

Mini Project Report on

**Application of Multi-Layer Perceptrons in Hardware
Detection of Brain Tumors from MRI Scans**



**Submitted for mini project of Digital Circuit Design laboratory as partial
fulfillment of B.Tech Course**

By:

1. G.S. Ramanujam- 23ECB0A01
2. B. Karthik - 23ECB0A02
3. N. Sasank- 23ECB0A03

Under the guidance of

1. Dr. P. Prithvi
2. Dr. Shuvam Pawar

Department of Electronics and Communication Engineering

NITW

Abstract: -

The early detection of brain cancer significantly influences treatment efficacy and patient survival rates. This project presents an innovative approach using Multi-Layer Perceptron (MLPs) to analyse MRI brain scans, effectively classifying them into categories with or without tumours.

Developing such systems from low level components (I.e. on FPGA boards/Basic languages such as Verilog/VHDL/SystemVerilog) helps increase the speed and decreases computation power required as well as decrease the power consumption

Contents: -

1. List of tables: -

None

2. List of figures: -

- Fig1.Multi-Layer Perceptron – Page 3
- Fig2.Relu function – Page 4
- Fig3.Nearest neighbour interpolation – Page 4

3. List of abbreviations: -

MLP -> multi-layer perceptron

SV -> SystemVerilog

FPGA -> Field Programmable Gate Array

HDL -> Hardware Description Language

VHDL ->Very High-Speed Integrated Circuit Hardware Description Language

Introduction: -

Deep learning has changed the recent era and is being used in various different fields for various types of applications (whether it be medical, research, scientific, etc etc)

This project focuses on developing a multi-layer perceptron neural network on hardware.

Building machine learning algorithms from scratch helps improve the computation efficiency and decrease power consumption.

Problem Statement: -

To develop a multi-layer perceptron in systemVerilog with a focus on medical applications

(Detection of brain tumours from MRI images)

Motivation: -

Training doctors is a costly process that requires significant resources, and India faces a shortage of medical facilities. This system aims to mitigate these challenges by providing a simple and automated solution for detecting brain tumours from MRI images.

Implementing this project using a basic hardware language such as systemVerilog not only enhances processing speed but also reduces the computational power and energy consumption required for this task.

This project has potential in increasing the medical facilities available in our country.

Novelty: -

- Low power and low computation implementation unlike traditional implementations.
- Requires much less power and computation power due to implementation on FPGA/Basic electronic languages
- Can help junior doctors with training
- Can help give doctors a second opinion during judgement
- More accurate than traditional methods such as histogram generation etc.

Objectives: -

- To design and implement a low power, low computation required multi-layer perceptron with other extra features for the specific problem statement.
- To get a highly accurate implementation which can aid medical industry in various ways (Such as the ones mentioned before (i.e. helping junior doctors, getting second opinion etc))

Theory: -

- **Multi-Layer perceptron: -**

Multi-layer perceptrons are feed forward neural networks containing generally 3 or more layers.

They are fully connected in nature (i.e. each node in previous layer is connected to every other node in the next later)

Each node in multi-layer perceptron contains a value that is obtained as a summation of the activation of the nodes in the previous layer multiplied with a certain value.

Multi-layer perceptrons are the most popular type of networks used in deep learning.

Generally, non-linear functions are used in multi-layer perceptrons.

Multi-layer perceptrons can be trained mathematically via back propagation.

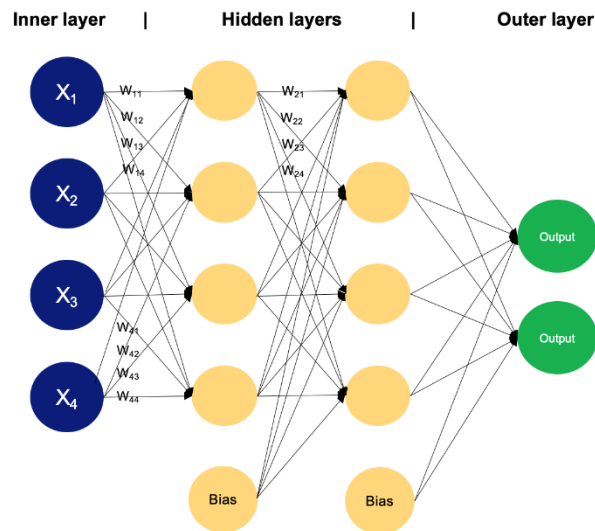


Fig1. Multi-Layer perceptron

Source: -DataCamp

- **Activation functions: -**

An activation function is a function in a neural network that calculates the outputs based on the input.

Generally, ReLU is used as the activation function for MLPs.

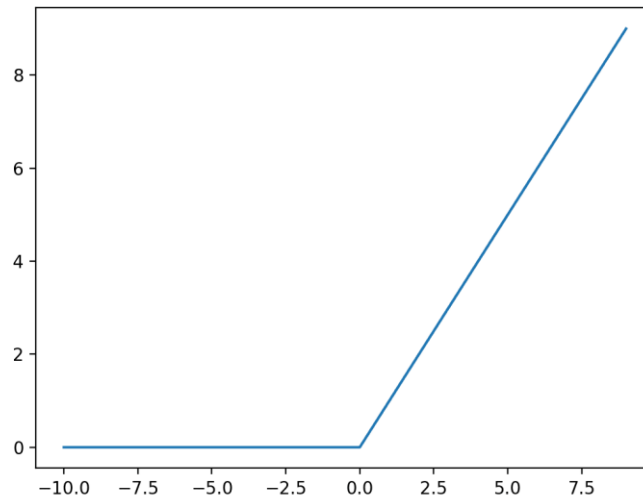


Fig2.ReLU function

(Source: - Machine learning Mastery)

- **Feature map generation: -**

Feature map (also called activation map) is the output array of images generated when an array of kernels/filters is passed over the original image.

Feature maps help highlight various elements of the image such as edges, diagonals, lights and shadows, certain shapes etc.

In our application, ideally, the generated kernels for the feature map will be able to determine the shape of various types of brain tumours.

- **Nearest neighbour interpolation:** -

To put simply, Nearest neighbour interpolation is an image resize technique. It works by finding the position of the nearest neighbouring pixel and setting the intensity value of the image to be scaled to that value.

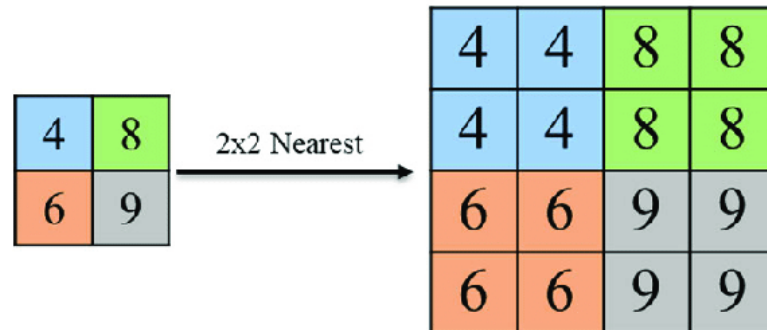
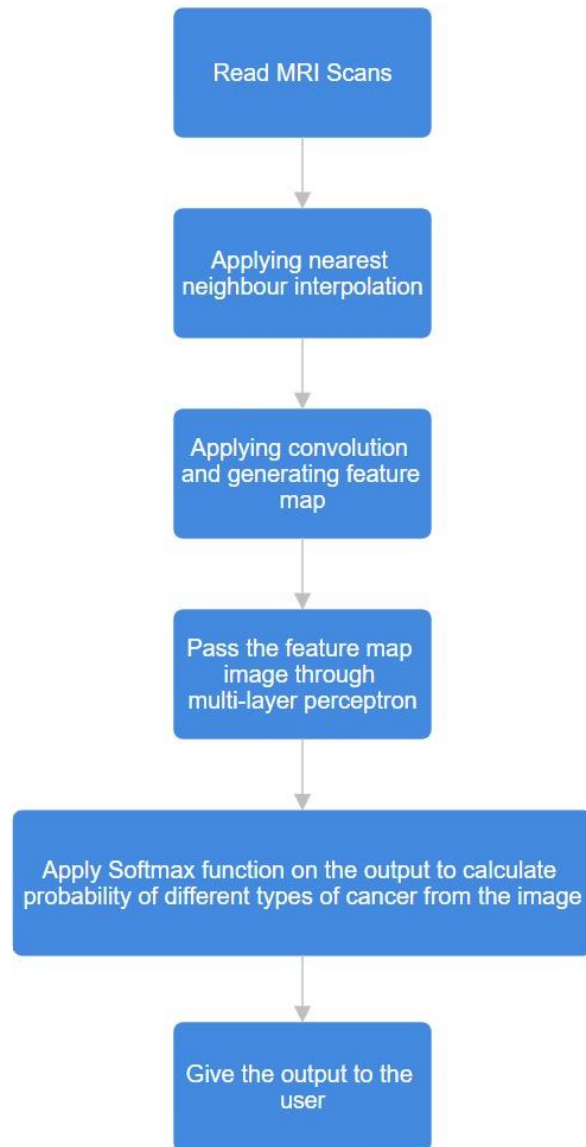


Fig3. Nearest neighbour interpolation

(Source: - ResearchGate)

Workflow: -

1. Take MRI scan images.
2. Resize the images (By applying nearest neighbour interpolation).
3. Apply convolution and generate feature map. (optional)
4. Pass the feature map/image through multi-layer perceptron.
5. Apply SoftMax function on the output to calculate probability of different types of cancer from the image.
6. Give the output to the user.



Results and conclusions: -

- Reached moderately high accuracy in detecting type of cancer from MRI image.
- Reached very high accuracy in determining whether the person has cancer or not from the MRI image
- Unlike traditional methods which have a fixed accuracy, the accuracy of this system will increase as it gets more and more data over time.

References: -

- Neural networks and deep learning from 3Blue1Brown
<https://youtu.be/aircAruvnKk?feature=shared>
- System Verilog reference from chipVerify
<https://www.chipverify.com/tutorials/systemverilog>
- Mathematical algorithm inspiration etc from pyTorch documentation
<https://pytorch.org/docs/stable/index.html>
- Nearest neighbour interpolation from Wikipedia
https://en.wikipedia.org/wiki/Nearest-neighbor_interpolation