Report

Of

Six Week Industrial Internship

On

**Hand gesture Recognition**

**As stone, paper and scissor**

at

**NEONEX TECHNOLOGY**

**Submitted By**

Student Name: **Dheeraj Nagpal**

Branch (CSE)

Semester: 5th Sem

Roll No: **11701025**



**Department of Computer Engineering**

**Punjabi University, Patiala**

**2017-2021**

Table of Contents

Table of Contents

[PREFACE 4](#_Toc24968750)

[ACKNOWLEDGMENT 5](#_Toc24968751)

[Certificate 6](#_Toc24968752)

[Company’s Profile 7](#_Toc24968753)

[Company details 7](#_Toc24968754)

[Website 7](#_Toc24968755)

[Introduction To Technology Used In Project 8](#_Toc24968756)

[Python 8](#_Toc24968757)

[History of Python 9](#_Toc24968758)

[Python Features 10](#_Toc24968759)

[Python Overview: 12](#_Toc24968760)

[Standard Data Types: 13](#_Toc24968761)

[Python Lists: 13](#_Toc24968762)

[Python Dictionary: 13](#_Toc24968763)

[Libraries Used: 14](#_Toc24968764)

[pandas: 14](#_Toc24968765)

[NumPy: 15](#_Toc24968766)

[Matplotlib: 15](#_Toc24968767)

[Tensorflow: 17](#_Toc24968768)

[Keras: 19](#_Toc24968769)

[ENVIRONMENT USED: 20](#_Toc24968770)

[Google Colab: 20](#_Toc24968771)

[Artificial intelligence 21](#_Toc24968772)

[Machine learning 21](#_Toc24968773)

[Deep learning 22](#_Toc24968774)

[Activation Function In deep learning: 22](#_Toc24968775)

[Training in Neural Network: 23](#_Toc24968776)

[Error: 23](#_Toc24968777)

[Forward Propagation: 24](#_Toc24968778)

[Back Propagation 24](#_Toc24968779)

[Regularisation: 25](#_Toc24968780)

[Optimization: 25](#_Toc24968781)

[Why Deep Learning: 25](#_Toc24968782)

[Artificial Neural Networks(ANN): 27](#_Toc24968783)

[ABOUT PROJECT 33](#_Toc24968784)

[VISION: 33](#_Toc24968785)

[DESCRIPTION: 33](#_Toc24968786)

[DATASET: 33](#_Toc24968787)

[DIRECTORY STRUCTURE: 34](#_Toc24968788)

[Layer in the model: 34](#_Toc24968789)

[Model Train and Test Accuracy: 35](#_Toc24968790)

[USER INTERFACE: 35](#_Toc24968791)

# PREFACE

In the field of Computer Engineering, the term project refers to a “Computerized work” that is completely performed using a computer.

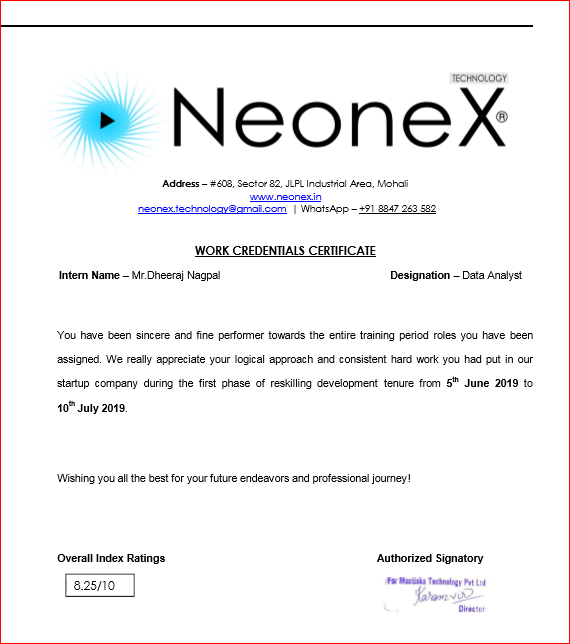
Practical Training or internship for a period of 6 weeks is a part of the Degree after 2nd year (B. Tech Computer Science and Engineering) according to the syllabus worked out by the Punjabi University, Patiala to which my department is affiliated. I did my 6-week Internship at “Neonex Technology”, sector 82 Mohali.

This report is a summary of what I learned and performed there. This project deals with every aspect of Python. It aims at providing a thorough base and understanding of various latest trends and techniques in Python.

# ACKNOWLEDGMENT

It is a great pleasure to present this report on the project named “Recognition of Hand gestures” undertaken by me as part of my B. Tech (CSE) curriculum. We are thankful to our college i.e. University College of Engineering, Punjabi University, Patiala Campus for offering us such a wonderful challenging opportunity and we express our deepest thanks to all coordinators, for providing all the possible help and assistance and their constant encouragement. It is a pleasure that we find ourselves penning down these lines to express our sincere thanks to the people who helped us along the way in completing our project. We find inadequate words to express our sincere gratitude towards them. First and foremost we would like to express our gratitude towards our Mr. Karamvir Singh Rajpal(Director Neonex Technology) for placing complete faith and confidence in our ability to carry out this project and for providing us his time, inspiration, encouragement, help, valuable guidance, constructive criticism, and constant interest. They took a personal interest in spite of numerous commitments and busy schedules to help us complete this project. Without the sincere and honest guidance of our respected project guide, we would have not been to reach the present stage.

# Certificate



# Company’s Profile

* The company based in Mohali, an originally registered under "**Mastiska Technology Private Limited**" with a Ministry of Corporate Affairs of Government of India and later in 2019, it was acquired by **Innovative Group**.
* Aims to flourish the concept of Smart City, Global Village by driving the philosophy of emerging Multifacet, multidimensional and trendsetter startup with having brands like **‘SCODEL**’, ’ **T2OC**’, ‘[**At The Tech**](http://www.neonex.in/atthetech)’, ‘[**PortoCafe PLX**](http://neonex.in/plx)’, ‘[**S2FY**](http://neonex.in/s2fy)’, ‘ [**JauntBee**](http://www.neonex.in/jauntbee)’, ‘[**MRAW**](http://www.neonex.in/mraw.app)’.
* Its status is officially registered as SME of Group D under the Ministry of MSME of the Government of India.
* Our start-up culture is cool and vibrant where a creatively charged team of dynamic potential works for the best and hence we are a real problem solver.
* The company is a team of seasoned players who are equipped with proven skills in their forte and schematic approach. Hence a philosophy of delivering the best strives in our dedication and determination in our worldwide served client list.

## Company details

#608, Sector 82, JLPL Industrial Area, Mohali

## Website

<http://neonex.in/>

Company type: **Product Based Company**

# Introduction To Technology Used In Project



## Python

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python’s elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactic constructions than other languages.

* **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
* **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

### History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

### Python Features

Python's features include:

* **Easy-to-learn:** Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
* **Easy-to-read:** Python code is more clearly defined and visible to the eyes.
* **Easy-to-maintain:** Python's source code is fairly easy-to-maintain.
* **A broad standard library:** Python's bulk of the library is very portable and cross-platform compatible with UNIX, Windows, and Macintosh.
* **Interactive Mode:** Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* **Portable:** Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* **Extendable:** You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* **Databases:** Python provides interfaces to all major commercial databases.
* **GUI Programming:** Python supports GUI applications that can be created and ported to many system calls, libraries and Windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
* **Scalable:** Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below:

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* IT supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

### Python Overview:

A Python identifier is a name used to identify a variable, function, class, module, or another object. An identifier starts with a letter A to Z or a to z or an underscore (\_) followed by zero or more letters, underscores, and digits (0 to 9).

Python does not allow punctuation characters such as @, $, and % within identifiers. Python is a case sensitive programming language. Thus **Manpower** and **manpower** are two different identifiers in Python.

Here are following identifier naming convention for Python:

* Class names start with an uppercase letter and all other identifiers with a lowercase letter.
* Starting an identifier with a single leading underscore indicates by convention that the identifier is meant to be private.
* Starting an identifier with two leading underscores indicates a strongly private identifier.
* If the identifier also ends with two trailing underscores, the identifier is a language-defined special name.

### Standard Data Types:

Python has five standard data types:

* Numbers
* String
* List
* Tuple
* Dictionary

### Python Lists:

Lists are the most versatile of Python's compound data types. A list contains items separated by commas and enclosed within square brackets ([]).

### Python Dictionary:

Python 's dictionaries are hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs.

## Libraries Used:

### pandas:

**pandas** is a python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, **real-world** data analysis in Python. Additionally, it has the broader goal of becoming **the most powerful and flexible open-source data analysis/manipulation tool available in any language**. It is already well on its way toward this goal.

pandas is well suited for many different kinds of data:

* Tabular data with heterogeneously-typed columns, as in a SQL table or Excel spreadsheet
* Ordered and unordered (not necessarily fixed-frequency) time-series data.
* Arbitrary matrix data (homogeneously typed or heterogeneous) with row and column labels
* Any other form of observational/statistical data sets. The data actually need not be labeled at all to be placed into a pandas data structure.

### NumPy:

NumPy’s main object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In NumPy dimensions are called *axes*.

For example, the coordinates of a point in 3D space [1, 2, 1 have one axis. That axis has 3 elements in it, so we say it has a length of 3. In the example pictured below, the array has 2 axes. The first axis has a length of 2, the second axis has a length of 3.

[[1, 0, 0]

[0, 1, 2]]

NumPy’s array class is called ndarray. It is also known by the alias array. Note that NumPy array is not the same as the Standard Python Library class array.array which only handles one-dimensional arrays and offers less functionality.

### Matplotlib:

Matplotlib has an extensive codebase that can be daunting to many new users. However, most of matplotlib can be understood with a fairly simple conceptual framework and knowledge of a few important points.

Plotting requires action on a range of levels, from the most general (e.g., 'contour this 2-D array') to the most specific (e.g., 'color this screen pixel red'). The purpose of plotting package is to assist you in visualizing your data as easily as possible, with all the necessary control -- that is, by using relatively high-level commands most of the time, and still have the ability to use the low-level commands when needed.

Therefore, everything in matplotlib is organized in a hierarchy. At the top of the hierarchy is the matplotlib "state-machine environment" which is provided by the matplotlib.pyplot module. At this level, simple functions are used to add plot elements (lines, images, text, etc.) to the current axes in the current figure.

The next level down in the hierarchy is the first level of the object-oriented interface, in which pyplot is used only for a few functions such as figure creation, and the user explicitly creates and keeps track of the figure and axes objects. At this level, the user uses pyplot to create figures, and through those figures, one or more axes objects can be created. These axes objects are then used for most plotting actions.

### Tensorflow:

Tensorflow is one of the widely used libraries for implementing Machine learning and other algorithms involving a large number of mathematical operations. Tensorflow was developed by Google

#### History of TensorFlow

A couple of years ago, deep learning started to outperform all other machine learning algorithms when giving a massive amount of data. Google saw it could use these deep neural networks to improve its services:

* Gmail
* Photo
* Google search engine

They build a framework called **Tensorflow** to let researchers and developers work together on an AI model. Once developed and scaled, it allows lots of people to use it.

It was first made public in late 2015, while the first stable version appeared in 2017. It is open source under Apache Open Source license. You can use it, modify it and redistribute the modified version for a fee without paying anything to Google.

#### TensorFlow Architecture

Tensorflow architecture works in three parts:

* Preprocessing the data
* Build the model
* Train and estimate the model

It is called Tensorflow because it takes input as a multidimensional array, also known as **tensors**. You can construct a sort of **flowchart** of operations (called a Graph) that you want to perform on that input. The input goes in at one end, and then it flows through this system of multiple operations and comes out the other end as output.

This is why it is called TensorFlow because the tensor goes in it flows through a list of operations, and then it comes out the other side.

#### Tensors:

The core component of TensorFlow is the computational graph and Tensors which traverse among all the nodes through edges.

Mathematically a Tensor is an N-dimensional vector, which means a Tensor can be used to represent N-dimensional datasets. For example, if we take a Tensor of the form (3x3) then I can simply call it a matrix of 3 rows and columns. If I select another Tensor of the form (1000x3x3), I can call it as a vector or set of 1000 3x3 matrices. Here we call (1000x3x3) as the shape or Dimension of the resulting Tensor. Tensors can either be a constant or a variable.

### Keras:

Keras is a high-level neural networks API, capable of running on top of Tensorflow, Theano, and CNTK. It enables fast experimentation through a high level, user-friendly, modular and extensible API. Keras can also be run on CPU, GPU and TPU.

Keras was developed and is maintained by Francois Chollet and is part of the Tensorflow core, which makes it Tensor Flows preferred high-level API.

## ENVIRONMENT USED:

### Google Colab:

It’s a **Jupyter notebook environment** that requires no setup to use. In simpler terms, it’s a jupyter notebook with all the collaboration abilities of Google docs, meaning more than one person can work on the same code at the same time. But, the real attraction is the **free computing power** that this tool offers.

Google colaboratory currently offers the computing services of a **Tesla K80 GPU** for free. Yeah, you heard that right -“*free*”. The only catch here is that you can use the computing services for a **maximum of 12 hours at a time** (you can think of it in terms of a session).



Basically, when you train your models on the colaboratory, you are connected to a GPU-based virtual machine where you are given a maximum of 12 hours at a time, after which you lose access to that particular virtual machine instance (all data, that is, model parameters as well as datasets that aren’t saved to the Google drive before this period will be lost, so make sure to save snapshots of your model parameters at regular intervals, else you will have to start training your models from scratch again). After 12 hours you are assigned a different virtual machine (for free, of course) and the cycle repeats. There’s **no limit** for how many virtual machines can be used through one account, so “*Train*” to your heart’s content.

## Artificial intelligence

Artificial intelligence is the study of agents that perceive the world around them, form plans and make decisions to achieve their goals. Its foundations include mathematics, logic, philosophy, probability, linguistics, neuroscience, and decision theory. Many fields fall under the umbrella of AI, such as computer vision, robotics, machine learning, and natural language processing*.*

## Machine learning

Machine learning is a subfield of artificial intelligence. Its goal is to enable computers to learn on their own. A machine’s learning algorithm enables it to identify patterns in observed data, build models that explain the world, and predict things without having explicit pre-programmed rules and models.

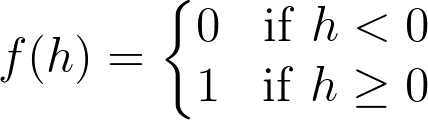
## Deep learning

Deep learning is a subfield of machine learning dealing with algorithms inspired by the structure and function of the brain called artificial neural networks. In other words, It mirrors the functioning of our brains. Deep learning algorithms are similar to how the nervous system structured where each neuron connected to each other and passing information.

### Activation Function In deep learning:

Activation functions are functions that decide, given the inputs into the node, what should be the node’s output? Because it’s the activation function that decides the actual output, we often refer to the outputs of a layer as its “activations”.

One of the simplest activation functions is the **Heaviside step function**. This function returns a **0** if the linear combination is less than 0. It returns a **1** if the linear combination is positive or equal to zero.



### Training in Neural Network:

Weights start out as random values, and as the neural network learns more about what kind of input data leads to a student being accepted into a university(above example), the network adjusts the weights based on any errors in categorization that the previous weights resulted in. This is called **training** the neural network. Once we have the trained network, we can use it for predicting the output for similar input.

## 

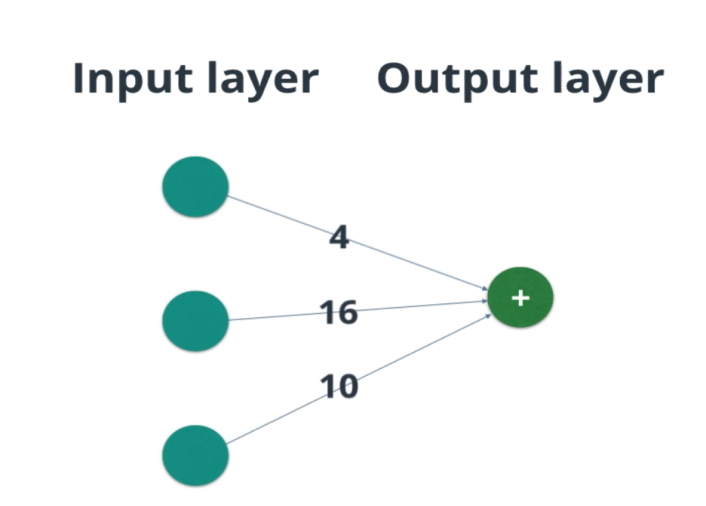
## 

### Error:

This very important concept to define how well a network performing during the training. In the training phase of the network, it makes use of error value to adjust the weights so that it can get reduced error at each step. The goal of the training phase to minimize the error

### Forward Propagation:

By propagating values from the first layer (the input layer) through all the mathematical functions represented by each node, the network outputs a value. This process is called a **forward pass**.



### Back Propagation

In neural networks, you forward propagate to get the output and compare it with the real value to get the error. Now, to minimize the error, you propagate backward by finding the derivative of the error with respect to each weight and then subtracting this value from the weight value. This is called backpropagation.

Before, we saw how to update weights with gradient descent. The backpropagation algorithm is just an extension of that, using the chain rule to find the error with respect to the weights connecting the input layer to the hidden layer (for a two-layer network).

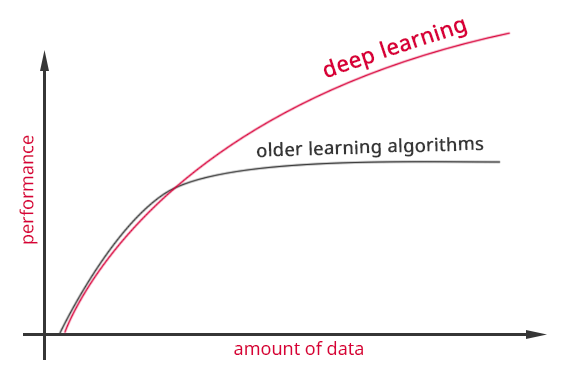
### Regularisation:

Regularisation is the technique used to solve the over-fitting problem. Over-fitting happens when the model is biased to one type of dataset. There are different types of regularisation techniques, I think the most used regularisation is dropout.

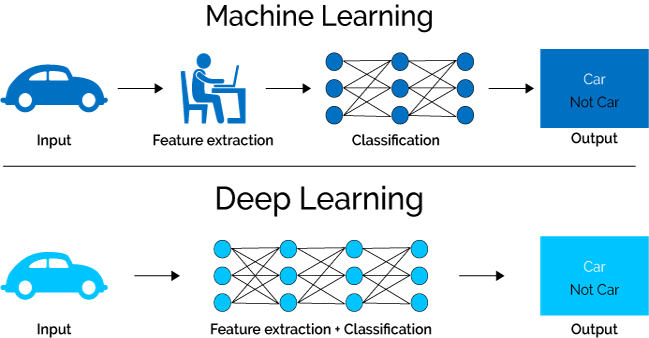
### Optimization:

The optimization is a technique used to minimize the loss function of the network. There are different types of optimization algorithms.

### Why Deep Learning:



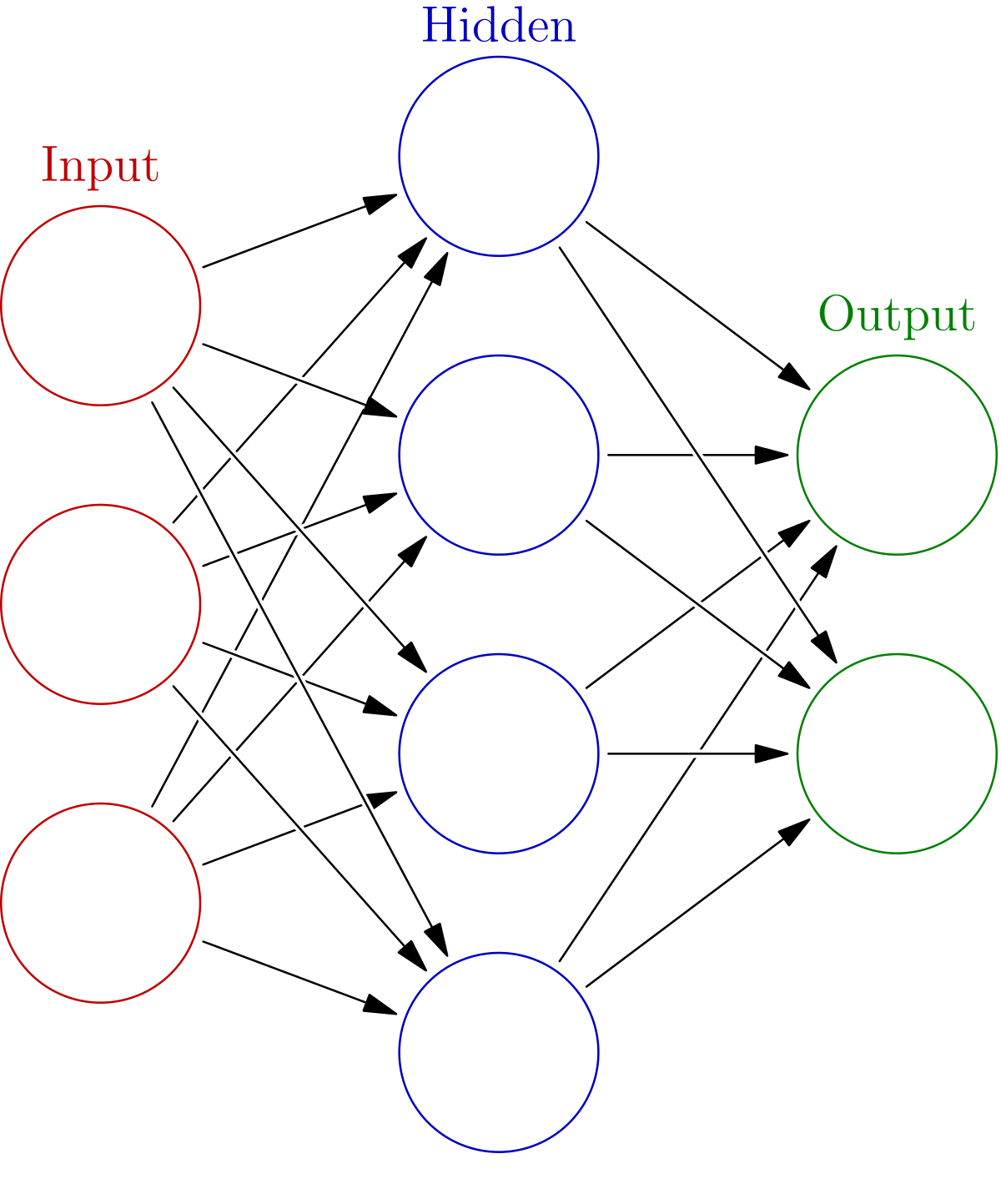
**Performance Of Deep Learning**



**Deep Learning VS Machine Learning**

## Artificial Neural Networks(ANN):

An ANN is a computational model inspired by networks of biological neurons, wherein the neurons compute output values from inputs. It learns from its past experience and errors in a non-linear parallel processing manner. The neuron is the basic calculating entities which compute from a number of inputs and delivers one output compared with a threshold value and turned on (fired). The computational processing is done by internal structural arrangement consists of hidden layers that utilize the backpropagation and feed-forward mechanism to deliver output close to accuracy. The learning is based on reinforcement (supervised) and unsupervised (no target) type. The unsupervised mimics the biological neuron pattern of learning.

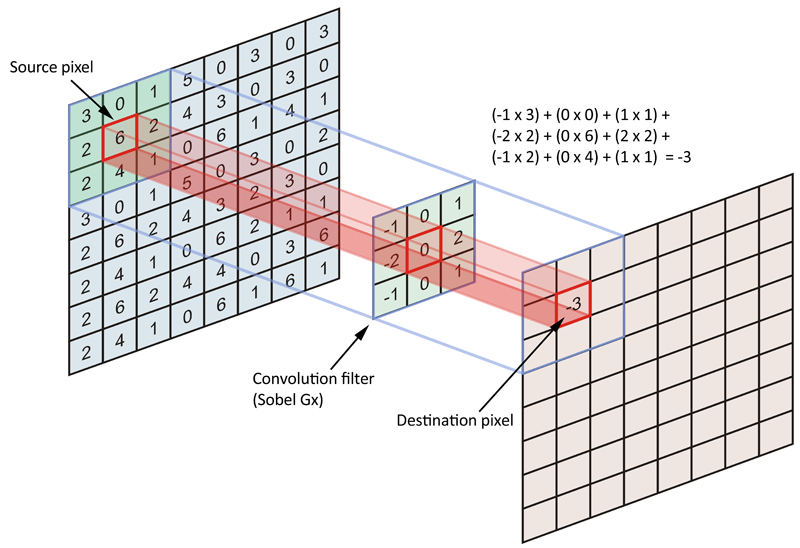


**ARTIFICIAL NEURAL NETWORKS**

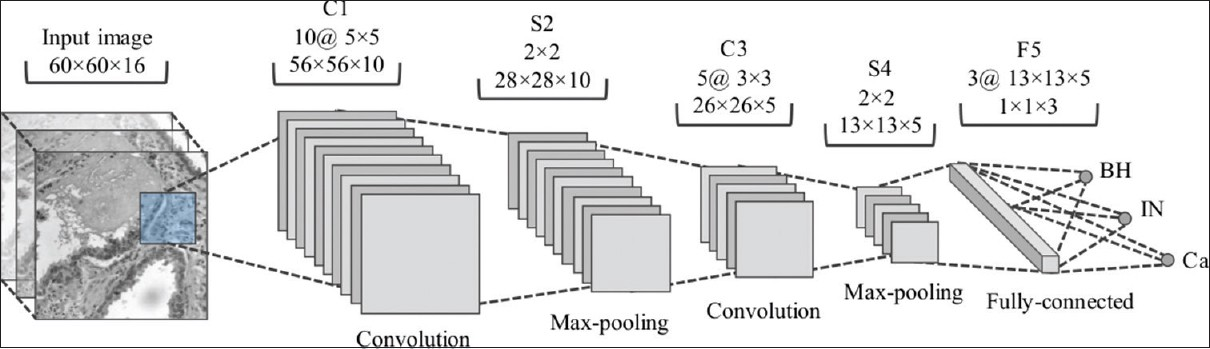
**Introduction to Convolutional Neural Networks (CNN):**

A Computer sees an image in the form of a set of pixels or simply numbers in a 2D array. If the image is grayscale there is only one such array of numbers. If there is a colored image, there is a set of 3 such arrays, one each for Red, Green, and Blue.

When an image is given as an input to the network, convolutions are calculated. This means that given an array of pixels, we have another set of array called a filter (or convolutional kernel), which has the same dimensions of the receptive field and this filter is operated on the pixel array through some operations.



**Convolution Process**

****

**Details Of CNN:**

#### Convolution Layer:

Convolution is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel

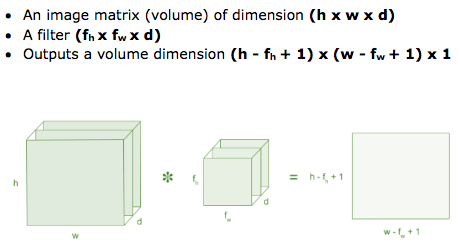


Image matrix multiplies kernel or filter matrix

#### Padding:

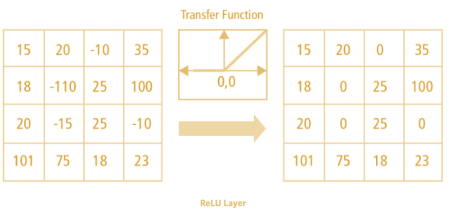
Sometimes filter does not fit perfectly fit the input image. We have two options:

* Pad the picture with zeros (zero-padding) so that it fits
* Drop the part of the image where the filter did not fit. This is called valid padding which keeps only valid part of the image.

#### Non Linearity (ReLU):

ReLU stands for Rectified Linear Unit for a non-linear operation. The output is ***ƒ(x) = max(0,x).***

Why ReLU is important : ReLU’s purpose is to introduce non-linearity in our ConvNet. Since, the real world data would want our ConvNet to learn would be non-negative linear values.



ReLU operation

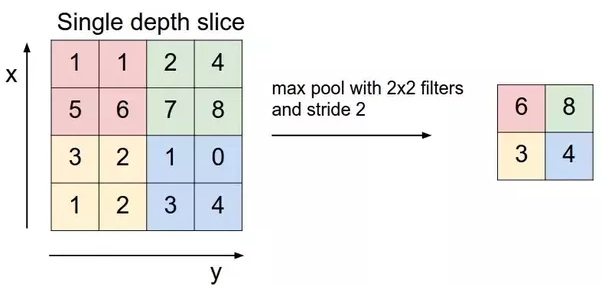
There are other nonlinear functions such as tanh or sigmoid can also be used instead of ReLU. Most of the data scientists uses ReLU since performance wise ReLU is better than the other two.

#### Pooling:

Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or downsampling which reduces the dimensionality of each map but retains the important information. Spatial pooling can be of different types:

* Max Pooling
* Average Pooling

Max pooling take the largest element from the rectified feature map. Taking the largest element could also take the average pooling. Sum of all elements in the feature map call as sum pooling.



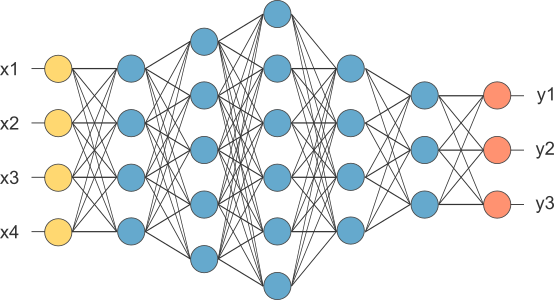
Max Pooling

#### Flattening:

All outputs from the pooling are flattened to make a 1 level tensor to feed it into artificial neural network.

#### Fully Connected Layer:

The layer we call as FC layer, we flattened our matrix into vector and feed it into a fully connected layer like neural network.



After pooling layer, flattened as FC layer

In the above diagram, feature map matrix will be converted as vector (x1, x2, x3, …). With the fully connected layers, we combined these features together to create a model. Finally, we have an activation function such as SoftMax or Sigmoid to classify the outputs as y1, y2, y3.

# ABOUT PROJECT

## VISION:

This project is useful in detection and recognition of hand movements and gestures. By detecting the hand gestures, this project can extend further to the object detection and computer vision.

## DESCRIPTION:

The name of this project is “Recognition of hand gestures as stone paper and scissor”. In this project we will detect the hand gesture using deep learning and Convolutional neural networks. In this project we can upload an image from our local machine.

* Then we have to load the deep learning model.
* After model has been loaded, now model is ready to predict the image.
* Then click the predict button, which will predict the uploaded image.
* System will sound or generate text for the predicted image result.

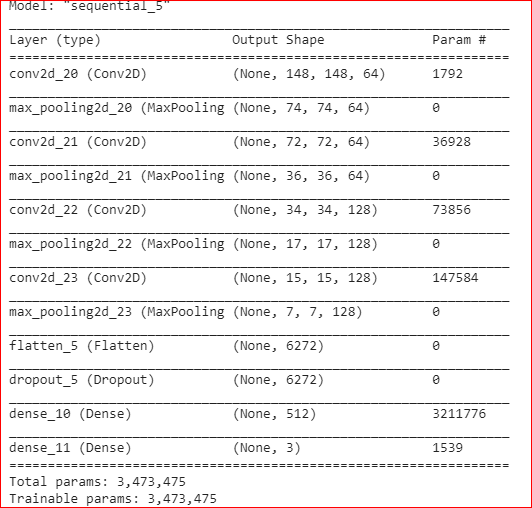
## DATASET:

Our dataset is like:

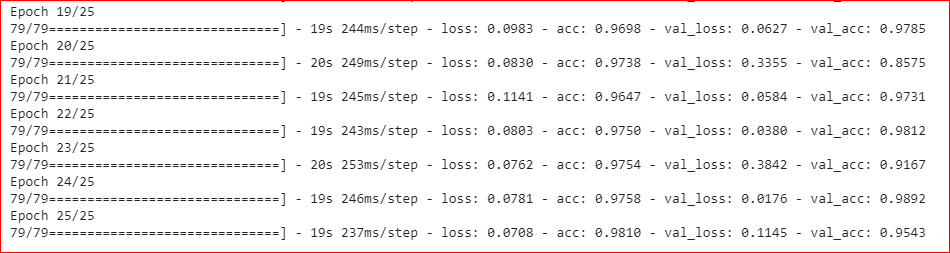


## DIRECTORY STRUCTURE:

## Layer in the model:



## Model Train and Test Accuracy:



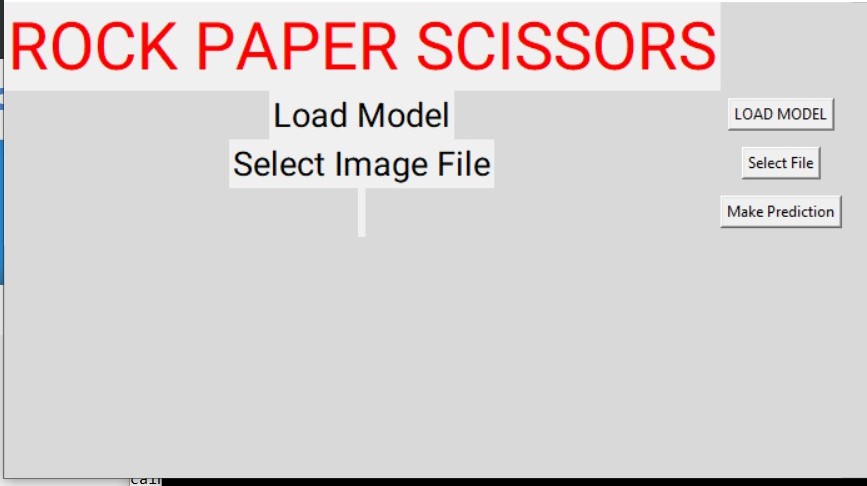
**Train Accuracy: 98.12**

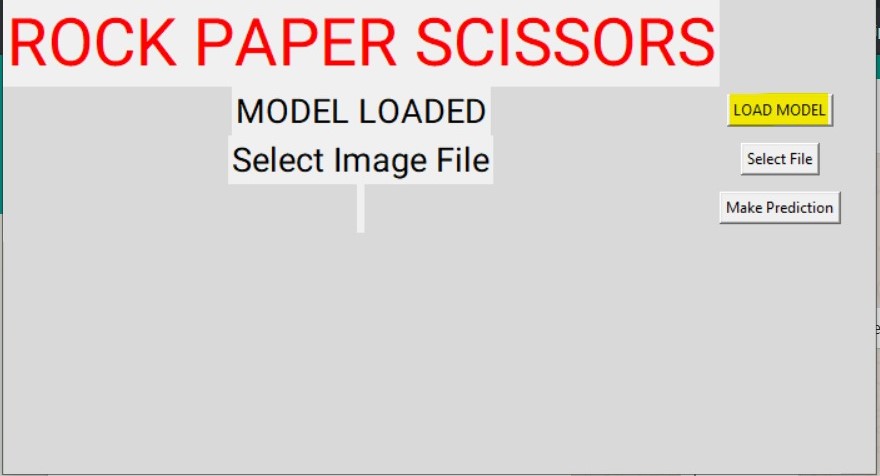
**Test Accuracy: 95.43**

## USER INTERFACE:

**Steps To deal with Interface:**

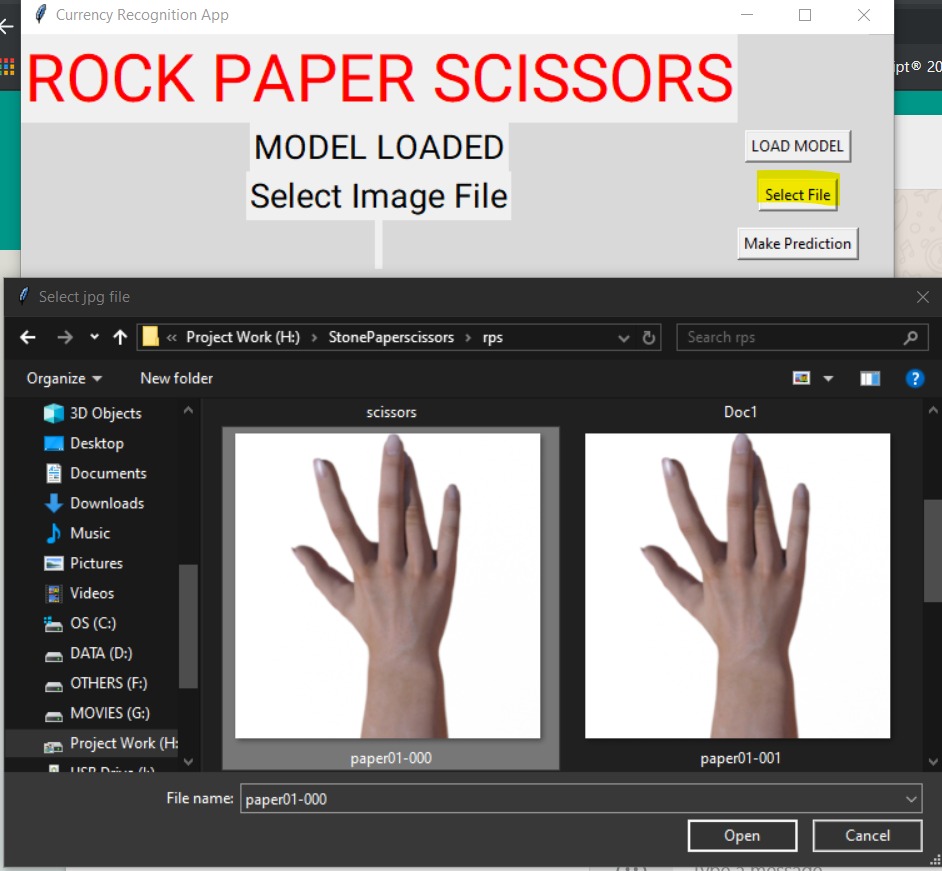
* **First load the model**: To load the model we have to click on ‘Load Model’ button which will load the deep learning model into the system to make further predictions.

****

****

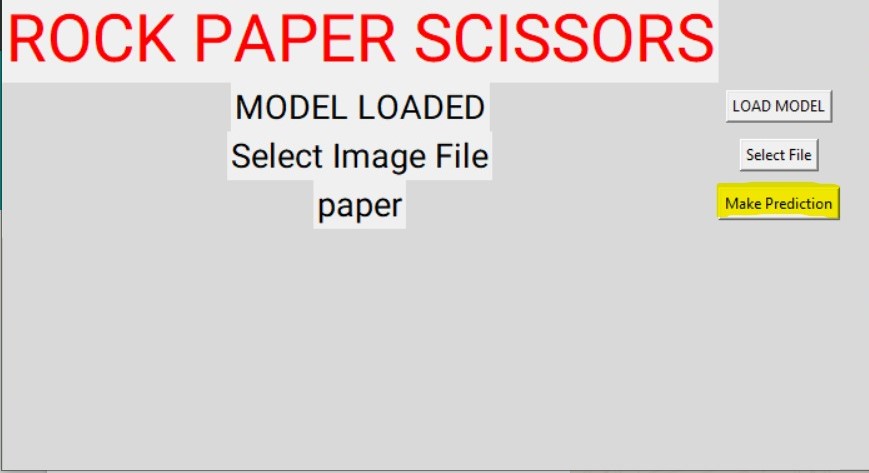
* **Now Select the image File:**

After loading the model, now select the image file from your local machine.



* **Predict the image:**

Now file has been selected now it's time to predict the selected image using ‘Make Prediction’ button on the interface.



**Conclusion:**

Here we had selected ‘paper’ image and the model had predicted the image as paper. Here we conclude that our model is predicting our image correctly. As our test accuracy was 95.43%, also model is predicting the selected image efficiently.

**BIBLIOGRAPHY**

ACADVIEW

Reference:

<https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/>

<https://python.org>