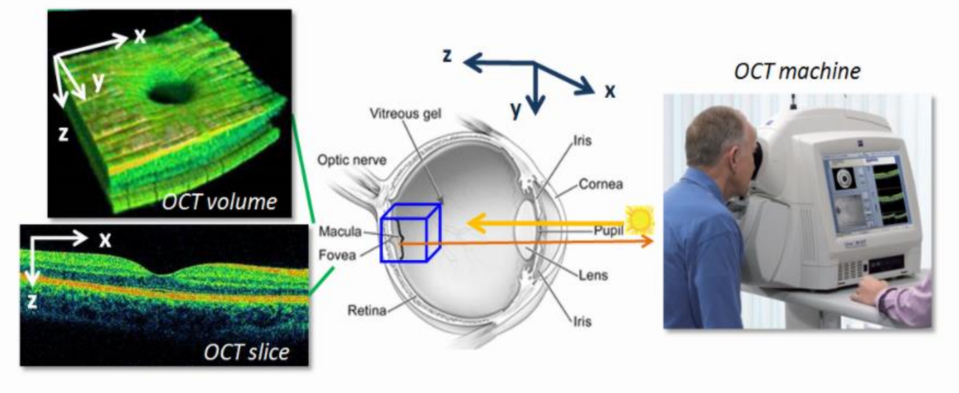


Fig 8: OCT Machine. [17]

2.3 Convolutional Neural Network

Convolutional neural networks are neural networks used to analyze, and classify data. For example, these data were images that are used to identify faces, signs, and tumors.
Convolutional neural networks (CNN) utilize layers involving filters, as shown in Fig.9 that are invented for computer vision, the results were effective for instance, Yih et al. have achieved great results in semantic parsing, Shen et al. have proposed sentence modeling, & Collobert et al. used traditional Neuro-linguistic programming (NLP) [18-20]. NLP is a psychological approach that involves analyzing strategies used by successful individuals and applying them to reach a personal goal. It relates thoughts, language, and patterns of behavior learned through experience to specific outcomes [21].

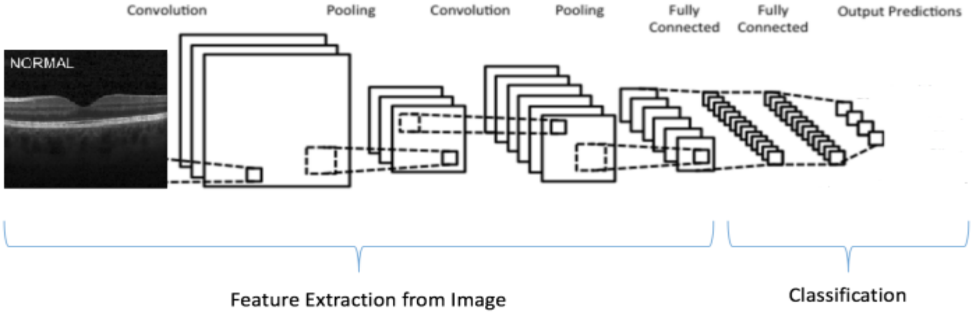


Fig 9: Convolutional Neural Networks Layers[22]

2.3.1 Convolutional Layer

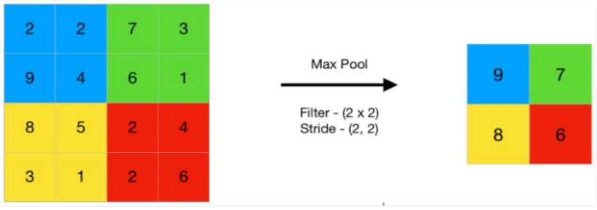
The main aim of a convolutional layer is to detect features or visual features in images such as edges, lines, color drops, etc. This is a very important property because, once it has learned a characteristic at a specific point in the image, it can recognize it later in any part of it [20], as shown in Fig.9.
Convolutional layer capture a feature map to predict the class probabilities for each feature through applying a filter that scans the whole image, few pixels at a time.

A filter is just a matrix of values, called weights, which are trained to detect specific features. The filter moves over each part of the image to check if the feature it is meant to detect is present. To provide a value representing how confident it is that a specific feature is present, the filter carries out a convolution operation, which is an element-wise product and sum between two matrices [23-25].

2.3.2 Pooling Layer

The pooling or down sampling layer is responsible for reducing the special size of the activation maps, in another meaning a down sampling operation along the spatial dimensions (width, height)), as shown in Fig.10.

In general, they are used after multiple stages of other layers, as shown in Fig.10 (i.e. convolutional and non-linearity layers) in order to reduce the number of parameters and computation in the network, and hence to also control overfitting. In special, max and average pooling are special kinds of pooling where the maximum and average value is taken, respectively [26-28].
Overfitting is an error from sensitivity to small fluctuations in the training set. Overfitting can cause an algorithm to model the random noise in the training data, rather than the intended result [29-30].



2	2	7	3
9	4	6	1
8	5	2	4
3	1	2	6

9	7
8	6

Fig 10: Maximum Pooling Filter [30]

2.3.3 Rectified Linear Unit Layer

A Rectified Linear Unit is a form of activation function used commonly in deep learning models. In essence, the function returns 0 if it receives a negative input, and if it receives a positive value, the function will return back the same positive value. The function is understood as: f(x) = max (0, x), as shown in Fig.11.
The rectified linear unit, or ReLU, allows for the deep learning model to account for non-linearities and specific interaction effects [31-33].

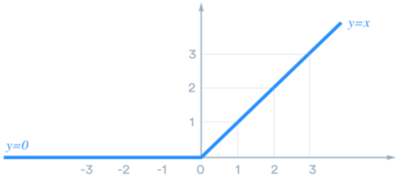


Fig 11: Relu Function [34]

How does ReLU work?

ReLU is linear for all positive values, and zero for all negative values. This means that:

1. It converges faster. Linearity means that the slope doesn't plateau, or "saturate," when x gets large. It doesn't have the vanishing gradient problem suffered by other activation functions like sigmoid or tanh [35].

2. It's sparsely activated. Since ReLU is zero for all negative inputs, it's likely for any given unit to not activate at all [36].

In machine learning, the vanishing gradient problem is encountered when training artificial neural networks with gradient-based learning methods and backpropagation. In such methods, each of the neural network's weights receives an update proportional to the partial derivative of the error function with respect to the current weight in each iteration of training. The problem is that in some cases, the gradient will be vanishingly small, effectively preventing the weight from changing its value. In the worst case, this may completely stop the neural network from further training [37].

2.3.4 Dropout Layer

The term "dropout" refers to dropping out units (both hidden and visible) in a neural network. Simply put, dropout refers to ignoring units (i.e. neurons) during the training phase of certain set of neurons which is chosen at random, as shown in Fig.12. By "ignoring", I mean these units are not considered during a particular forward or backward pass.
How to Dropout work?

1. Dropout is implemented per-layer in a neural network.

2. It can be used with most types of layers, such as dense fully connected layers, convolutional layers, and recurrent layers such as the long short-term memory network layer.

3. Dropout may be implemented on any or all hidden layers in the network as well as the visible or input layer. It is not used on the output layer.

A fully connected layer occupies most of the parameters, and hence, neurons develop co-dependency amongst each other during training which curbs the individual power of each neuron leading to over-fitting of training data, so dropout used to prevent over-fitting [38-42].

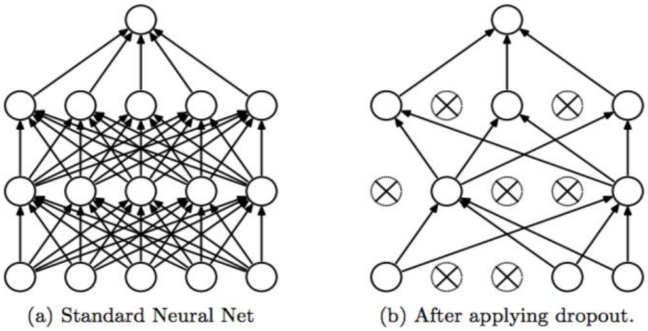

(a) Standard Neural Net
(b) After applying dropout.

Fig 12: Dropout Layer [42]

2.3.5 Batch Normalization Layer

Batch Normalization is a technique for improving the speed, performance, and stability of artificial neural networks. It is used to normalize the input layer by adjusting and scaling the activations [43-46].

Batch normalization is a normalization method that normalizes activations in a network across the mini-batch. For each feature, batch normalization computes the mean and variance of that feature in the mini-batch. It then subtracts the mean and divides the feature by its mini-batch standard deviation.

Resulting in transforming the inputs to be mean 0 and unit variance [47], as shown in Fig.13.

Input: Values of x over a mini-batch: β = {x1…..m}; parameters to be learned: γ, β

Output: $\{y_i = BN_{\gamma,\beta}\ (x_i)\}$;

$\mu\beta \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i$ // mini-batch mean

$\sigma^2\beta \leftarrow \frac{1}{m}\ \sum_{i=1}^{m}(x_i - \mu\beta\ )^2$ // mini-batch variance

$\hat{x_i} \leftarrow \frac{x_i-\mu\beta}{\sqrt{\sigma^2\beta+\epsilon}}$ // normalize

$y_i \leftarrow \gamma\ \hat{x_i} +\ \beta\ \equiv BN_{\gamma,\beta}\ (x_i)$ // scale and shift

The BN transform can be added to a network to manipulate any activation. In the notation $y_i =\ BN_{\gamma,\beta}\ (x_i)$, we indicate that the parameters γ and β are to be learned, but it should be noted that the BN transform does not independently process the activation in each training example [48-52].

Rather, $BN_{\gamma,\beta}\ (x_i)$ depends both on the training example and the other examples in the mini-batch.

The scaled and shifted values y are passed to other network layers. The normalized activations x̂ are internal to our transformation, but their presence is crucial. The distributions of values of any x̂ has the expected value of 0 and the variance of 1, as long as the elements of each mini-batch are sampled from the same distribution, and if we neglect ∈.

This can be seen by observing that $\sum_{i=1}^{m} \hat{x}_i = 0$ and $\frac{1}{m} \sum_{i=1}^{m} \hat{x}_i^{\,2} = 1$ and taking expectations. Each normalized activation $\hat{x}^{(k)}$ can be viewed as an input to a sub network composed of the linear transform $y^{(k)} \leftarrow \gamma^{(k)} \hat{x}^{(k)} + \beta^{(k)}$, followed by the other processing done by the original network. These sub-network inputs all have fixed means and variances, and although the joint distribution of these normalized $\hat{x}^{(k)}$ can change over the course of training, we expect that the introduction of normalized inputs accelerates the training of the sub-network and, consequently, the network as a whole [53-55].

2.3.6 Fully Connected Layer

In this step, the flattened feature map is passed through a neural network, as shown in Fig.9 and Fig.14. This step is made up of the input layer, the fully connected layer, and the output layer. The output layer is where we get the predicted classes. The information is passed through the network and the error of prediction is calculated. The error is then backpropagated through the system to improve the prediction [56, 60].

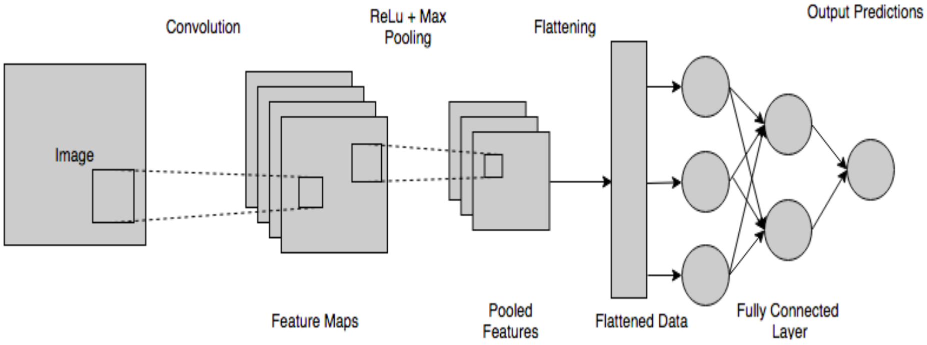


Fig 14: Fully Connected Layer [56]

2.3.7 Softmax

The Softmax regression is a form of logistic regression that normalizes an input value into a vector of values that follows a probability distribution whose total sums up to 1, as shown in Fig.15 and Table 0. The output values are between the range [0, 1] which is nice because we are able to avoid binary classification and accommodate as many classes or dimensions in our neural network model. This is why softmax is sometimes referred to as a multinomial logistic regression [57-60].

$$\sigma\,(Z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \qquad \text{, for j=1, ⋯, K.}$$

Softmax Formula Symbols Explained:

Table 1: Explained Softmax Formula Symbols [61]

$(Z)_i$	The input vector to the softmax function, made up of (z0, ... zK)
z_i	All the zi values are the elements of the input vector to the softmax function, and they can take any real value, positive, zero or negative. For example a neural network could have output a vector such as (-0.62, 8.12, 2.53), which is not a valid probability distribution, hence why the softmax would be necessary.
e^{z_i}	The standard exponential function is applied to each element of the input vector. This gives a positive value above 0, which will be very small if the input was negative, and very large if the input was large. However, it is still not fixed in the range (0, 1) which is what is required of a probability.
$\sum_{j=1}^{k} e^{z_j}$	The term on the bottom of the formula is the normalization term. It ensures that all the output values of the function will sum to 1 and each be in the range (0, 1), thus constituting a valid probability distribution.
K	The number of classes in the multi-class classifier.

2.3.8 Backpropagation

The Backpropagation algorithm looks for the minimum value of the error function in weight space using a technique called the delta rule or gradient descent. The weights that minimize the error function is then considered to be a solution to the learning problem. Backpropagation, short for "backward propagation of errors," is an algorithm for supervised learning of artificial neural networks using gradient descent. Given an artificial neural network and an error function, the method calculates the gradient of the error function with respect to the neural network's weights. It is a generalization of the delta rule for perceptrons to multilayer feedforward neural networks [62], as shown in Fig.16.

The "backwards" part of the name stems from the fact that calculation of the gradient proceeds backwards through the network, with the gradient of the final layer of weights being calculated first and the gradient of the first layer of weights being calculated last. Partial computations of the gradient from one layer are reused in the computation of the gradient for the previous layer. This backwards flow of the error information allows for efficient computation of the gradient at each layer versus the naive approach of calculating the gradient of each layer separately.

Backpropagation's popularity has experienced a recent resurgence given the widespread adoption of deep neural networks for image recognition and speech recognition. It is considered an efficient algorithm, and modern implementations take advantage of specialized GPUs to further improve performance [63].

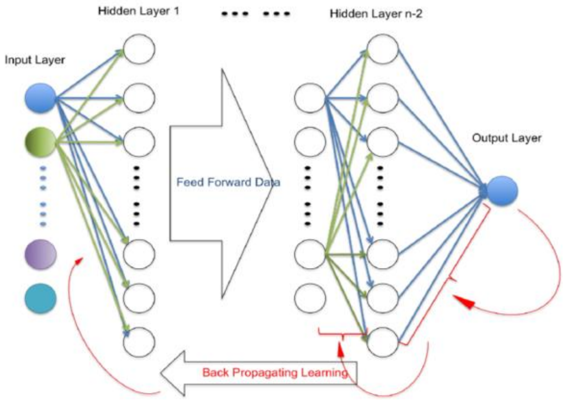


Fig 16: Backpropagation of Weights [64]

2.3.9 Adam optimization

Adam is an optimization algorithm that can be used to update network weights iteratively based on training data instead of the traditional stochastic gradient descent method. Adam is derived from the calculation of the evolutionary moment.

Adam combines the benefits of two other stochastic gradient descent extensions Adaptive Gradient Algorithm (AdaGrad), which retains a learning speed per-parameter that improves performance on sparse gradients issues (e.g., natural language issues and computer vision issues). Root Mean Square Propagation (RMSProp) which also preserves per-parameters learning rates adjusted to the weight based on the average of recent magnitudes. Offline and non-stationary problems, this algorithm does well [65].

2.4 Network Architectures

The architecture of CNN is a key factor in determining its performance and efficiency. The way in which the layers are structured, which elements are used in each layer, and how they are designed will often affect the speed and accuracy with which it can perform various tasks.

2.4.1 VGG 16

VGGNet is a simple architecture introduced in 2014 by Karen Simonyan and Andrew Zisserman, using only blocks composed of an incremental number of convolutional layers with 3x3 size filters , as shown in (Fig.17) Besides, to reduce the size of the activation maps obtained, max-pooling blocks are interspersed between the convolutional ones, reducing the size of these activation maps by half. Finally, a classification block is used, consisting of two dense layers of 4096 neurons each, and the last layer, which is the output layer, of 1000 neurons.
The 16 and 19 refer to the number of weighted layers that each network has.

The problem with VGGNet is that it consists of 138 million parameters, 34.5 times more than GoogleNet, which makes it challenging to run [66].

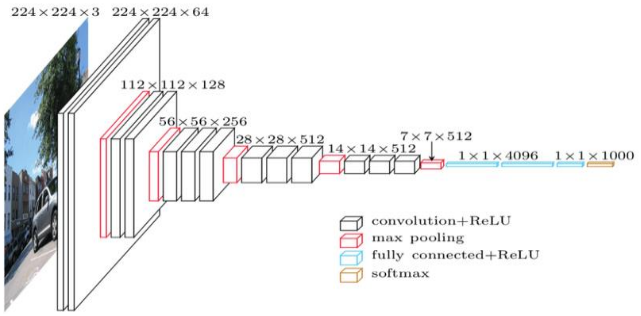


Fig 17: VGGNet Architecture [67]

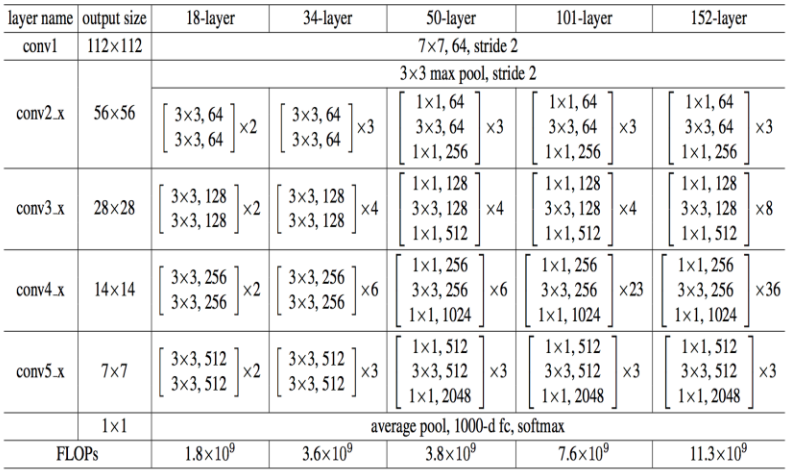
2.4.2 ResNet-50

There are variations of ResNet with a different number of layers, but the most used is ResNet50, which consists of 50 layers with weights.

ResNet is one of the early adopters of batch normalization authored by Ioffe and Szegedy was submitted to ICML in 2015.

ResNet- 50 -layer each 2-layer block is replaced in the 34-layer net with this 3-layer bottleneck block, resulting in a 50-layer ResNet (see table1).This model has 3.8 billion FLOPs [68].

Table 2: ResNet Architecture [38]

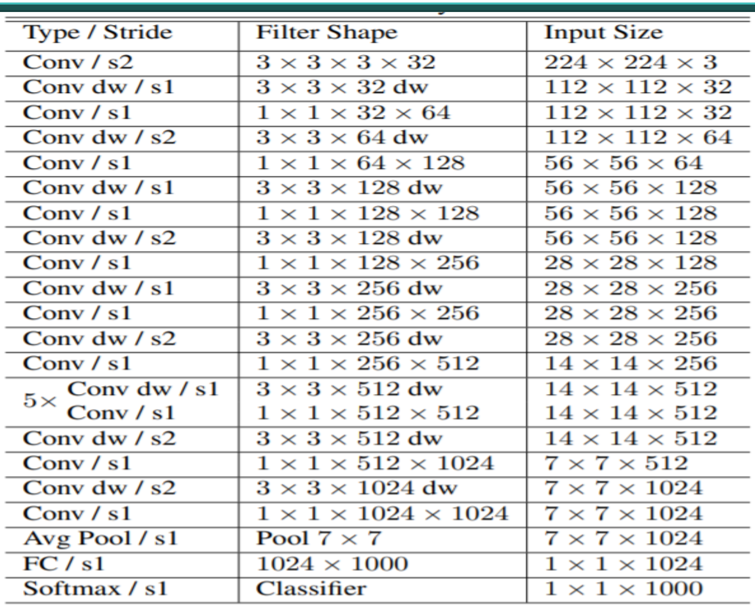


layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

2.4.3 MobileNet

Mobilenet is lightweight in its architecture was proposed as a deep learning model by Andrew G. Howard et. al of Google Research team in their research work entitled “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”. This model was proposed as a family of mobile-first computer vision models for TensorFlow, designed to effectively maximize accuracy while being mindful of the restricted resources for an on-device or embedded application. MobileNets are small, low-latency, low-power models parameterized to meet the resource constraints of a variety of use cases. They can be built upon for classification, detection, embedding, and segmentation similar to how other popular large scale models [69], (see table 2).

Table 3: MobileNet Architecture [69]



Type / Stride		Filter Shape	Input Size
Conv / s2		3 × 3 × 3 × 32	224 × 224 × 3
Conv dw / s1		3 × 3 × 32 dw	112 × 112 × 32
Conv / s1		1 × 1 × 32 × 64	112 × 112 × 32
Conv dw / s2		3 × 3 × 64 dw	112 × 112 × 64
Conv / s1		1 × 1 × 64 × 128	56 × 56 × 64
Conv dw / s1		3 × 3 × 128 dw	56 × 56 × 128
Conv / s1		1 × 1 × 128 × 128	56 × 56 × 128
Conv dw / s2		3 × 3 × 128 dw	56 × 56 × 128
Conv / s1		1 × 1 × 128 × 256	28 × 28 × 128
Conv dw / s1		3 × 3 × 256 dw	28 × 28 × 256
Conv / s1		1 × 1 × 256 × 256	28 × 28 × 256
Conv dw / s2		3 × 3 × 256 dw	28 × 28 × 256
Conv / s1		1 × 1 × 256 × 512	14 × 14 × 256
5×	Conv dw / s1	3 × 3 × 512 dw	14 × 14 × 512
	Conv / s1	1 × 1 × 512 × 512	14 × 14 × 512
Conv dw / s2		3 × 3 × 512 dw	14 × 14 × 512
Conv / s1		1 × 1 × 512 × 1024	7 × 7 × 512
Conv dw / s2		3 × 3 × 1024 dw	7 × 7 × 1024
Conv / s1		1 × 1 × 1024 × 1024	7 × 7 × 1024
Avg Pool / s1		Pool 7 × 7	7 × 7 × 1024
FC / s1		1024 × 1000	1 × 1 × 1024
Softmax / s1		Classifier	1 × 1 × 1000

2.4.4 InceptionV3

Inception v3 was proposed in the paper rethinking the Inception Architecture for Computer Vision, published in 2015, as shown in Fig.18.

In comparison to VGGNet, Inception Networks (GoogLeNet/Inception v1) have proved to be more computationally efficient, both in terms of the number of parameters generated by the network and the economic cost incurred (memory and other resources). If any changes are to be made to an Inception Network, care needs to be taken to make sure that the computational advantages aren't lost. Thus, the adaptation of an Inception network for different use cases turns out to be a problem due to the uncertainty of the new network's efficiency. In an Inception v3 model, several techniques for optimizing the network have been put suggested to loosen the constraints for easier model adaptation. The techniques include factorized convolutions, regularization, dimension reduction, and parallelized computations.

Parameters for 5 million (V1) and 23 million (V3) [70].

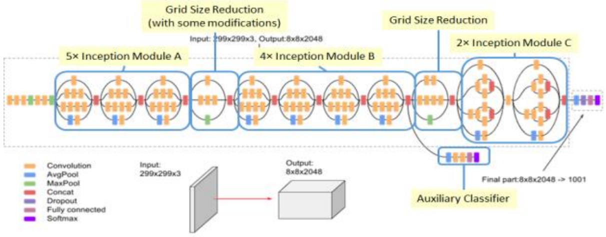


Fig 18: Inception-v3 Architecture [70]

2.4.5 Xception

Xception is an adaptation from Inception, where the Inception modules have been replaced with depthwise separable convolutions, as shown in Fig.19.

The Xception architecture has 36 convolutional layers forming the feature extraction base of the network. It has also roughly the same number of parameters as Inception-v3 (23M) [70].

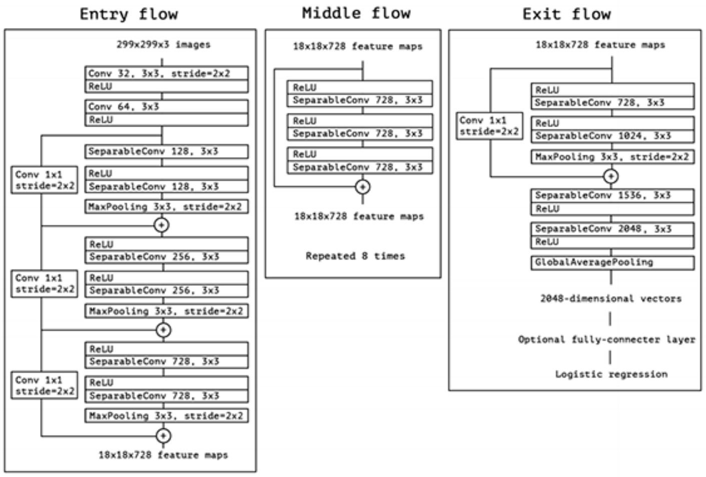


Fig 19: Xception Architecture. [70]

2.5 Data Processing

2.5.1 Laplacian as focus measure

Laplacian Operator is a derivative operator which is used to find edges in an image. The major difference between Laplacian and other operators like Prewitt, Sobel, Robinson, and Kirsch is that these all are first-order derivative masks but Laplacian is a second-order derivative mask. In this mask we have two further classifications one is Positive Laplacian Operator and the other is the Negative Laplacian Operator [71].
The Laplacian masks of 4-neighborhoods and 8- neighborhoods are given in Fig.20

0	**-1**	**0**
-1	**4**	**-1**
0	**-1**	**0**

4- Neighborhoods

-1	**-1**	**-1**
-1	**8**	**-1**
-1	**-1**	**-1**

8- Neighborhoods

Fig 20: Laplacian Masks [69]

2.5.2 Normalization

Normalization is used to scale the data before training, image data encoded as integers in the 0–255 range, encoding grayscale values.
One option is to scale the input and output variables to [0, 1] by computing cumulative distribution function using the mean and standard deviation values of each variable, independently. This improves the performance and training stability of the model [69].

2.5.3 Augmentation

Having a large dataset is critical for the performance of the deep learning model. However, we can modulate, and improve the performance of the model by augmenting the data we already have. Image data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset, and the augmentation techniques can improve the ability of the fit models to generalize what they have learned to new images.
The Keras deep learning neural network library provides the capability to fit models using image data augmentation via the ImageDataGenerator class. so some of the common augmentation techniques such as Image Horizontal and Vertical Shift Augmentation, Horizontal and Vertical Flip Augmentation, Random Rotation Augmentation, Random Brightness Augmentation, and Random Zoom Augmentation [70].

2.6 Model Evaluation

The aim of deep learning is to produce results comparable to and in some cases surpassing the human expert performance and get good predictions on data, also voids overfitting and underfitting, so we use one of model evaluation techniques for increasing efficiency, trainability, and understandability.

Underfitting usually happens when we have fewer data to build an accurate model and also when we try to build a linear model with non-linear data. In such cases, the rules of the machine learning model are too easy and flexible to be applied on such minimal data and therefore the model will probably make a lot of wrong predictions. Underfitting can be avoided by using more data and also reducing the features by feature selection [68].

2.6.1 Performance Estimation

There are some elements for measuring deep learning performance:

1. Programmability:

Deep learning is experiencing explosive growth not only in the size and complexity of the models but also in the burgeoning diversity of neural network architectures. It is difficult even for experts to understand the model choices and then choose the appropriate model to solve their AI business problems.

After a deep learning model is coded and trained, it is then optimized for a specific runtime inference environment. NVIDIA addresses training and inference challenges with two key tools. For coding, AI-based service developers use CUDA, a parallel computing platform and programming model for general computing on GPUs.

For inference, AI-based service developers use TensorRT, NVIDIA's programmable inference accelerator. As a result of that provides great performance on all kinds of neural networks [67].

2. Latency:

There is a time between taking actions and receiving requests in humans and machines, which is called latency. In practice, the latency time is often measured in milliseconds [66].

3. Accuracy:

The accuracy of the model as the number of correct predictions from all predictions made. So how the accuracy of a model might be improved [66].

4. Size of the model:

The size of the model and capacity of the network among processors have impacts on performance. There are several factors in deep learning, like number of layers, number of neurons per layer, and the complexity of computation per layer [66].

5. Rate of learning:

Deep learning models are dynamic, involving training, deployment, and retraining. Understanding how and how fast models can be trained, re-trained, and deployed as new data arrives helps to define success [67].

2.6.2.1 Accuracy

The accuracy of the model as the number of correct predictions from all predictions made. So how the accuracy of a model might be improved.

The error can take many forms, in classification problems it might be a measure of true and false positives compared to the number of false positives and false negatives. You might see these categories depicted quite intuitively in a Confusion Matrix [68].

Positives and Negatives also provide the basis for two common information retrieval measures:

$$\textbf{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\textbf{Recall} \;=\; \frac{\text{True Positive}}{\text{True Positive } + \text{ False Negative}}$$

$$\textbf{Accuracy} \;=\; \frac{\text{True Positive}}{\text{True Positive } + \text{ False Positive} + \text{ True Negative} + \text{ False Negative}}$$

Together these used to calculate a weighted average called the **F1 score**, a numerical value often quoted as a measure of classification accuracy.

$$\textbf{F1 Score} = \frac{2 * (\text{Precision } * \text{ Recall})}{\text{Precision} + \text{ Recall}}$$

Confusion Matrix:
A clean and unambiguous way to present the prediction results of a classifier is to use a confusion matrix (also called a contingency table) [48].

Table 4: Contingency Table [49]

	Predictive Values		
		Positive	**Negative**
Actual Values	**Positive**	**TP**	**FN**
	Negative	**FP**	**TN**

3.1 Previous Studies

There is enormous potential for machine learning to help with many important societal issues, especially health and medicine issues. Deep learning has an important role in disease diagnosis and treatment, health management, drug research and development, and precision medicine.

The Institute of Medicine of the National Academies of Science, Engineering, and Medicine states that "diagnostic errors contribute to about 10 percent of patient deaths", and also account for 6 to 17 percent of hospital complications. Researchers attribute the cause of diagnostic errors to a variety of different factors not only the doctor's performance such as gaps in communication between patients' family & doctor & ineffective cooperation of health information technologies. The goal of this thesis is to help the speed and accuracy of the diagnosis [70].

Although AI has been around for decades, in machine learning, a machine can take a dataset, analyze it, and make a decision or prediction based on what it has learned. Deep learning is a more complex version of this, where there are several layers of process features and each layer takes some information, deep learning helps researchers analyze medical data to treat diseases such as OCT images [71].

So many research papers have been published to use deep learning, and neural networks such as diagnose Alzheimer's years before symptoms appear, and predicting diseases such as severe diabetes, schizophrenia, and certain cancers. Also, the use of deep learning in diagnosing some eye diseases and diseases related to the retina. Here, some of the papers related to detected and diagnoses of retinal diseases using various techniques of deep learning, some of them shown below,

Many methods and models have been introduced that have contributed to increased diagnostic efficiency. Some of these prior studies involving OCT images segmentations: Lee et. al [72] developed a model to discover intraretinal fluid (IRF) on OCT in a way that can't be distinguished by doctors.
The model was based on a convolutional neural network (CNN) and they have extracted 1,289 OCT images from the database at the University of Washington Ophthalmology Department. These images were divided to 70% as training & the rest for the validation set. The model was trained using Keras, Tensorflow, Graphics Processing Units (GPU) with NVIDIA Cuda and Adam optimizer with a learning rate set at 1e-7.The results were great compared by experts, the model achieved a cross-validation Dice coefficient of 0.911.

In this paper[53] focused on classification OCT images using pre-trained CNN (Convolutional Neural Network) "VGG16" images to abnormal and normal OCT. The dataset used in the proposed algorithm from the Singapore Eye, that consists of 32 OCT volumes (16 DME and 16 normal cases) & the model achieved an accuracy of 87.5%, with sensitivity and specificity being 93.5% and 81% respectively.

Xu et. al [74] developed an approach using 40000 images to increase the resolution of Optical Coherence Tomography (OCT). They built architecture called SRUnet used Keras framework with a Tensorflow backend with learning rate was being 10−4.

Lee et. al [75] used a deep learning method to distinguish normal OCT images from patients with age-related macular degeneration (AMD). So that the extraction of an OCT database was performed and linked to clinical endpoints from the EMR.
This study was approved by the Institutional Review Board of the University of Washington (UW) and adhered to the tenets of the Declaration of Helsinki and the Health Insurance Portability and Accountability Act. Macular OCT scans from 2006 to 2016 were extracted using an automated extraction tool from the Heidelberg Spectralis (Heidelberg Engineering, Heidelberg, Germany) imaging database.
The image set was then divided into 2 sets, with 20% of the patients in each group placed into the validation set; the rest were used for training. Main Outcome Measure was used Receiver operating characteristic (ROC) curves were constructed at an independent image level, macular OCT level, and patient level.
The model has used a pre-trained modified version of the VGG16 convolutional neural network23 with python that was used to perform deep learning. As a result, deep learning technique achieves high accuracy and is effective as a new image classification technique.

In this paper [76], the authors proposed an end-to-end trained deep learning-based retinal fluid segmentation type (IRF/SRF/PED), a method that works well across 3D-OCT images. The proposed methods do not require any additional information such as layer segmentation for training or pre-processing.
The proposed method is based on a modified version of U-Net architecture trained using a combined loss function consisting of an adversarial loss term. Validation results show that the proposed method has been successful in predicting the presence and voxel-level segmentation of retinal fluids.

Karri et.al [77] proposed an approach to detect and identifying retinal pathologies given retinal using (OCT) images. Their way through fine-tunes a pre-trained convolutional neural network especially GoogLeNet to improve its prediction capability. They deal with the dataset including images with diabetic macular edema, or dry age-related macular degeneration, or normal. GoogLeNet is an inception layer that informs sparsity and multiscale (convolving with different filter sizes) information in one block. Functionally it is equivalent to a small network inside a large network. The filters and weights of GoogLeNet are identified iteratively through error back-propagation and the size of images 224 × 224 with three input channels.

In this paper [58], the authors have discussed the importance of artificial intelligence, deep learning, and machine learning techniques in medicine, significant progress of DL systems has been explaining in image-centric specialties such as radiology, dermatology, pathology, and ophthalmology, they described eye disease burden, unmet needs and common situation of public health importance for which AI and DL systems may be available.

Yoo et. al [79] have built models consisted of pre-trained VGG-19 and along with MatConvNet. VGG-19 included 16 convolutional layers and 3 fully connected layers. The results obtained were very useful AMD diagnosis through OCT data in deep learning techniques. Sertkaya et. al [80] have used different CNN models for example LeNet, AlexNet, and Vgg16 architectures of deep learning for the diagnosis of some diseases in the retina of the eye such as Neovascularization, Diabetic Macular Edema, Drusen and healthy eye retinal images using OCT images. The results of the implementation presented successful outcomes in Vgg16 and AlexNet architecture. The dropout layer structure in AlexNet has been shown to reduce the loss by minimizing loss. The VGG-16 architecture given a good classification result of 93.01 %. Motozawa et al [81] have proposed two classification models. The first was trained and validated using 1382 AMD images and 239 normal images, and the second was trained and validated with 721 AMD images with exudative changes and 661 AMD images, and both models showed good performance.
Li et. al [82] have proposed a new approach to provide an optical coherence tomography (OCT) image-based diagnostic technology for automated early DR diagnosis. They developed and evaluated a new version deep network OCTD_Net, for early-stage DR detection that consists of Org_Net and Seg_Net. The Org_Net (red dot arrows) used DenseNet blocks integrated with Squeeze-and-Excitation (SE) blocks to extract features from the original OCT images. The aim of their study is to reduce the rate of vision loss and enabling timely and accurate diagnosis.

Alqudah [83] has proposed an automated convolutional neural network (CNN) architecture for a multiclass classification system based on spectral-domain optical coherence tomography. They used to classify five types of retinal diseases AMD, CNV, DME, and drusen in addition to normal cases. They trained and tested using a huge dataset of images (136,187 images) by tuning the proposed CNN network structure and using the ADAM optimizing method.

The authors of this paper [84] have notified about some of the challenges that face performances of deep learning models when using OCT images and discussed the present possible solutions. The first issue is lack of image datasets from multiple OCT devices to optimize training and reduce overfitting, so some researchers used common transformations, including flipping, shearing, rotation, and resizing, are generalizable from original OCT imaging, the second issue is nonstandardized imaging or postprocessing protocols between devices, that means images of different sizes, contrast levels, and textures, lead to signal-to-noise ratios among devices result in variable image quality, so other researchers united image size and intensities for all the scans across OCT devices, the third issue is limited graphics processing unit capabilities for utilizing 3-dimensional features, some of the architectures consist of 16 layers or more, that requires multiple GPUs for a large batch size or reducing the batch size significantly during training, so advances in GPUs are incoming , including application-specific integrated circuits and the recently released NVIDIA DGX-2, which has 256 GB of GPU DRAM. Until then, this GPU very useful for deep learning for training OCT algorithms.

3.2 Comment about previous Studies

For a long time ago, there has been increased interest in deep learning, a promising class of machine learning models that use multiple neural network layers to rapidly. Trained with large data sets, deep learning has been used successfully in parallel with the advent of electronic medical records such as OCT images, X-rays, CT, and others has provided a wonderful opportunity to achieve milestones in automated image analysis, and the ability to increase of the resolution of medical images has the potential to impact significantly on medical imaging at the point of care, allowing significant small details to be revealed efficiently and accurately to improve deep learning model with high performance, and more accurate diagnosis of disease.

4.1 Dataset

Implementing a deep learning model to detect early deformities within the retina through a collection of images of eye patients stored in an electronic medical database. There are 84,495 images (JPEG) and 4 categories (NORMAL, CNV, DME, DRUSEN) were selected from a network research center and another eye surgery center, as shown in Fig.21. But in the training dataset, NORMAL dataset with 31.54% volume, CNV with 44.56%, DME with 13.6%, and DRUSEN with 10.3%. As a result of that, we faced the problem of training a model to perform against highly imbalanced data, to solve this problem, we used oversampling techniques by augmentation data for unbalanced class randomly to increase the number of images which are just copies of existing images. This ideally gives us a sufficient number of samples to play with, so the data was redistributed in another way as follows in table.5:

Table 5: Distribution of OCT images.

Categories Number of	Number of Training Images	Number of Testing Images	Images size (pixels)
CNV	8000	2000	64 X 64
DME	8000	2000	64 X 64
DRUSEN	8000	2000	64 X 64
NORMAL	8000	2000	64 X 64

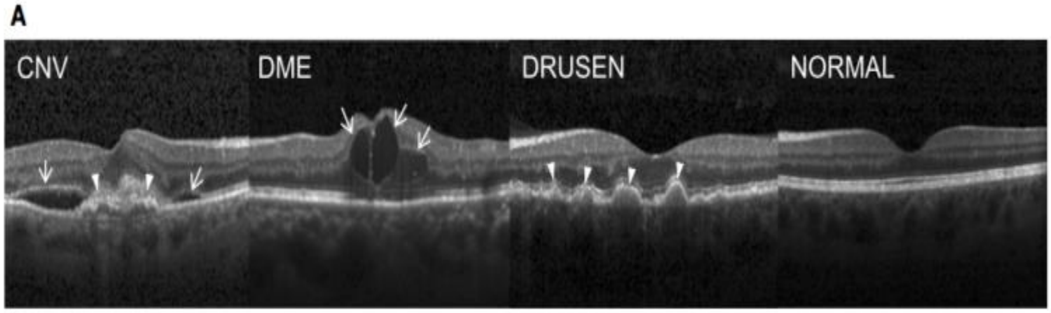


Fig 21: OCT Classifications (CNV, DME, DUSEN, and NORMAL).

4.2 Language and tool used

The model was designed using Python language with Keras & Tensorflow libraries language in the Google Colab environment and the system is supported through Graphics Processing Units (GPU) with NVIDIA to make the execution of the deep learning algorithms faster. Google Colab is a free, browser-based notebook environment that runs entirely in the cloud.

Here are a few reasons to use it:

1. Colab allows us to instantly spring up a Jupyter Notebook completely serverless in the browser.
2. We don't require doing an environment setup.
3. Free GPU.
4. Store Notebooks on Google Drive.
5. Load Data from Drive.

4.3 Image format

There are OCT images (JPEG) from 4 categories (NORMAL, CNV, DME, DRUSEN), and these OCT images (Spectralis OCT, Heidelberg Engineering, Germany) were selected from retrospective cohorts of adult patients from the Shiley Eye Institute of the University of California San Diego, the California Retinal Research Foundation, Medical Center Ophthalmology Associates, the Shanghai First People's Hospital, and Beijing Tongren Eye Center between July 1, 2013, and March 1, 2017.

The image format (png, jpg, gif) does not affect the training of the neural network model, the format in which the image is encoded has to do with its quality. Neural networks are essentially mathematical models that perform lots and lots of operations (matrix multiplications, element-wise additions, and mapping functions). A neural network sees a Tensor as its input (i.e. a multi-dimensional array). Its shape usually is (number of images per batch, image height, image width, number of channels).

4.4 Preprocessing

The images must match the input size of the network to train a network and make predictions on new data but the images come from different sources. so preprocessing is used to conduct steps that reduce the complexity and increase the accuracy of the applied algorithm, for example, the images were resized to 64 by 64 Pixels or 75 by 75 Pixels as the images were of various sizes to fit the training model.

4.5 Data augmentation

It is an approach of generating more training data from existing training samples, by augmenting the samples via a number of random transformations that yield believable-looking images to prevent overfitting.

The goal is that at training time, our model would never see the exact same picture twice. This helps the model get exposed to more aspects of the data and generalize better.

In Keras, this can be done by configuring a number of random transformations to be performed on the images read by our ImageDataGenerator instance, I used rotation_range 30 degrees, width_shift and height_shift are equals 0.2, horizontal_flip, and vertical_flip, as shown in Fig.22 and Fig.23 .

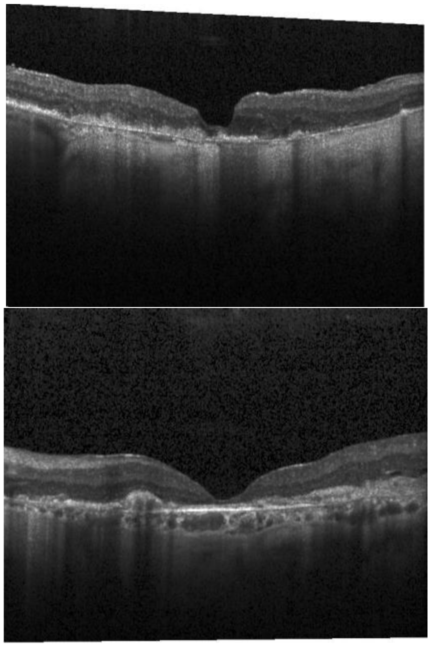


Fig 22: OCT Images before Augmentation.

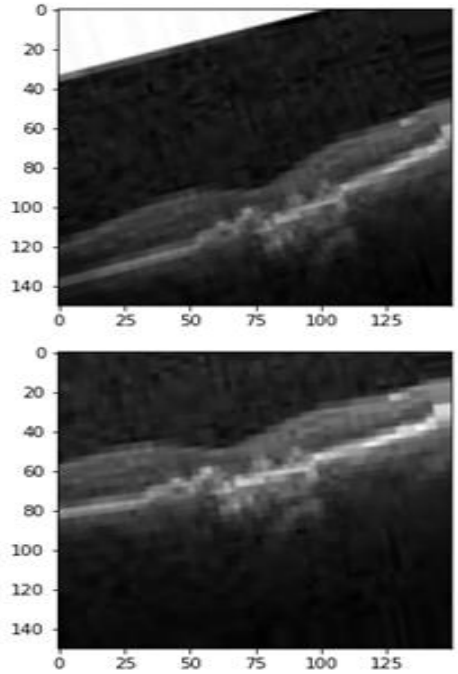


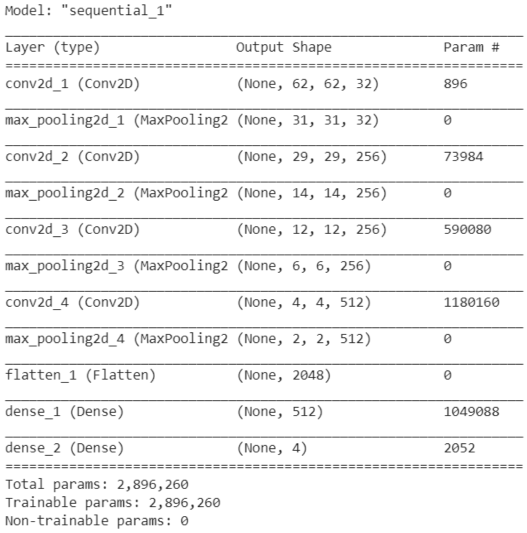
Fig 23: OCT Images after Augmentation.

4.6 Network Architecture

In this thesis used the scratch model to train datasets, besides five classical CNN models used to training and testing datasets including (VGG-16, MobileNet, InceptionV3, Xception, and ResNet-50).

4.6.1 Scratch model

The scratch model was built with 8000 training samples for every class. The model consists of a general structure convnet will be a stack of alternated Conv2D (with Relu activation) and MaxPooling2D layers. The images of the size 64 x64, end up with feature maps of size 2x2 right before the Flatten layer, and the depth of the feature maps is progressively increasing in the network (from 32 to 512), while the size of the feature maps is decreasing (from 62x62 to 2x2), as shown in Fig.24.



```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_1 (Conv2D)            (None, 62, 62, 32)        896       
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 32)        0         
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 29, 29, 256)       73984     
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 256)       0         
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 12, 12, 256)       590080    
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 6, 6, 256)         0         
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 4, 4, 512)         1180160   
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 2, 2, 512)         0         
_________________________________________________________________
flatten_1 (Flatten)          (None, 2048)              0         
_________________________________________________________________
dense_1 (Dense)              (None, 512)               1049088   
_________________________________________________________________
dense_2 (Dense)              (None, 4)                 2052      
=================================================================
Total params: 2,896,260
Trainable params: 2,896,260
Non-trainable params: 0
```

Fig 24: Architecture of Scratch Network.

4.6.1.1 Training the model

Training Dataset is the actual dataset that we use to train the model (weights and biases in the case of a Neural Network).
For the compilation step, the Adam optimizer was used with the learning rate (lr=0.0001) to provides perhaps the most important hyperparameter to tune for the model in order to achieve good performance.

To resume model training, and get the best possible model, also saved the one with the best (maximum) validation accuracy, Keras provide a great function, it's of checkpoint networks. In Deep Learning, the checkpoint can be used to continue training on the same dataset for additional epochs or to train on new data for additional epochs.

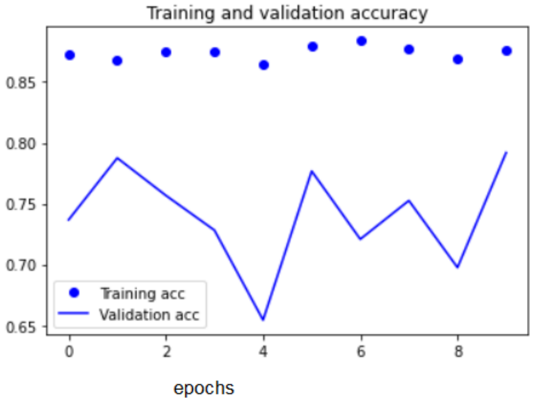


Fig 25: Accuracy Plot of Initial Training

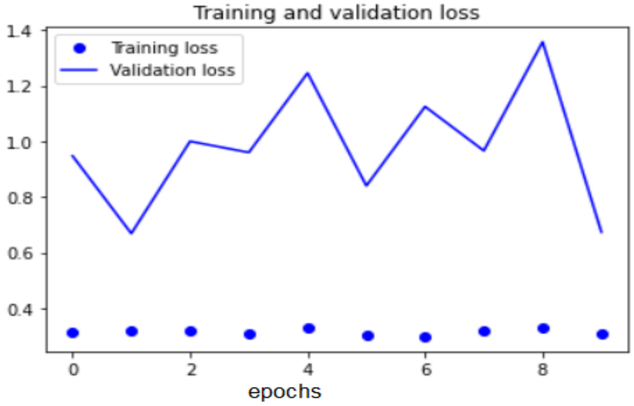
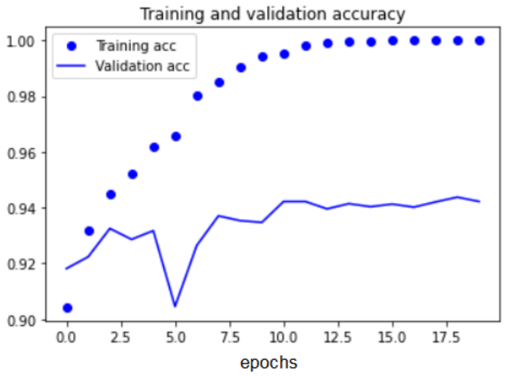


Fig 26: Loss Plot of Initial Training.

As shown in Fig.25 and Fig.26, these charts are characteristic of the training and validation loss side by side, as well as the training and validation accuracy, in which the number of epochs represents 10.

4.6.1.2 Validation of the model

To estimate model properties (mean error for numeric predictors, classification errors for classifiers, recall, and precision for the model, etc.). The data set was divided into a training set equals to 25600 images and the validation set equals 6400 images. That's means 80% was designated as a training set used for the optimization of the model. The remaining 20% were designated as a validation set. These were used for cross-validation during the optimization procedure.



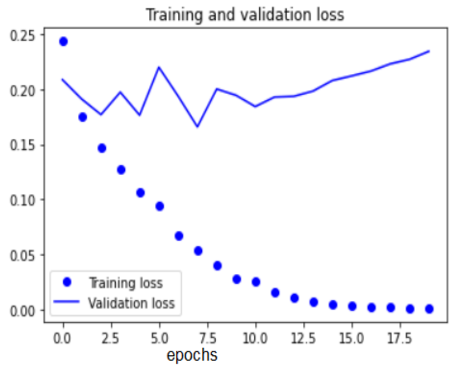


Fig 28: Training and Validation Accuracy.

Fig 29: Training and Validation Loss.

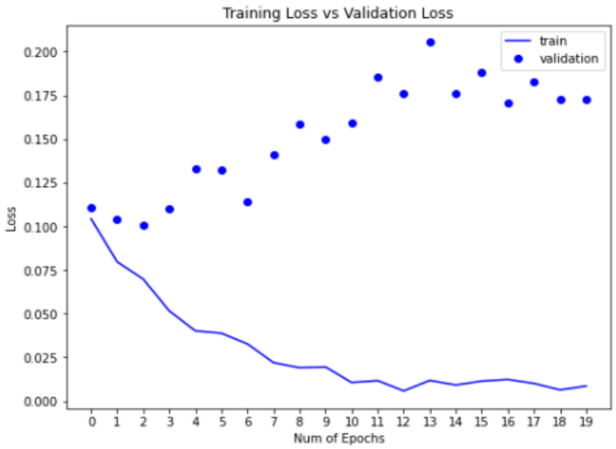
As shown in Fig.28 and Fig.29, all of the above indicates good learning results of the network during all stages. However, it still can serve as a baseline for comparison against the other methods.

4.6.2 VGG16 Model

VGG-16 was created from a 16-layer network comprised of convolutional and fully connected layers, also used dropout layers in between, dropout regularizes the networks to prevent overfitting, and all layers have ReLU activations except the output layer. The training of the model was done on a training set of 25600 images, and a validation set 6400 images using Transfer Learning with batch size and epochs as 256 and 20 respectively. The Categorical Crossentropy loss function, softmax activation, Adam optimizer, and metric function Accuracy were used to evaluate VGG16 model performance.

4.6.2.1 Training and Validation of VGG16 Model

For the VGG16 base model was resized the images to 75×75 pixels, the Adam optimizer was used with learning rate (lr=0.001), and the model achieved an accuracy of 96.38% on the validation set, and 99.73% for the training), as shown in Fig.31 and Fig.32.



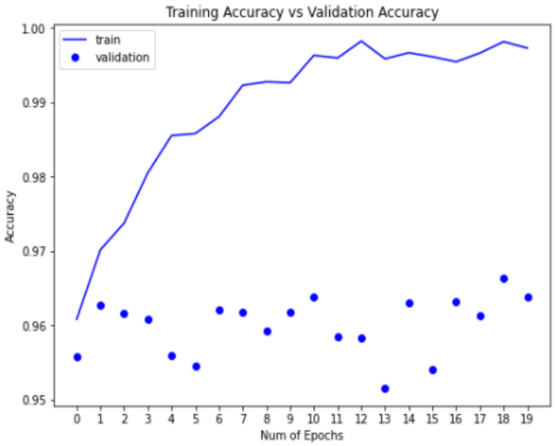


Fig 31: Training and Validation Accuracy

Fig 32: Training and Validation Loss

4.6.3 MobileNet Model

MobileNet is a pre-trained version of the network trained on more than a million images from the ImageNet database. The training of the model was done on a training set of 25600 images, and a validation set 6400 images using Transfer Learning with batch size and epochs as 256 and 20 respectively. The Categorical Crossentropy loss function, Adam optimizer, and metric function Accuracy were used to evaluate MobileNet model performance.

4.6.3.1 Training and Validation of MobileNet Model

For the MobileNet base model was resized the images to 75×75 pixels, the Adam optimizer was used with a learning rate (lr=0.0001), and the model achieved an accuracy of 95.53% on the validation set, and 99.99% for the training, as shown in Fig.34 and Fig.35.

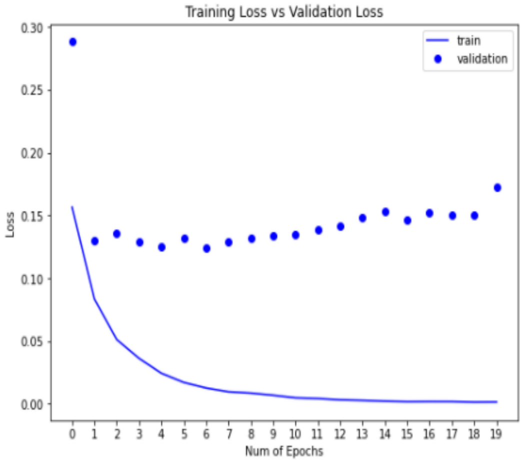


Fig 34: Training and Validation Accuracy

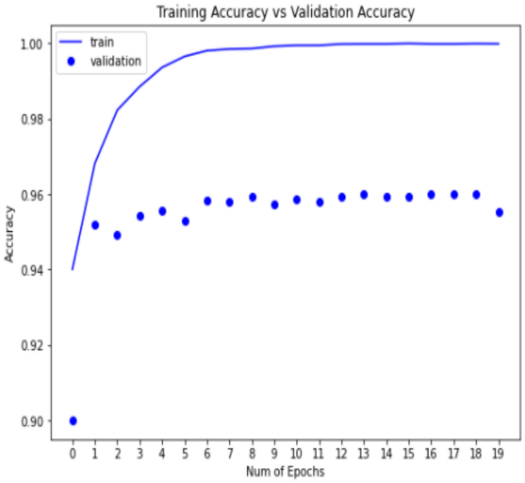


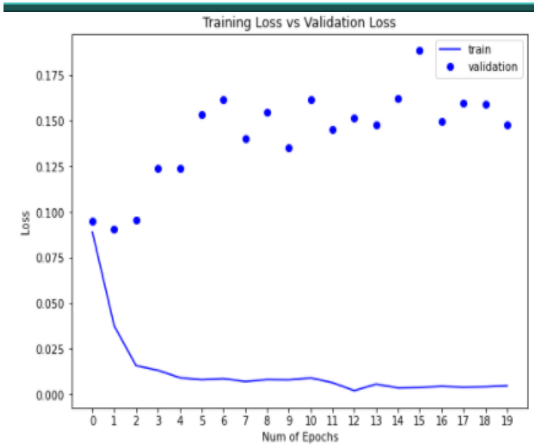
Fig 35: Training and Validation Loss

4.6.4 InceptionV3 Model

InceptionV3 is a convolutional neural network for assisting in image analysis and object detection and got its start as a module for GoogleNet. It is the third edition of Google's Inception, originally introduced during the ImageNet Recognition Challenge. The training of the model was done on a training set of 25600 images, and a validation set 6400 images using Transfer Learning with batch size and epochs as 256 and 20 respectively. The Categorical Crossentropy loss function, Adam optimizer, and metric function Accuracy were used to evaluate the InceptionV3 model performance.

4.6.4.1 Training and Validation of InceptionV3 Model

For the Inception V3 base model was resized the images to 75×75 pixels, the Adam optimizer was used with a learning rate (lr=0.0001), and the model achieved an accuracy of 97.1% on the validation set, and 99.86% for the training, as shown in Fig.37 and Fig.38.



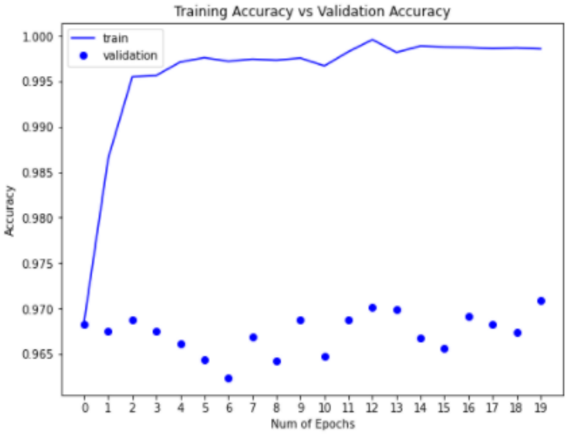


Fig 37: Training and Validation Accuracy

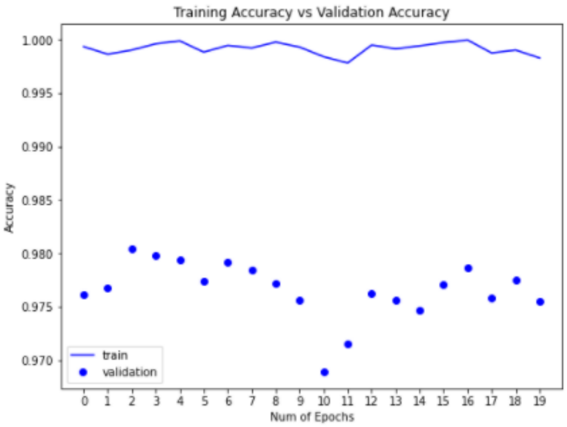
Fig 38: Training and Validation Loss

4.6.5 Xception Model

Xception is the improvement of Inception in 2017, which is a kind of extreme inception and uses the depthwise separable convolution layer to replace the convolution layer within InceptionV3. The training of the model was done on a training set of 25600 images, and a validation set 6400 images using Transfer Learning with batch size and epochs as 256 and 20 respectively. The Categorical Crossentropy loss function, Adam optimizer, and metric function Accuracy were used to evaluate Xception model performance.

4.6.5.1 Training and Validation of Xception Model

For the Xception base model was resized the images to 75×75 pixels, the Adam optimizer was used with a learning rate (lr=0.0001), and the model achieved an accuracy of 97.55% on the validation set, and 99.83% for the training , as shown in Fig.40 and Fig.41.



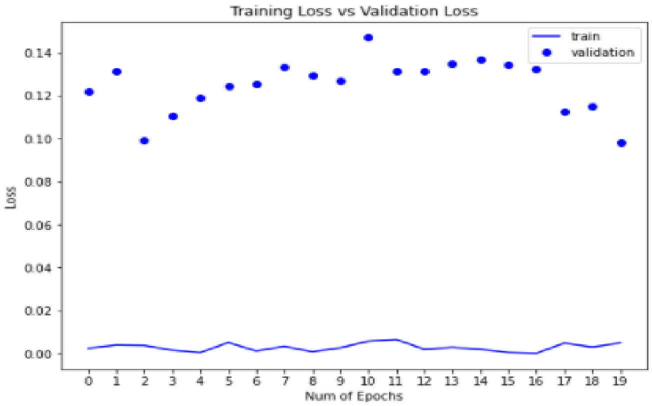


Fig 40: Training and Validation Accuracy

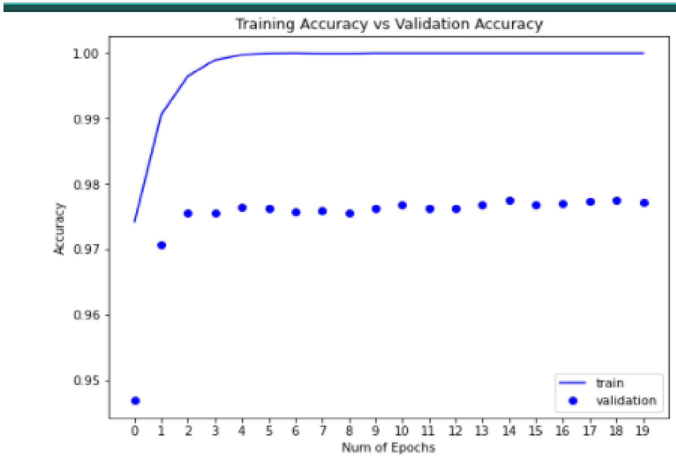
Fig 41: Training and Validation Loss

4.6.6 ResNet-50 Model

ResNet-50 is a convolutional neural network that is 50 layers deep. The training of the model was done on a training set of 25600 images, and a validation set 6400 images using Transfer Learning with batch size and epochs as 256 and 20 respectively. The Categorical Crossentropy loss function, Adam optimizer, and metric function Accuracy were used to evaluate ResNet-50 model performance.

4.6.6.1 Training and Validation of ResNet-50 Model

For the Xception base model was resized the images to 64×64 pixels, the Adam optimizer was used with a learning rate (lr=0.0001), and the model achieved an accuracy of 97.69% on the validation set, and 100% for the training.



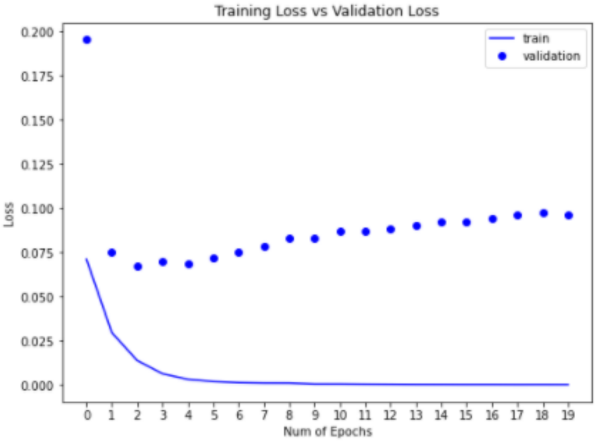


Fig 43: Training and Validation Accuracy

Fig 44: Training and Validation Loss

4.7 Result of training models and discussion

Based on the results in Fig.45, Fig.46, Table.6, and Table.7, the best model was ResNet of training data, because of achieving the highest training accuracy 100 %, and the highest validation accuracy 97.69%, also the lowest training loss .001 % and, validation loss 9.6%, these results obtained by using Keras techniques such as Adam optimization, data augmentation, and others, in addition to these techniques, the structures of models play a role of this result. ResNet Unlike traditional sequential network architectures such as AlexNet, OverFeat, VGG16, and others, ResNet is instead a form of “exotic architecture” that relies on micro-architecture modules (also called “network-in-network architectures”).

The term micro-architecture refers to the set of “building blocks” used to construct the network. A collection of micro-architecture building blocks (along with your standard CONV, POOL, etc. layers) leads to the macro-architecture (i.e, the end network itself).

Table 6: Training and Validation Accuracy for All Models

Model	**Training Accuracy (%)**	**Validation Accuracy (%)**	**Validation F-score (%)**
Scratch	100	94.22	94.20
VGG16	99.73	96.38	96.39
MobileNet	99.99	95.53	95.54
Inception V3	99.86	97.1	97.1
Xceptio	99.83	97.55	97.55
ResNet-50	100	97.72	97.69

Table 7: Training and Validation Losses for All Models

Model	**Training Loss (%)**	**Validation Loss (%)**
Scratch	0.14	23.41
VGG16	0.86	17.29
MobileNet	0.15	17.24
Inception V3	0.46	14.76
Xception	0.49	9.8
ResNet-50	0.001	9.6

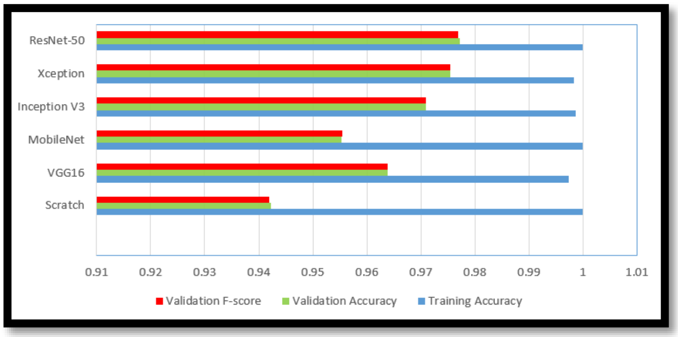


Fig 45: Accuracy Chart of All Models.

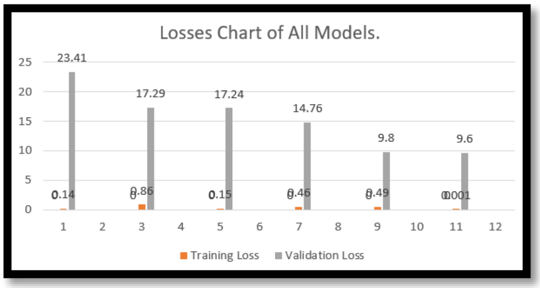


Fig 46: Losses Chart of All Models.

5.1 Data Set for testing of each model

Testing dataset organized into one folder and contains 8000 of OCT images, (JPEG) format , different from the images that used in the original dataset for training; they are images of four classifications of Retina diseases, as shown in Fig.47, the same dataset used for each model (Scratch, VGG-16, MobileNet, InceptionV3, Xception, and ResNet-50) distributed as in the following table.8:

Table8 : Distribution of Testing Images

Categories	**Number of Testing Images**	**Images size (pixels)**
CNV	2000	64 X 64
DME	2000	64 X 64
DRUSEN	2000	64 X 64
NORMAL	2000	64 X 64

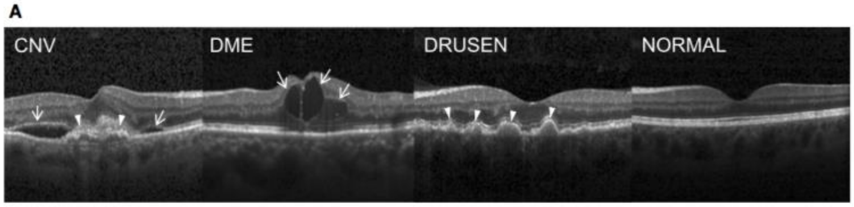


Fig 47: OCT Classifications (CNV, DME, DUSEN, and NORMAL).

5.2 Testing each model

After training and evaluating the model on the validation, the models are tested using 8000 OCT images, among them, 2000 CNV, 2000 DME, 2000 DRUSEN, and 2000 NORMAL. Testing the models are done through load the test images and predict their classes using the model.predict(x) function, probabilities of each image belonging to a specific class were calculated, by the following classification:

5.3 Result and Discussion

This thesis presents a deep learning approach for diagnosis through the classification of OCT images into Retinal diseases including CNV, DME, DRESUN, and Normal, the deep learning models tested included five classical CNN models and the following table shows the testing accuracy obtained for each model.

Table 9: Testing Accuracy

Model	Testing Accuracy (%)
Scratch	92.05
VGG16	93.65
MobileNet	93.60
Inception V3	95.3
Xception	95.89
ResNet-50	96.21

Based on the results in Fig.45, Fig.46, Fig.48, Table.6, Table.7 and Table.9, the best model was ResNet, because of achieving the highest testing accuracy of 96.21%, and the highest training accuracy 100%, also the highest validation accuracy 97.69%, the results obtained by using Keras techniques such as Adam optimization, data augmentation, learning rate, and others, the internal structure of the ResNet-50 model "network-in-network architectures" helped to obtain these results.

Also, when we used the Oversampling technique to modify the unequal data classes to create balanced datasets, and this technique had a very significant impact on the results by preventing overfitting.

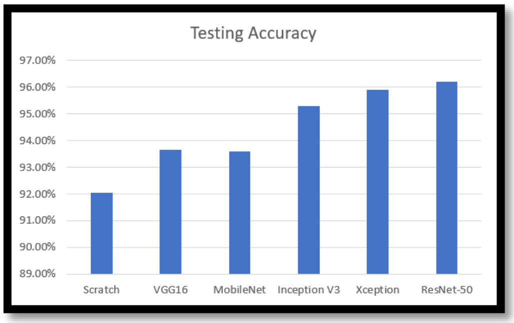


Fig 48: Testing Accuracy Chart.

OCT is a great tool for detecting eye disease, but interpreting this data takes time, which creates difficult and hard in the diagnostic process. The deep learning model achieves high accuracy and is efficient as a new image classification technique. This feedback has important and very helpful in utilizing OCT in automated and the development of computer-aided. By connecting previous results with the medical field, so algorithms and models can help patients, also they guide the doctors and provide them a new and effective tool for the diagnosis of eye diseases.

We were able to achieve all the goals that we mentioned in the first chapter by classifying 4 retinal classes (NORMAL, CNV, DME, and DRUSEN), and accordingly, it helps Ophthalmologists, doctors make faster, and more accurate diagnoses.

5.4 Comparison between the obtained result and the previous studies

Most of the previous studies were based on VGG16, SRUnet, U-Net, GoogLeNet models, and the testing accuracy results ranged between 90% - 94%, but we were based on VGG16, MobileNet, InceptionV3, Xception, and ResNet-50 models, we obtained a testing accuracy result of 96.21% from the ResNet-50 model is higher than the previous results, also the amount of dataset we obtained was different from the previous studies.

6.1 Conclusion

Nowadays, eye diseases are among the most common reason for vision loss, these diseases can cause blindness and affect the lives of patients.

In this thesis, some of the techniques were used deep learning-based, included five pre-trained deep learning models (VGG-16, MobileNet, Inception V3, Xception, and Res-Net) to classify four types of retinal (CNV, DME, DRUSEN, in addition to normal cases) from OCT images. Furthermore, the proposed models were also trained and tested using a huge dataset of images (40,000 images). Out of these, the ResNet-50 is the best network which observed that a step by step optimization process with achieves better overall performance and a better testing accuracy rate of 96.21%.

Despite several challenges need to be resolved to increase DL adoption in medicine, deep learning solutions are among the most promising solutions to improve the general health of a people, and to lower the costs of healthcare.

6.2 Future Work

In the future, the aim is to reduce the size of deep models for portable machines like mobiles to implement an online application to diagnose OCT images for helping doctors, patients, and clinics, also extension of this work will focus on developing hybrid algorithms that mean a combination of two models such as VGG-16 and ResNet-50 to increase the recognition rate of the final classification process underscoring the advantages of hybrid algorithms, and enhancing the performance of the model.

References:

1. Al Barsh, Y. I., et al. (2020). "MPG Prediction Using Artificial Neural Network." International Journal of Academic Information Systems Research (IJAISR) 4(11): 7-16.
2. Alajrami, E., et al. (2019). "Blood Donation Prediction using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 3(10): 1-7.
3. Alajrami, E., et al. (2020). "Handwritten Signature Verification using Deep Learning." International Journal of Academic Multidisciplinary Research (IJAMR) 3(12): 39-44.
4. Al-Araj, R. S. A., et al. (2020). "Classification of Animal Species Using Neural Network." International Journal of Academic Engineering Research (IJAER) 4(10): 23-31.
5. Al-Atrash, Y. E., et al. (2020). "Modeling Cognitive Development of the Balance Scale Task Using ANN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 74-81.
6. Alghoul, A., et al. (2018). "Email Classification Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 8-14.
7. Abu Nada, A. M., et al. (2020). "Age and Gender Prediction and Validation Through Single User Images Using CNN." International Journal of Academic Engineering Research (IJAER) 4(8): 21-24.
8. Abu Nada, A. M., et al. (2020). "Arabic Text Summarization Using AraBERT Model Using Extractive Text Summarization Approach." International Journal of Academic Information Systems Research (IJAISR) 4(8): 6-9.
9. Abu-Saqer, M. M., et al. (2020). "Type of Grapefruit Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 4(1): 1-5.
10. Afana, M., et al. (2018). "Artificial Neural Network for Forecasting Car Mileage per Gallon in the City." International Journal of Advanced Science and Technology 124: 51-59.
11. Al-Araj, R. S. A., et al. (2020). "Classification of Animal Species Using Neural Network." International Journal of Academic Engineering Research (IJAER) 4(10): 23-31.
12. Al-Atrash, Y. E., et al. (2020). "Modeling Cognitive Development of the Balance Scale Task Using ANN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 74-81.
13. Alghoul, A., et al. (2018). "Email Classification Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 8-14.
14. Al-Kahlout, M. M., et al. (2020). "Neural Network Approach to Predict Forest Fires using Meteorological Data." International Journal of Academic Engineering Research (IJAER) 4(9): 68-72.
15. Alkronz, E. S., et al. (2019). "Prediction of Whether Mushroom is Edible or Poisonous Using Back-propagation Neural Network." International Journal of Academic and Applied Research (IJAAR) 3(2): 1-8.
16. Al-Madhoun, O. S. E.-D., et al. (2020). "Low Birth Weight Prediction Using JNN." International Journal of Academic Health and Medical Research (IJAHMR) 4(11): 8-14.
17. Al-Massri, R., et al. (2018). "Classification Prediction of SBRCTs Cancers Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 1-7.
18. Al-Mobayed, A. A., et al. (2020). "Artificial Neural Network for Predicting Car Performance Using JNN." International Journal of Engineering and Information Systems (IJEAIS) 4(9): 139-145.
19. Al-Mubayyed, O. M., et al. (2019). "Predicting Overall Car Performance Using Artificial Neural Network." International Journal of Academic and Applied Research (IJAAR) 3(1): 1-5.
20. Alshawwa, I. A., et al. (2020). "Analyzing Types of Cherry Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 4(1): 1-5.
21. Al-Shawwa, M., et al. (2018). "Predicting Temperature and Humidity in the Surrounding Environment Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(9): 1-6.
22. Ashqar, B. A., et al. (2019). "Plant Seedlings Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 3(1): 7-14.
23. Bakr, M. A. H. A., et al. (2020). "Breast Cancer Prediction using JNN." International Journal of Academic Information Systems Research (IJAISR) 4(10): 1-8.
24. Barhoom, A. M., et al. (2019). "Predicting Titanic Survivors using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 3(9): 8-12.
25. Belbeisi, H. Z., et al. (2020). "Effect of Oxygen Consumption of Thylakoid Membranes (Chloroplasts) From Spinach after Inhibition Using JNN." International Journal of Academic Health and Medical Research (IJAHMR) 4(11): 1-7.
26. Dalffa, M. A., et al. (2019). "Tic-Tac-Toe Learning Using Artificial Neural Networks." International Journal of Engineering and Information Systems (IJEAIS) 3(2): 9-19.
27. Dawood, K. J., et al. (2020). "Artificial Neural Network for Mushroom Prediction." International Journal of Academic Information Systems Research (IJAISR) 4(10): 9-17.
28. Dheir, I. M., et al. (2020). "Classifying Nuts Types Using Convolutional Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(12): 12-18.
29. El-Khatib, M. J., et al. (2019). "Glass Classification Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 3(2): 25-31.
30. El-Mahelawi, J. K., et al. (2020). "Tumor Classification Using Artificial Neural Networks." International Journal of Academic Engineering Research (IJAER) 4(11): 8-15.
31. El-Mashharawi, H. Q., et al. (2020). "Grape Type Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 41-45.
32. Elzamly, A., et al. (2015). "Classification of Software Risks with Discriminant Analysis Techniques in Software planning Development Process." International Journal of Advanced Science and Technology 81: 35-48.
33. Elzamly, A., et al. (2015). "Predicting Software Analysis Process Risks Using Linear Stepwise Discriminant Analysis: Statistical Methods." Int. J. Adv. Inf. Sci. Technol 38(38): 108-115.
34. Elzamly, A., et al. (2017). "Predicting Critical Cloud Computing Security Issues using Artificial Neural Network (ANNs) Algorithms in Banking Organizations." International Journal of Information Technology and Electrical Engineering 6(2): 40-45.
35. Habib, N. S., et al. (2020). "Presence of Amphibian Species Prediction Using Features Obtained from GIS and Satellite Images." International Journal of Academic and Applied Research (IJAAR) 4(11): 13-22.
36. Harz, H. H., et al. (2020). "Artificial Neural Network for Predicting Diabetes Using JNN." International Journal of Academic Engineering Research (IJAER) 4(10): 14-22.
37. Hassanein, R. A. A., et al. (2020). "Artificial Neural Network for Predicting Workplace Absenteeism." International Journal of Academic Engineering Research (IJAER) 4(9): 62-67.
38. Heriz, H. H., et al. (2018). "English Alphabet Prediction Using Artificial Neural Networks." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 8-14.
39. Jaber, A. S., et al. (2020). "Evolving Efficient Classification Patterns in Lymphography Using EasyNN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 66-73.
40. Kashf, D. W. A., et al. (2018). "Predicting DNA Lung Cancer using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(10): 6-13.
41. Khalil, A. J., et al. (2019). "Energy Efficiency Predicting using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 3(9): 1-8.
42. Kweik, O. M. A., et al. (2020). "Artificial Neural Network for Lung Cancer Detection." International Journal of Academic Engineering Research (IJAER) 4(11): 1-7.
43. Maghari, A. M., et al. (2020). "Books' Rating Prediction Using Just Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 4(10): 17-22.
44. Mettleq, A. S. A., et al. (2020). "Mango Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 22-29.
45. Metwally, N. F., et al. (2018). "Diagnosis of Hepatitis Virus Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 1-7.
46. Mohammed, G. R., et al. (2020). "Predicting the Age of Abalone from Physical Measurements Using Artificial Neural Network." International Journal of Academic and Applied Research (IJAAR) 4(11): 7-12.
47. Musleh, M. M., et al. (2019). "Predicting Liver Patients using Artificial Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(10): 1-11.
48. Oriban, A. J. A., et al. (2020). "Antibiotic Susceptibility Prediction Using JNN." International Journal of Academic Information Systems Research (IJAISR) 4(11): 1-6.
49. Qwaider, S. R., et al. (2020). "Artificial Neural Network Prediction of the Academic Warning of Students in the Faculty of Engineering and Information Technology in Al-Azhar University-Gaza." International Journal of Academic Information Systems Research (IJAISR) 4(8): 16-22.
50. Sadek, R. M., et al. (2019). "Parkinson's Disease Prediction Using Artificial Neural Network." International Journal of Academic Health and Medical Research (IJAHMR) 3(1): 1-8.
51. Salah, M., et al. (2018). "Predicting Medical Expenses Using Artificial Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 2(20): 11-17.
52. Salman, F. M., et al. (2020). "COVID-19 Detection using Artificial Intelligence." International Journal of Academic Engineering Research (IJAER) 4(3): 18-25.
53. Samra, M. N. A., et al. (2020). "ANN Model for Predicting Protein Localization Sites in Cells." International Journal of Academic and Applied Research (IJAAR) 4(9): 43-50.
54. Shawarib, M. Z. A., et al. (2020). "Breast Cancer Diagnosis and Survival Prediction Using JNN." International Journal of Engineering and Information Systems (IJEAIS) 4(10): 23-30.
55. Zaqout, I., et al. (2015). "Predicting Student Performance Using Artificial Neural Network: in the Faculty of Engineering and Information Technology." International Journal of Hybrid Information Technology 8(2): 221-228.
56. Saleh, A. et al. (2020). "Brain Tumor Classification Using Deep Learning." 2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech). IEEE, 2020.
57. Almadhoun, H. et al. (2021). "Classification of Alzheimer's Disease Using Traditional Classifiers with Pre-Trained CNN." International Journal of Academic Health and Medical Research (IJAHMR) 5(4):17-21.
58. El-Habil, B. et al. (2022). "Cantaloupe Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 5(12): 7-17.
59. Alkahlout, M. A. et al. (2022). "Classification of Fruits Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 5(12): 56-63.
60. Alfarra, A. H. et al. (2022). "Classification of Pineapple Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 5(12): 37-41.
61. Al-Masawabe, M. M. et al. (2022). "Papaya maturity Classification Using Deep Convolutional Neural Networks." International Journal of Engineering and Information Systems (IJEAIS) 5(12): 60-67.
62. Abu-Jamie, T. N., et al. (2022). "Six Fruits Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 6(1): 1-8.
63. Aish, M. A., et al. (2022). "Classification of pepper Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(1): 24-31.
64. Alsaqqa, A. H., et al. (2022). "Using Deep Learning to Classify Different types of Vitamin." International Journal of Academic Engineering Research (IJAER) 6(1): 1-6.
65. MAlnajjar, M. K. and S. S. Abu-Naser (2022). "Heart Sounds Analysis and Classification for Cardiovascular Diseases Diagnosis using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(1): 7-23.
66. Aldammagh, Z., Abdeljawad, R., & Obaid, T. (2021). Predicting Mobile Banking Adoption: An Integration of TAM and TPB with Trust and Perceived Risk. Financial Internet Quarterly, 17(3), 35-46.
67. Obaid, T. (2021). Predicting Mobile Banking Adoption: An Integration of TAM and TPB with Trust and Perceived Risk. Available at SSRN 3761669.
68. Jouda, H., Abu Jarad, A., Obaid, T., Abu Mdallalah, S., & Awaja, A. (2020). Mobile Banking Adoption: Decomposed Theory of Planned Behavior with Perceived Trust. Available at SSRN 3660403.
69. Obaid, T., Abdaljawad, R., & Mdallalahc, S. A. (2020). Factors Driving E-Learning Adoption In Palestine: An Integration of Technology Acceptance Model And IS Success Model. Available at SSRN 3686490.
70. Obaid, T. F., & Eneizan, B. M. (2016). Transfer of training and post-training on job performance in Middle Eastern countries. Review of Public Administration and Management, 400(3786), 1-11.
71. Obaid, T. F., Zainon, M. S., Eneizan, P. D. B. M., & Wahab, K. A. TRANSFER OF TRAINING AND POST-TRAINING ON JOB PERFORMANCE IN MIDDLE EASTERN COUNTRIES.

72. Lee, C. S., Tyring, A. J., Deruyter, N. P., Wu, Y., Rokem, A., & Lee, A. Y.: Deep-learning based, automated segmentation of macular edema in optical coherence tomography. Biomedical Optics Express, 8(7), 3440, (2017).
73. Awais, M., Muller, H., Tang, T. B., & Meriaudeau, F.: Classification of SD-OCT images using a Deep learning approach. 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), (2017).
74. Xu,Y., Williams,B.,Al-Bander, B., Yan,Z., Shen,Y.,Zheng, Y.: Improving the Resolution of Retinal OCT with Deep Learning.MIUA2018, 046, v2(2018).
75. Lee, C., Baughman,D., Lee,A.: Deep Learning Is Effective for Classifying Normal versus Age-Related Macular Degeneration OCT Images, American Academy of Ophthalmology, (2016).
76. Venhuizen, F. G., van Ginneken, B., Liefers, B., van Asten, F., Schreur, V., Fauser, S., Hoyng, C., Theelen, T., Sánchez, C. I.: Deep learning approach for the detection and quantification of intraretinal cystoid fluid in multivendor optical coherence tomography. Biomedical Optics Express, 9(4), (2018).
77. Karri, S. P. K., Chakraborty, D., & Chatterjee, J.: Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration. Biomedical Optics Express, 8(2), (2017).
78. Ting, D. S. W., Peng, L., Varadarajan, A. V., Keane, P. A., Burlina, P., Chiang, M. F., Schmetterera,L.,Pasqualek,L.,Bresslerd,N.,Websterb ,D., Abramoffl,M., Wong, T. Y.: Deep learning in ophthalmology: The technical and clinical considerations. Progress in Retinal and Eye Research, (2019).
79. Yoo, T. K., Choi, J. Y., Seo, J. G., Ramasubramanian, B., Selvaperumal, S., & Kim, D. W.: The possibility of the combination of OCT and fundus images for improving the diagnostic accuracy of deep learning for age-related macular degeneration: a preliminary experiment. Medical & Biological Engineering & Computing, (2018).
80. Sertkaya, M. E., Ergen, B., & Togacar, M. : Diagnosis of Eye Retinal Diseases Based on Convolutional Neural Networks Using Optical Coherence Images. 2019 23rd International Conference Electronics, (2019).
81. Motozawa, N., An, G., Takagi, S., Kitahata, S., Mandai, M., Hirami, Y., Yokota, H., Akiba,M., Tsujikawa, A., Takahashi,M., Kurimoto, Y. :Optical Coherence Tomography-Based Deep-Learning Models for Classifying Normal and Age-Related Macular Degeneration and Exudative and Non-Exudative Age-Related Macular Degeneration Changes. Ophthalmology and Therapy, (2019).
82. Li, X., Shen, L., Shen, M., Tan, F., & Qiu, C. S.: Deep learning based early stage diabetic retinopathy detection using optical coherence tomography. Neurocomputing , (2019).
83. Alqudah, A.: a convolutional network automated classification of multiclass retinal diseases using spectral-Ddomain optical coherence tomography images, Medical & Biological Engineering & Computing, (2019).
84. Yanagihara, R. T., Lee, C. S., Ting, D. S. W., & Lee, A. Y.: Methodological Challenges of Deep Learning in Optical Coherence Tomography for Retinal Diseases: A Review. Translational Vision Science & Technology, 9(2), 11,(2020).