Can Functional States of Neurons be Predicted via Machine Learning?

Angel Wadhawan
The University of Adeliade

Mohsen Dorraki
The Australian Institute of Machine Learning

Abstract

Over the years, there has been a growing understanding of the fact that early intervention in disease is extremely vital to achieving positive outcomes, and curtailing the exponential growth of healthcare spending. This is evident by the fact that the risk for the transition of sustained psychosis can be reduced, at least temporarily by 50%. The sooner these diseases of the brain are identified, it becomes significantly easier to mitigate major harm that may be caused. The Australian Institute of Machine Learning (AIML) and the South Australian Health and Medical Research Institute (SAHMRI) collaborated on this project to explore the option of predicting the functional states of human neurons. This can provide us with insight into diseases of the human brain and how they affect the neurons. Here we show how machine learning algorithms can be used to classify images of human neurons. We found that with the use of Convolutional neural network we were able to achieve a 97% accuracy in correctly classifying the images of neurons. This demonstrates that using machine learning can help identify and diagnose diseases of the brain from the images of the neurons. We anticipate our research to be a starting point in using extensive machine learning algorithms in diagnosing the diseases of the brain using the behaviour and images of the neurons under various conditions.

1 Introduction

The major part of the human brain comprises neurologically active cells called ‘neurons’. Neurons are the fundamental units of the brain and the nervous system, these special cells are responsible for receiving sensory input, sending motor responses, and transforming electrical signals at every step in between. A typical neuron has three main parts: Dendrites, an axon, and soma or cell body [1]. These three parts of the neuron combined can be compared to that of the branches, roots, and trunk of a tree respectively.

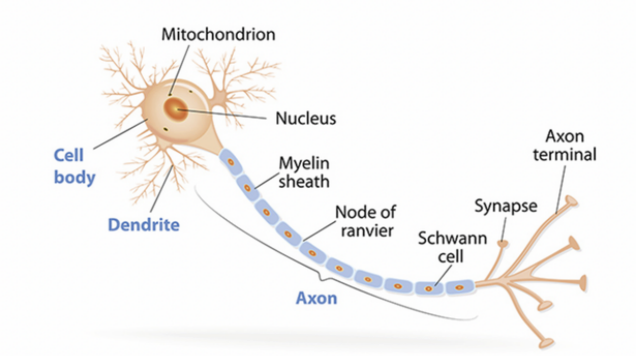

Figure 1.1: Structure of a neuron [1]

The Axon is the cable-like structure that develops away from the cell body and is responsible for the transmission of electrical impulses to other neurons. The dendrites are the receiving part of the neuron that receives that electrical signal from other neurons. The sum total of the dendritic inputs determines if the neuron will fire an action potential. The neurons communicate using electrical impulses called ‘synapses’. These synapses occur between the axon terminals and the dendrites or cell bodies [1].

The analysis of these neurons is important in investigations of numerous physiological and pathological conditions. Different alterations of neurons are often not easily observed biochemically but require detailed visual analysis [2].

1.1 Introduction to Patch Clamping

Early attempts to characterize the specific biophysical properties of ion channels were conducted in large preparations, such as the squid giant axon that Hodgkin and Huxley used for their seminal characterization of sodium and potassium channels,1 work that merited the Nobel Prize in 1963. Subsequently, techniques were developed to study ionic currents in smaller preparations and in single cells, a process that culminated in 1976 with the landmark publication by Erwin Neher and Bert Sakmann, "Single-channel currents recorded from membrane of denervated frog muscle fibers". They reported a new technique that involved the use of micropipettes to form high resistance seals on tiny patches of the cell membrane, thus significantly reducing the surface area of the membrane being studied and allowing, for the first time, high fidelity recordings of single ion channels. The importance of this development was recognized immediately by the scientific community and was highlighted by their being awarded the Nobel Prize in Medicine in 1991. The work of Neher and Sakmann ushered in a new era of inquiry into the role of ion channels in a diverse spectrum of essential biological processes and paved the way for an improved understanding of how ion channels function under both physiologic and pathologic conditions [3]–[5].

In 1963, the initial attempts to characterize specific biophysical properties of ion channels were conducted, this was when the squid giant axons were used by Hodgkin and Huxley for their seminal characterization of sodium and potassium channels, a study which later went on to win the Nobel Prize in 1963. Following this, in 1976, techniques were developed to study with a focus on

The technical aspects of the now ubiquitous "patch-clamp" technique have been reviewed in a number of excellent publications, as has the application of this technique to recording ionic currents in a wide variety of tissue types [6]–[8].

1.2 Machine Learning and its application in Medicine

The adoption of data-intensive machine-learning methods can be found throughout science, technology, and commerce, leading to more evidence-based decision-making across many walks of life, including health care, manufacturing, education, financial modeling, policing, and marketing [9]. Machine learning is the practice of using algorithms to parse and evaluate the data and learn from the data and use this learning to make predictions about something. It is the science of getting computers learn from observations as humans do and used that learning for future decision making.

The algorithms of Machine learning are useful in identifying complicated patterns within prosperous and huge data. This facility is especially well-suited to clinical applications, particularly those for people who rely on advanced genomics and proteomics measurements. It is often used in numerous illness diagnoses and detection. In medical applications, machine learning algorithms will manufacture higher decisions regarding treatment plans for patients by suggestions of implementing useful healthcare systems [10].

All of the above-mentioned methods are performed in a laboratory and require a lot of time and effort. Moreover, these methods are not only hard to reproduce but are also not scalable to much extent.

2 Motivation

There is a growing consensus that early intervention is vital for more positive outcomes in diseases and to curtail the exponential growth of healthcare spending. The prolonged period of delay in diagnosis and treatment initiation places a substantial burden on the patient and puts them at risk along with increasing the cost of healthcare [6]. The sooner we identify the diseases, especially those of the brain, the easier it is to prevent major harm or loss.

Diseases of the brain are often associated with aging, which is on the increase as the human lifespan lengthens, causing untold suffering to individuals, families, and society as a whole. For instance, in cases of Multiple Sclerosis, using the current diagnostic gold standard, even patients with an early diagnosis have experienced some level of brain damage [11]. Following that thought, it is the brain that makes who we are and it can be argued that the other organs are there to simply support the brain. For this reason, understanding the biology of the brain is a major goal for modern science [12]. The study of the varied development rate of the human progenitors into electrophysiologically active neurons will shine a light on the new aspects of disease-related discoveries in neurology, and psychiatry and shine a light on newer ways to diagnose diseases of the brain.

Focus on early detection, diagnosis, and treatment of Alzheimer's disease (AD) has been increasing. The rationale is that, as with any other serious illness, early intervention will lead to better outcomes for patients and families [13].

The current method of studying the neurons is Patch clamping with morphology and RNA-seq on a single neuron. However, this method is lengthy and highly delicate [14]. Additionally, most of the studies include isolating the neuron from its natural habitat, which clearly leads to the loss of anatomical positional information. Moreover, the dissociation process is very stressful, if not devastating, to cells, especially neurons. This could change the transcriptional program quite easily [15].

In this study, we are proposing the use of machine learning to classify the images of neurons. The model will classify images of neurons into various functional states. Image classification is a quick and relatively simple method that will not only help save time but will also require less effort. The model can also be scaled and reproduced without much time and effort.

3 Literature Review

A new technique of isolating the neurons *in-vitro* and using the combination of Patch clamping, RNA-seq, and morphology on a single human neuron was proposed recently. Then, using a machine learning model the study classifies the data and reveals biomarkers to predict the functional states of neurons. The authors used the recorded action potentials also called 'spike' data to classify the isolated neurons from 'Type 0' to 'Type 5', where 'Type 0' has little to no functionality and 'Type 5' represents a fully functional neuron.

To classify the functional states of the neurons based on their molecular phenotype, an extremely randomized tree classifier was used over a 90-10 split of the data. 10 cross fold-validation was also used to achieve better accuracy. With the aforestated methods, the authors were able to achieve more than 85% accuracy. For this experiment, the neurons had to be kept in the neural medium for around 30 days. However, the full range tested throughout the study was 0 to 30 weeks. We believe that by using multimodal machine learning we can use image classification by itself and achieve a comparable result, on the images of neurons.

Another neuroscience study compiled a large ground truth dataset of simultaneous action potentials and somatic calcium signals on zebrafish and mice. The authors developed a supervised deep network named CASCADE.

CASCADE outperformed other model-based methods owing to its large ground truth database. It also made good predictions for the broadest set of neurons, where this level of high-level performance by the model-based algorithms was observed in fewer datasets. However, the algorithm is not very well suited for the signals of multiple neurons i.e. the neuron understudy has to be isolated. This makes the algorithm impractical for the study of neurons that are parts of the brain at the moment. Additionally, the reliability of the spike inference depends on the recording of the calcium aging data, which is very sensitive to human error and bias.

In another study [15], the authors argue that the study of neurons that are isolated leads to a loss of not only the anatomical positional information of the cells but also the process of dissociation causes the cells to lose their transcription program. To counter this drawback the authors used the human cell-based in vitro model. These neurons matured in cultures that retain their anatomic information to some extent. They also used 'patch-seq' as one of the primary methods in the study. They were able to successfully identify the neurons based on the nine electrophysiological properties recorded. However, the study still lacks the use of the actual anatomical property in regards to the brain.

In another study [2], the authors used image processing to analyze the morphology of semi-isolated neurons. The study used images of a scale of 265 through a medium green optical filter. The images of the neurons are normalized, expanded, and enhanced using a lookup table. The analysis is done using mathematical morphological operations. The cell body shape is recognized using geometrical parameters, and then the identified cell is assigned to one of the classes through a multi-valued hierarchical decision tree. The results achieved are then checked by experts, who setter the overall output of the study is said to be satisfactory. This can be attributed to the fact that the study uses mathematical limits to classify the cells instead of training the model on a labeled set of images of neurons.

Another study [16] proposed a novel algorithm based on a new way of feature vector extraction and a deep learning method, which we call SpikeDeeptector. SpikeDeeptector considers a batch of waveforms to construct a single feature vector and enables contextual learning. The feature vectors are then fed to a deep learning method, which learns contextualized, temporal and spatial patterns, and classifies them as channels containing neural spike data or only noise. The study however is conducted on a very limited amount of data available. Which makes it difficult to scale.

In another paper, the researchers carefully develop an accurate and interpretable AD diagnosis and progression detection model. In the first layer, the model carries out a multi-class classification for the early diagnosis of AD patients. In the second layer, the model applies binary classification to detect possible MCI-to-AD progression within three years from a baseline diagnosis. The performance of the model is optimized with key markers selected from a large set of biological and clinical measures. The results of this study are very impressive, however, it too comes with its limitations that include the data that is used in the study is only the baseline data, Alzheimer's is a chronic disease so a time-series data would give better and more explainable results [17].

Another study that is done regarding a machine learning approach for the classification of electroencephalogram (EEG) recordings in dementia that happens due to chronic Alzheimers. In this study, the continuous wavelet transform (CWT) and Bispectrum features (BiS)were vectorized and used as a multimodal input for several different machine learning models. Autoencoder, multilayer perceptron, logistic regression, and support vector machine were used. The study yielded better results compared to only CWT and only BiS features. The study mentions that the BiS features are not sufficient in representing all the required information from the EEGs. However, there is much to take away from this study. We base our classification models on the models and techniques used in the study [18].

A new loss function is proposed by the researchers in [19]. The loss function was developed based on the objective function of the fuzzy C-mean algorithm. The paper experimentally demonstrates that the proposed method yielded good segmentation results on a clinical dataset.

The neural network architecture is based on the recurrent convolutional neural network proposed by Liang et al. in [20]. The network consists of eight convolutional layers, where the first five are recurrent convolutional layers. For each recurrent convolutional layer, T=3 time-steps were used, resulting in a feed-forward subnetwork with a depth of T+1=4. Each convolutional layer has a kernel size of 3 by 3, and it is followed by batch normalization and a Rectified linear unit (ReLU). In the final layer, a 3-channel softmax activation function is used, where each channel represents the probability of being classified as background, bone, or lesion. Through the study, this bureau network architecture had been proven very effective in classifying clinical data.

The researchers in [21] propose a novel Deep Learning model that is based on 'Transfer Learning' but that has been specifically trained for medical use from 3M publically available grayscale medical images, addressing classification tasks. The study introduces two Deep learning models 'Gray-edNet' and 'ColorMedNet' which is as observed from the name corresponding to training by gray and colored publicly available medical images. The model yields a comparable accuracy to a model solely trained on specific data in a large amount. A similar model can be applied to our research so as to act as a baseline for further research.

In our study, we are using multimodal machine learning to classify the images of neurons. Multimodal machine learning aims to build models that can process information from multiple modalities. Instead of focusing on a single feature.

4 Methodology

4.1 Baseline Model

We use a dense model to act as a baseline for our experiment. We initially used 3 hidden layers of sizes 32, 64, and 128. But after some experiments, the combination of 64, 128, and 256 layers performed better and hence this combination was used in all the further experiments. The hyperparameters are optimally chosen through experiments. By iterating over different combinations and plotting the loss curves. We train the model using the identical training data, then validate them over the same data. This gave us the consistency to compare the various loss curves and accuracies. The models are using the 'adam' optimizer with a learning rate of 0.001. The output layer is using 'Softmax' as its activation function. The full list of parameters is in Table 4.1

Table 4.1: Base Line model architecture hyperparameters

Hyperparameters	**Values**
Hidden Size	64, 128, 256
Activation Function	Rectified Linear Unit (Relu), Softmax
Optimization Function	Adam optimizer
Loss	Sparse Categorical Crossentropy
Metrics	Accuracy
Epochs	30
Batch Size	64
Learning Rate	0.001

4.2 Convolution Neural Network

A Convolutional Neural Networks (CNN) are computational processing systems that are heavily inspired by the way the biological nervous system operates. These comprise neurons that self-optimize through learning. Each neuron will receive input and performs an operation such as a scalar product or a non-linear function [22]. A CNN's are comprised of three main layers, these are convolutional layers, pooling layers, and fully connected layers. The convolutional layer will determine the output of neurons that are connected to local regions of the input through the calculation of the scalar product between their weights and the region connected to the input volume. The pooling layer will then simply perform downsampling along with the spatial dimensionality of the given input, further reducing the number of parameters within that activation. The fully-connected layers will then perform the same duties found in standard Artificial Neural Networks (ANNs) and attempt to produce class scores from the activations, to be used for classification.

We use a simple convolutional neural network for the experiment. The idea was to start from the baseline and steadily increase the computational complexity so as to manage the limited resources available from Google Colab. The CNN initially used in the experiment has a simple architecture with three hidden layers of the same size as the baseline model i.e., 64, 128, 256. The CNN includes the convolutional layer with a kernel size of 5x5 and a stride of 1. The padding on the image is kept 'same'. It also includes the max-pooling layer with a pool size of 2x2. The output layer includes a dense layer with the activation function set to 'softmax' similar to the baseline. A list of all the parameters for the initial CNN is given in Table 4.2.

Table 4.2: CNN model hyperparameters

Hyperparameters	**Values**
Hidden Size	64, 128, 256
Activation Function	Rectified Linear Unit (Relu), Softmax
Optimization Function	Adam optimizer
Loss	Sparse Categorical Crossentropy
Metrics	Accuracy
Epochs	30
Batch Size	64
Learning Rate	0.001
Kernel Size	5 x 5
Stride	1
Padding	Same
Pool Size	2 x 2

We implemented early callback in the next step so as to avoid overfitting of the model. The overall architecture was kept the same, with the only difference being the early callback function.

The model was then changed into a model with 4 hidden layers of sizes 64, 128, 512, and 1024. The number of increased layers increases the computational cost but also gives better results. L2 regularisation is also added to the convolutional layers in the later stages.

5 Experimental Setup

The project was to be started in Matlab using a pre-built Convolutional Neural Network (CNN). The data was to be imported from Google Drive, where the researchers from SAHMRI uploaded it. All of the mentioned steps were to be performed on a local machine, an Apple MacBook Pro, with an M1 chip and 8GBs of memory.

However, the limitation of Google Drive being unable to wholly download any files above 2 Gigabytes, prevented us from using the data locally. So after extensive attempts in trying to work on the local machine, by using Google APIs and Constructing Desktop apps using Google Cloud with Google's OAuth, we finally decided to use Google Colab for running the experiment. Google Colab was the best available option as a notebook in Colab is reproducible in almost any device with an internet connection regardless of its internal specifications.

We used the 'drive' module from 'google.colab' to connect the Google Drive to the Colab notebook. The entirety of the project is completed in python, as a jupyter notebook format in the Google Colab.

5.1 Data

The project uses the data from the researchers at the South Australian Health and Medical Research Institute (SAHMRI). The data was a multimodal compilation of the various behaviors of neurons. The neurons are bathed in an extracellular solution that mimics the cerebrospinal fluid of the human brain, which is commonly called artificial cerebrospinal fluid (ACSF). The patch-clamp pipette is filled with an intracellular solution and the neurons that express synapsin are selected. The electrical activity of the neurons is recorded after the pipette and the neuron forms a 'seal' and the membrane of the neuron breaks.

After the above step, the 'Test Pulse' is recorded of the neuron. This helps to monitor the passive properties of the neuron under study. This recording is generally used to run quality controls on our recordings. Cells with a high access resistance will generally be excluded (e.g. > 40). The next protocol is the IVVC at -60mV. This is used to observe the ionic currents from the cell under study. The voltage clamp method is used here, this allows us to hold the neuron at the set membrane potentials using the step current of -5 mV to +65 mV. This also allows us to monitor the changes in the ion flow across the voltage ion channels.

The action potential firing of the neurons is then tested, and it is performed in the current clamp. This allows us to hold the cells at around 60 mV and inject current to depolarise them. When reaching a threshold, the neurons will fire. This is the variable between the disease and the control groups. The synaptic input recordings were looked at in the final step. The recordings are from the AMPA receptors which bind an amino acid called glutamate. These are responsible for the most excitatory synaptic inputs in the central nervous system.

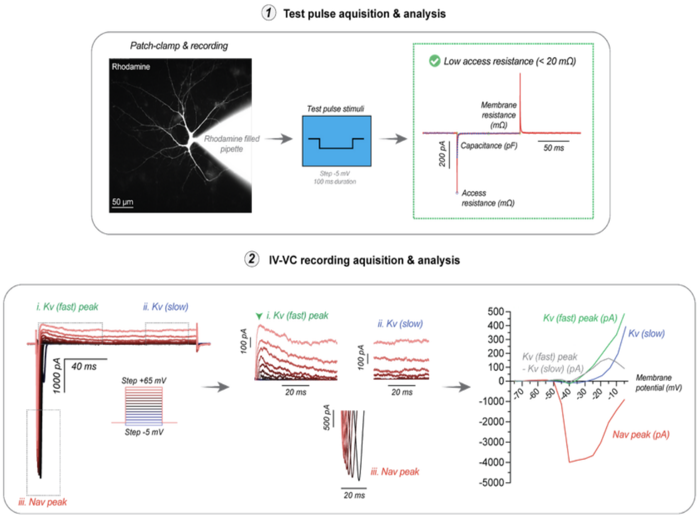

Figure 5.1: Recordings taken from the neurons

For the initial experiments, we used only the raw images of the neurons under study and considered their labels i.e., disease or control. The images are taken at the end of patch clamping. Once the cell membrane is broken, the neurons are perfused with a 'rhodamine' dye which is a component of the intracellular solution that we used for the patch clamping. This helps us to visualize the single neuron that we have patched. These are also the images that would later be used for morphological reconstruction.

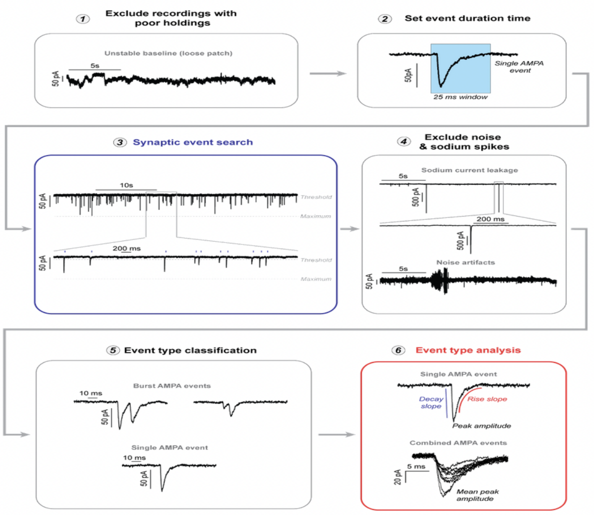

Figure 5.2: Steps in the analysis of the neurons

For the initial experiments, we used only the raw images of the neurons under study and considered their labels i.e., disease or control. The images are taken at the end of patch clamping. Once the cell membrane is broken, the neurons are perfused with a 'rhodamine' dye which is a component of the intracellular solution that we used for the patch clamping. This helps us to visualize the single neuron that we have patched. These are also the images that would later be used for morphological reconstruction.

The images were loaded into the notebook, using the Matplotlib library. The size of each image was 2048 by 2048 which is extremely large for the Colab to handle as we get limited memory access. Each image averaged around 8 Megabytes, this caused the notebook to crash and lose all the progress and variables. The images were reduced in size from 2048×2048 to 256×256 using the Image module from the Pillow (PIL) library in Python. Around 700 images were used in the training which included equal proportions of both the 'control' and 'diseased' labels. 60 images were used as the validation.

5.2 Models

5.2.1 The Baseline Model

For the experiment was set as a model with ≥ three hidden layers and one output dense layer. The detailed model hyperparameters are listed in table1. This model has 4,236,226 trainable parameters. This model gave us a training accuracy of 63% and the validation accuracy of 53%. This model and the results were noted to act as a baseline for all our further experiments. This model was kept computationally simple so as to be quick and give us an idea of how much computation to expect to achieve the ideal and expected results.

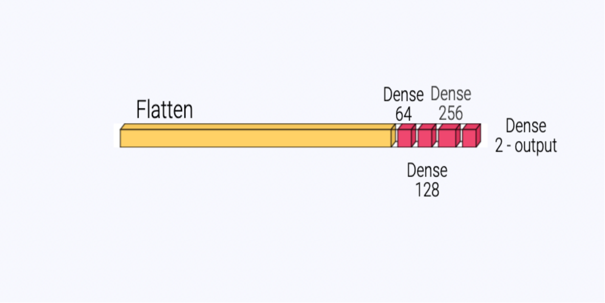

Figure 5.3: The model summary of the baseline model

5.2.2 The Convolutional Neural Network

A convolutional neural network was used on the same training and validation data. The architecture of the model includes three 2D convolution layers with sizes of 64, 128, and 256 and max-pooling layers. This was followed by a 'flatten' and 'dense' layer. The number of the trainable parameters in this model was 7,482,626.

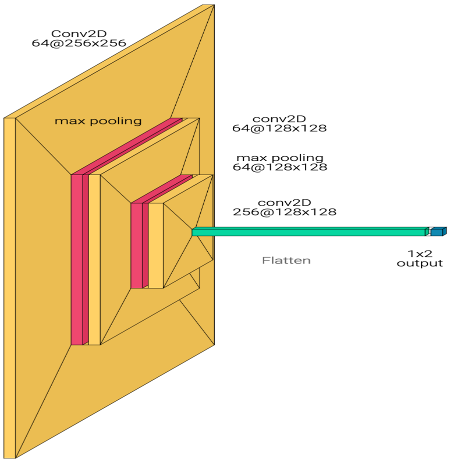

Figure 5.4: The model summary of the CNN

We experimented with different regularization parameters, optimizers, and activation functions. The L2 regularization was added to each of the convolutional layers with a scale of 0.01. Along with that, we modified the early callback patience from 6 to 8. This increased the overall performance and also saved on the computation cost of running the full 30 epochs cycle.

6 Results

In this section we will show the performance of our Dense and CNN models. The performance is calculated using the F1 score [23]. The F1 score is the harmonic mean of precision and recall. The F1 score is bounded to [0, 1], where 1 represents maximum precision and 0 illustrates 0 precision/recall.

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

TP = True Positive
FP = False Positive
FN = False Nagative

precision = $\frac{TP}{TP+FP}$

recall = $\frac{TP}{TP+FN}$

Along with the F1 score, we also make use of the ROC curves in our experiment, as Omar et. al. suggests that ROC curves are a useful tool to facilitate classifier development [24]. Confusion matrix and Loss and accuracy versus epochs plots are also used as metrics of evaluation. Through the experiments, we carefully select the hyperparameters that best fit the problem. A detailed summary of all the hyperparameters used in the models is given in Tables 4.1 and 4.2.

We found that the CNN achieved the best performance when used with early callback and l2 regularisation. We also found that the CNN models overall outperformed the Dense models with ease. This can be contributed to the fact that CNN detects the valuable features from a training image, it learns the distinctive features for each of the classes by itself.

6.1 The Baseline

The baseline model as discussed previously is a dense net with architecture as shown in the figure 5.3. The baseline model gave us a maximum validation accuracy of 83% however, after the set of 30 epochs, the validation accuracy achieved was 73%, with the training accuracy of 83%. The baseline gave us an F1 score of 0.666. This score represents a good enough value. It's greater than 50%, so we can say it's not a random prediction and the model fit the data, although it's not high enough to be used in any practice.

Table 6.1: Metrics from the baseline model

Metrics	**Values**
Training Accuracy	63.3%
Training Loss	43.5%
Validation Accuracy	53.3
Validation Loss	69.9%
F1 Score	0.66

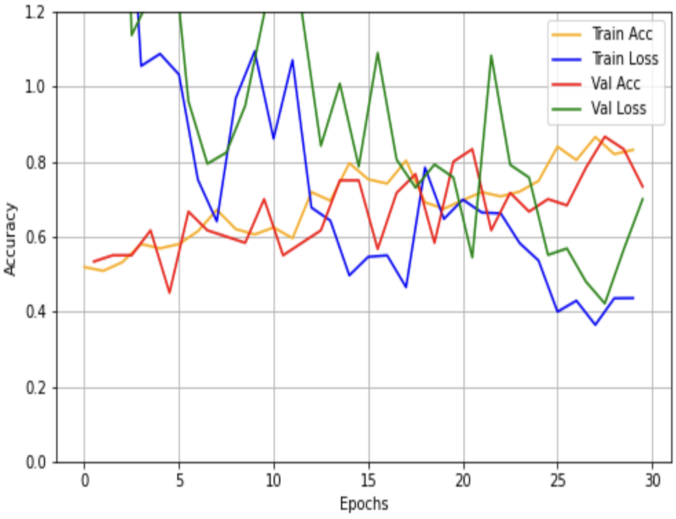

Figure 6.1: learning curve for the Baseline model

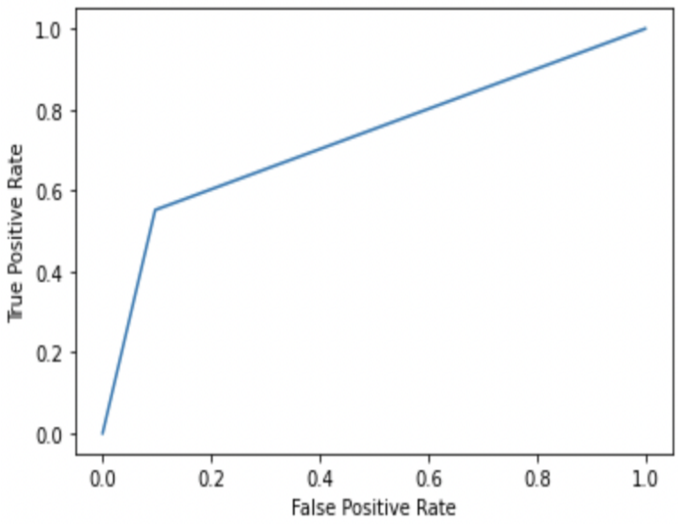

Figure 6.2: ROC curve for the Baseline model

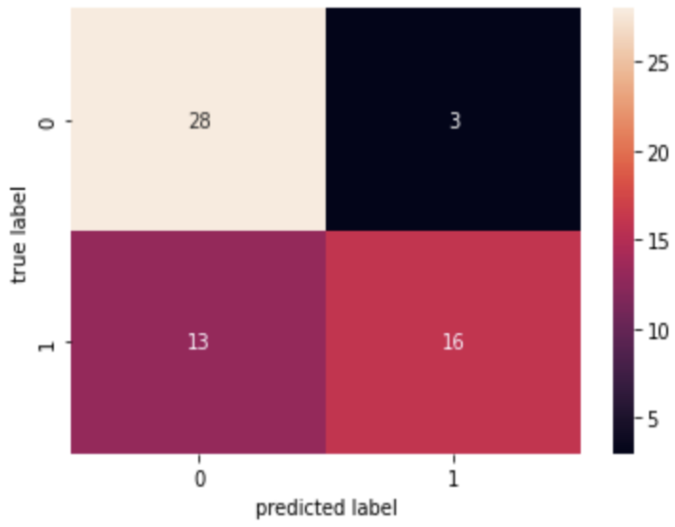

Figure 6.3: Confusion Matrix for the Baseline Model

6.2 The Convolutional Neural Network

The CNN used in the experiment is discussed earlier in Table 4.2 and Fig. 5.4. The initial test on the CNN is used without using the early callback function which resulted in the overfitting of the model. Overfitting is basically when a model is unable to learn effectively. It happens when we work too hard to find the best fit in the training data so the model fits the noise as well. This will then lead to poor performance on the validation or the test data. The model after the set of 30 epochs gave a validation accuracy of 93% and a training accuracy of a 100%. The model gave an F1 score of 0.952.

Metrics	Values
Training Accuracy	100%
Training Loss	0.4%
Validation Accuracy	93.3 %
Validation Loss	48%
F1 Score	0.952

Table 6.2: Metrics from the CNN

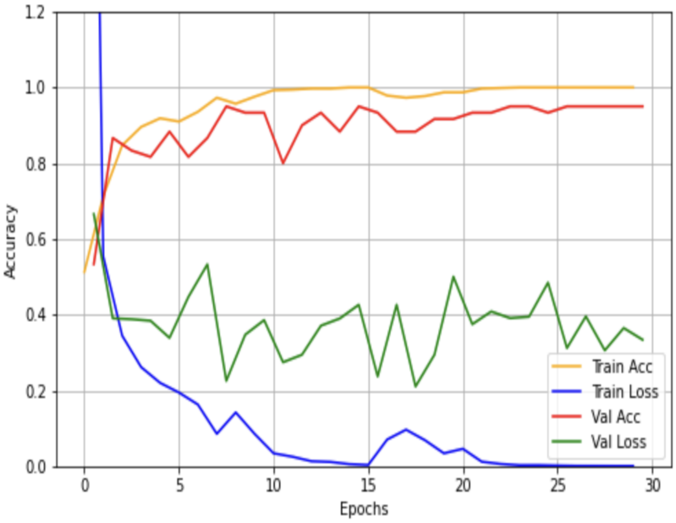

Figure 6.4: Learning curve for the CNN

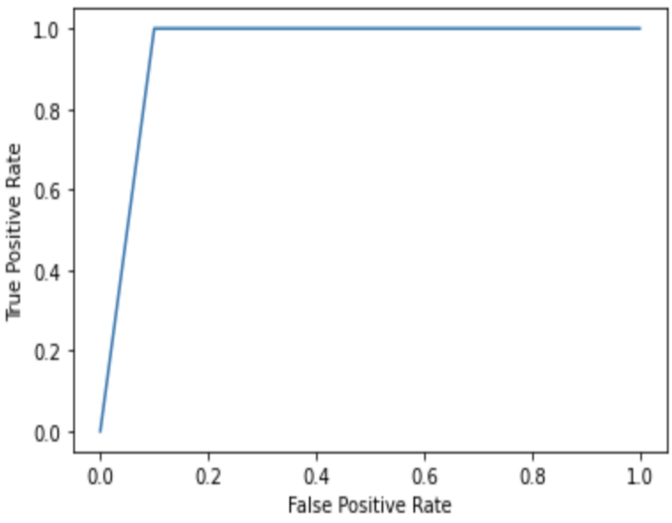

Figure 6.5: ROC curve for the CNN

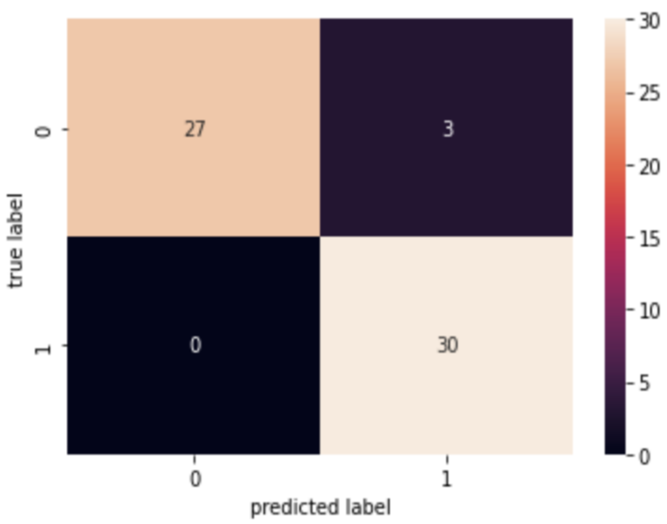

Figure 6.6: Confusion Matrix for the CNN

Table 6.2 shows the result for the different matrices for the CNN. The table shows that we achieved a 100% accuracy on the training data with minimal error. However, this can also constitute as overfitting.

6.3 CNN with Early Callback

As we can see from Fig. 6.4, the model overfits in the end as the validation loss increases. To counter this we implemented 'Early Callback' with a 'patience' of 6 to monitor the validation accuracy. The callback is a piece of code that prevents our model to surpass that level of optimal fit and overtrain or overfit. This function monitors the losses generated on each epoch and stops the training according to the 'patience' value set.

With the early callback function in place, the model ran for 17 epochs and was trained to a validation accuracy of 95% and a training accuracy of 100%. The model got an F1 score of 0.952, this is similar to the base model but in fewer training epochs i.e. 17, and hence no overfitting and lower validation error.

Metrics	Values
Training Accuracy	100%
Training Loss	0.2%
Validation Accuracy	95 %
Validation Loss	42%
F1 Score	0.952

Table 6.3: Metrics from the CNN with early stopping

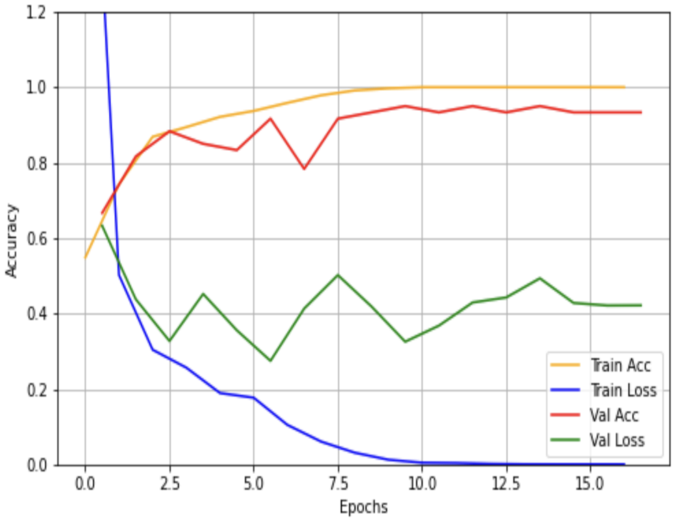

Figure 6.7: Learning curve for CNN with Early Callback

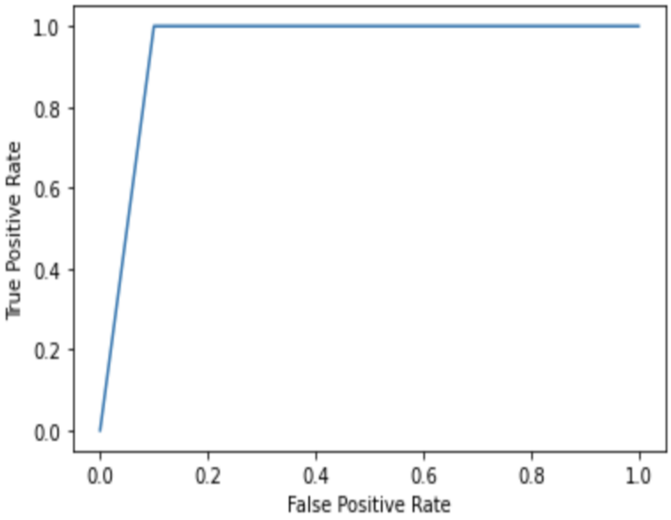

Figure 6.8: ROC plot for CNN with Early Stopping

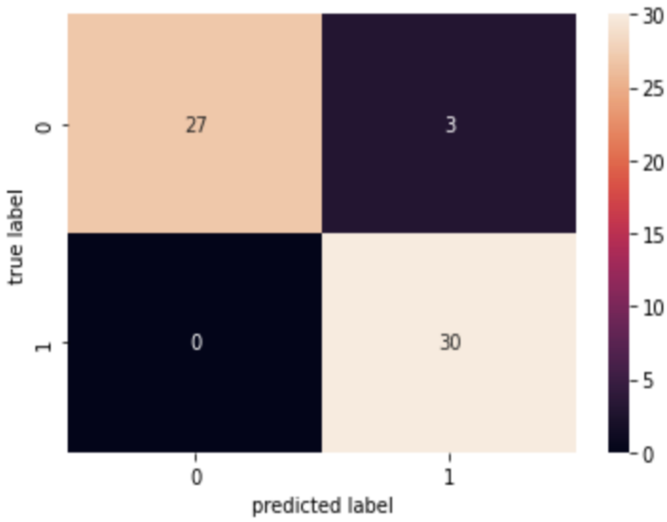

Figure 6.9: Confusion matrix for CNN with early stopping

6.4 CNN with L2 regularization

The model was performing very well, but in an attempt to improve the performance further and to prevent overfitting, we decided to add L2 regularization to the convolutional layers of the model. The L2 was applied to the layers with a scale of 0.01. With the regularization in place, the performance bumped up further providing a validation accuracy of 96.66 and a training accuracy of 100%, in just 14 epochs from the earlier 17. This model gave an F1 score of 0.967. This value of the F1 score is very good. A 0.967 F1 score shows that we have a very little amount of false-positive and false-negative predictions.

Metrics	Values
Training Accuracy	100%
Training Loss	0.2%
Validation Accuracy	98 %
Validation Loss	38.3%
F1 Score	0.967

Table 6.4: Metrics from the CNN with L2 regularization

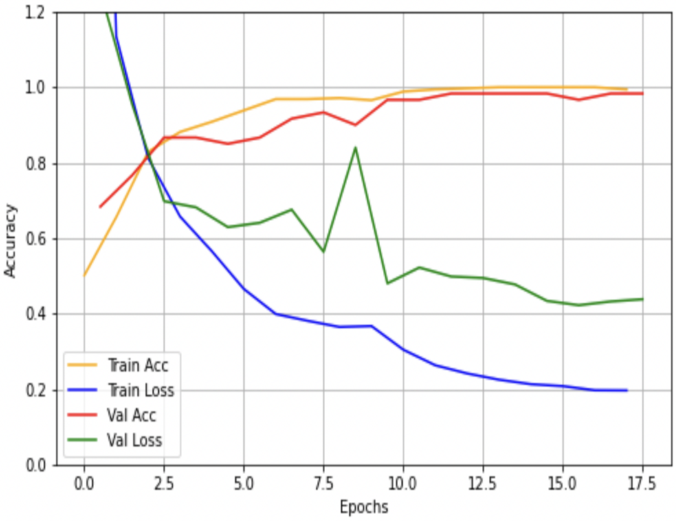

Figure 6.10: Learning curve for CNN with L2 regularization

The figure 6.9 shows the confusion matrix for the predictions. The matrix shows that there are two false negative cases and zero false positive cases. This is a very impressive result as we get no false positive cases. However, we must set some protocol for the false positive cases, this is discussed more in the 'Discussion' section later on in the paper.

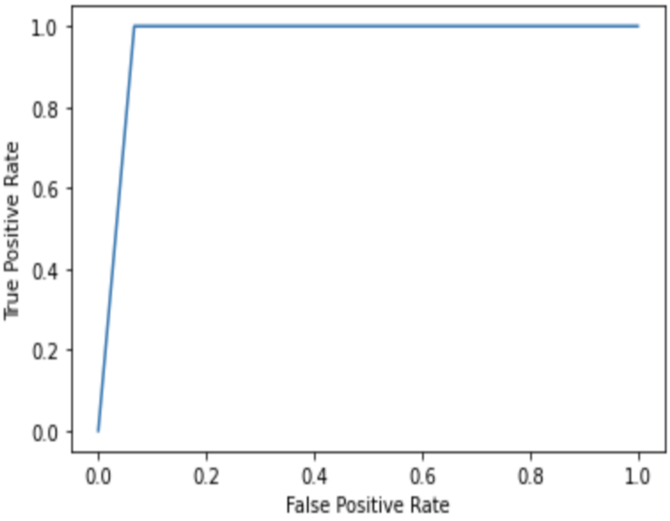

Figure 6.11: ROC plot for CNN with Early Stopping

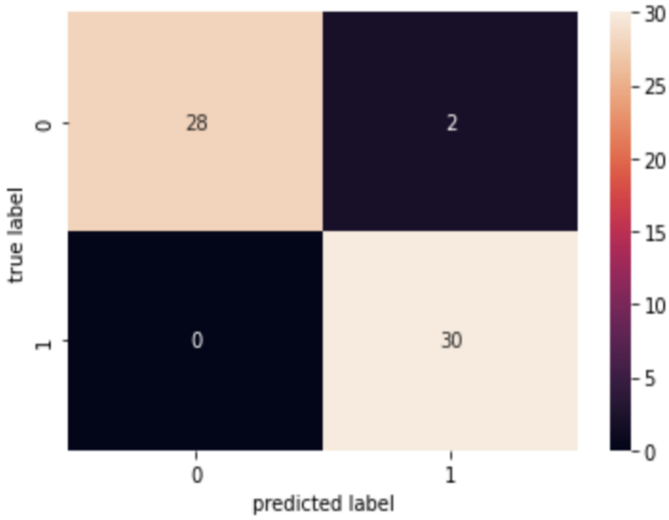

Figure 6.12: Confusion matrix for CNN with early stopping

7 Discussion

The proposed CNN model performed extremely well on the given data. This is a significant step forward from the research in [2], where the model looks directly at the images of the neurons free from mathematical limitations in assigning shapes to the neurons. This study also provides a computationally inexpensive solution as it is entirely performed with limited resources available from the free base version of Google Colab. The solution proposed is also scalable, and usable in real cases after considering some of the techniques suggested below.

We propose the use of Transformers as suggested in [25]. The researchers show that a pure transformer when applied directly to the sequence of image patches can perform very well on image classification tasks. The use of transfer learning is also a notable step to explore, one of the issues in that is that the models are usually trained on general image data and not medical image data, which can cause the accuracy to go down. However, this is an interesting domain to explore because for the cases where transfer learning works it produces highly accurate results.

Something to note is that models with more parameters have better predictive performance, however, these models require more mathematical operations, and thus longer time [26]. Therefore, model selection is a valuable step that will determine the success of the real-life application of the study. The development of the model and validation is a necessary part of the translation process, but the performance of a model is not sufficient to create an actual impact. There are many other challenges in real-world scenarios as no model is fully accurate. Hence, the system should also be designed to be useful in case of failure such as false positives [26].

The current model only operates on the raw image data of the neurons, however, that data provided by SAHMRI has a lot more features that should be added so that the model can operate on multi-modal inputs. This is the next step forward for the study to be used in real-life examples where there are a number of factors that can affect the conditions. This increase in the modality and the results from those experiments would give us substantial insight into how the diseases of the brain operate on a neural level.

8 Conclusion

In this paper, we proposed the use of machine learning models to classify the images of neurons, so as to help identify and diagnose diseases of the brain. To this end, we have successfully trained a CNN using early callback and l2 regularization on the data provided by The South Australian Health and Medical Research Institute. Our experiments have shown that our model outperformed the respective baseline models. We also showed that our CNN achieved the best results compared to other models on the same data. Our model is capable of successfully classifying the images of neurons even when trained on a small batch of the data.

9 Code Availability

The code used in the Google Colab is converted to a jupyter notebook with .ipynb extension and is uploaded to GitHub. It is accessible here.

References

[1] A. Woodruff. "What is a neuron?" Queensland Brain Institute. (2019), [Online]. Avail-

able: `https://qbi.uq.edu.au/brain/brain-anatomy/what-neuron` (visited on 06/07/2022).

[2] M. Masseroli, A. Bollea, and G. Forloni, "Quantitative morphology and shape classification of neurons by computerized image analysis," *Computer methods and programs in biomedicine*, vol. 41, no. 2, pp. 89–99, 1993.

[3] B. G. Kornreich, "The patch clamp technique: Principles and technical considerations," *Journal of Veterinary Cardiology*, vol. 9, no. 1, pp. 25–37, 2007.

[4] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of physiology*, vol. 117, no. 4, p. 500, 1952.

[5] E. Neher and B. Sakmann, "Single-channel currents recorded from membrane of denervated frog muscle fibres," *Nature*, vol. 260, no. 5554, pp. 799–802, 1976.

[6] O. P. Hamill, A. Marty, E. Neher, B. Sakmann, and F. J. Sigworth, "Improved patch-clamp techniques for high-resolution current recording from cells and cell-free membrane patches," *Pflügers Archiv*, vol. 391, no. 2, pp. 85–100, 1981.

[7] D. P. Wellis, L. J. DeFelice, and M. Mazzanti, "Outward sodium current in beating heart cells," *Biophysical journal*, vol. 57, no. 1, pp. 41–48, 1990.

[8] C. Antzelevitch, S. Sicouri, S. H. Litovsky, A. Lukas, S. C. Krishnan, J. M. Di Diego, G. A. Gintant, and D.-W. Liu, "Heterogeneity within the ventricular wall. electrophysiology and pharmacology of epicardial, endocardial, and m cells.," *Circulation research*, vol. 69, no. 6, pp. 1427–1449, 1991.

[9] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[10] J. Sukanya and S. Kumar, "Applications of big data analytics and machine learning techniques in health care sectors," *International Journal of Engineering and Computer Science*, vol. 6, no. 7, pp. 21 963–21 967, 2017.

[11] L. S. Wylezinski, J. D. Gray, J. B. Polk, A. J. Harmata, and C. F. Spurlock, "Illuminating an invisible epidemic: A systemic review of the clinical and economic benefits of early diagnosis and treatment in inflammatory disease and related syndromes," *Journal of Clinical Medicine*, vol. 8, no. 4, p. 493, 2019.

[12] I. B. Levitan, I. B. Levitan, L. K. Kaczmarek, *et al.*, *The neuron: cell and molecular biology.* Oxford University Press, USA, 2002.

[13] P. R. Solomon and C. A. Murphy, "Early diagnosis and treatment of alzheimer's disease," *Expert Review of Neurotherapeutics*, vol. 8, no. 5, pp. 769–780, 2008.

[14] C. Bardy, M. Van Den Hurk, B. Kakaradov, J. Erwin, B. Jaeger, R. Hernandez, T. Eames, A. Paucar, M. Gorris, C. Marchand, *et al.*, "Predicting the functional states of human ipsc-derived neurons with single-cell rna-seq and electrophysiology," *Molecular psychiatry*, vol. 21, no. 11, pp. 1573–1588, 2016.

[15] X. Chen, K. Zhang, L. Zhou, X. Gao, J. Wang, Y. Yao, F. He, Y. Luo, Y. Yu, S. Li, *et al.*, "Coupled electrophysiological recording and single cell transcriptome analyses revealed molecular mechanisms underlying neuronal maturation," *Protein & cell*, vol. 7, no. 3, pp. 175–186, 2016.

[16] M. Saif-ur-Rehman, R. Lienkämper, Y. Parpaley, J. Wellmer, C. Liu, B. Lee, S. Kellis, R. Andersen, I. Iossifidis, T. Glasmachers, *et al.*, "Spikedeeptector: A deep-learning based method for detection of neural spiking activity," *Journal of neural engineering*, vol. 16, no. 5, p. 056 003, 2019.

[17] S. El-Sappagh, J. M. Alonso, S. Islam, A. M. Sultan, and K. S. Kwak, "A multilayer multimodal detection and prediction model based on explainable artificial intelligence for alzheimer's disease," *Scientific reports*, vol. 11, no. 1, pp. 1–26, 2021.

[18] C. Ieracitano, N. Mammone, A. Hussain, and F. C. Morabito, "A novel multi-modal machine learning based approach for automatic classification of eeg recordings in dementia," *Neural Networks*, vol. 123, pp. 176–190, 2020.

[19] J. Chen, Y. Li, L. P. Luna, H. W. Chung, S. P. Rowe, Y. Du, L. B. Solnes, and E. C. Frey, "Learning fuzzy clustering for spect/ct segmentation via convolutional neural networks,"

Medical physics, vol. 48, no. 7, pp. 3860–3877, 2021.

[20] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3367–3375.

[21] L. Alzubaidi, J. Santamarıa, M. Manoufali, B. Mohammed, M. A. Fadhel, J. Zhang, A. H. Al-Timemy, O. Al-Shamma, and Y. Duan, "Mednet: Pre-trained convolutional neural network model for the medical imaging tasks," *arXiv preprint arXiv:2110.06512*, 2021.

[22] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[23] W. F. Hendria, Q. T. Phan, F. Adzaka, and C. Jeong, "Combining transformer and cnn for object detection in uav imagery," *ICT Express*, 2021.

[24] L. Omar and I. Ivrissimtzis, "Using theoretical roc curves for analysing machine learning binary classifiers," *Pattern Recognition Letters*, vol. 128, pp. 447–451, 2019.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[26] P.-H. C. Chen, Y. Liu, and L. Peng, "How to develop machine learning models for healthcare," *Nature materials*, vol. 18, no. 5, pp. 410–414, 2019.