A

Project Report on

**HAND GESTURE DETECTION**

**Submitted in partial fulfillment of the requirement for the award of**

**BACHELOR OF TECHNOLOGY**

in

**Branch(COMPUTER SCIENCE AND ENGINEERING)**

UNDER THE GUIDANCE OF**:**

**Mr. AMIT BAGHEL**

**(Assistant professor)**

**SUBMITTED BY:**

**NAME:**Vikram Shishupalsingh Bais **Roll No:**18103059

**NAME**:Vetcha.Yaswanthi **Roll No**:18103058

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

****

**INSTITUTE OF TECHNOLOGY**

**GURU GHASIDAS VISHWAVIDYALAYA, BILASPUR, INDIA**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**INSTITUTE OF TECHNOLOGY**

****

**GURU GHASIDAS VISHWAVIDYALAYA ,BILASPUR (C.G.)**

**CERTIFICATE**

I hereby certify that the work which is being presented in the B.Tech. Minor Project Report entitled “Hand Gesture Detection”, in partial fulfillment of the requirements for the award of the Bachelor of Technology in Computer Science and Engineering and submitted to the Department of Computer Science and Engineering,Institute of Technology, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, Chhattisgarh, India is an authentic record of my own work carried out during a period from January 2021 to May 2021( 6th semester) under the supervision of Mr.AMIT BHAGEL Assistant Professor CSE Department. The matter presented in this Project Report has not been submitted by me or by anyone else for the award of any other degree elsewhere.

Signature of Student (S)

STUDENT NAME: Vikram Shishupalsingh Bais Roll No:1810305

STUDENT NAME: Vetcha.Yaswanthi Roll No:18103058

This is to certify that the above statement made by the student(s) is correct to the best of my knowledge.

Signature of Supervisor

AMIT BAGHEL(Asst.Professor)

Date:

Dr. Alok Kumar Singh Kushwaha

Deptt. of Computer Science and Engineering Department

**DECLARATION**

We hereby declare that the project work entitled “Hand Gesture detection” submitted to Institute of Technology, Guru Ghasidas Vishwavidyalaya, is a record of an original work done by us under the guidance of **Mr.Amit Baghel**, Assistant Professor Department of Computer Science and Engineering, and this project is submitted in the partial fulfillment of requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**STUDENT NAMES:**

Vikram Shishupalsingh Bais(18103059)

Vetcha.Yaswanthi(18103058)

**ACKNOWLEDGEMENTS**

We would like to place on record my deep sense of gratitude to Mr.Amit Baghel,Assistant Professor, Dept. of Computer Science and Engineering, Institute of Technology, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, Chhattisgarh, India, for his stimulating guidance, continuous encouragement and supervision throughout the course of present work. We express my sincere gratitude to Dr. Alok Kumar Singh Kushwaha, Associate Professor and HOD-Dept.of Computer Science and Engineering, Institute of Technology, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, Chhattisgarh, India for his generous guidance, help and useful suggestions. We are extremely thankful to Prof T.V.ARJUNAN , Dean, School of Studies(Engg. & Tech.), Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, Chhattisgarh, India for providing us with infrastructural facilities to work in, without which this work would not have been possible.

Date:

Vikram Shishupalsingh Bais(18103059)

Vetcha.Yaswanthi (18103058)

**INDEX:**

CHAPTER 1 INTRODUCTION

1.1 GENERAL BACKGROUND

1.2 STATEMENT OF PROBLEM

1.3 OBJECTIVE

1.4 METHOD

CHAPTER 2 LITERATURE REVIEW

CHAPTER 3 PROPOSED METHODOLOGY

CHAPTER 4 RESULTS AND DISCUSSION

CHAPTER 5 CONCLUSION AND FUTURE DIRECTION

CHAPTER 6 REFERENCES

**CHAPTER 1 INTRODUCTION**

**1.1 GENERAL BACKGROUND**

* In this project we have created a model which uses 3D Convolutional Neural Networks for detecting the hand gestures of a person standing in front of a camera.
* For this particular model we have used The 20BN-jester Dataset V1The 20BN-JESTER dataset is a large collection of labeled video clips that show humans performing pre-defined hand gestures in front of a laptop camera or webcam. The dataset was created by a large number of crowd workers. It allows for training robust machine learning models to recognize human hand gestures. It is available free of charge for academic research. Commercial licenses are available upon request. We built a web application that implements the hand gesture recognition system and provides the recognition service to users

**1.2 STATEMENT OF PROBLEM**

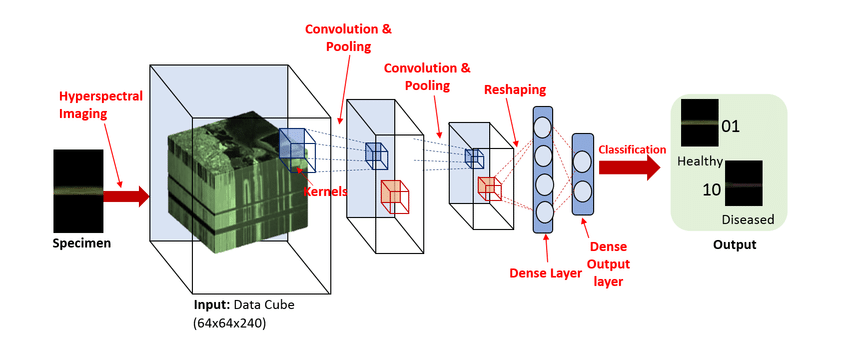
Making a A 3D CNN model that can classify different gestures using dataset of The 20BN-jester Dataset V1 taken from <https://20bn.com/datasets/jester>

**1.3 OBJECTIVE**

To make a 3D Convolutional neural network that will be able to detect hand gesture of a person standing in front of camera

**1.4 METHOD**

* Convolutional Neural Networks (CNNs) are a type of deep model that can act directly on the raw inputs. This model extracts features from both the spatial and the temporal dimensions by performing 3D convolutions, thereby capturing the motion information encoded in multiple adjacent frames.
* CNNs make image processing convenient by filtering connections through the given proximity. Instead of connecting every input to the neurons in a given layer, CNNs intentionally shorten connections so that only one neuron accepts inputs from a small subsection of the layer before it. Making each neuron responsible for only processing a specific part of an image.
* A simple CNN is a sequence of layers, and every layer transforms one volume of activations to another. There are three main types of layers in the convolutional neural network. The Convolutional Layer, the Pooling Layer, and the Fully Connected Layer.



**The Convolutional Layer:**

* The Convolutional Layer is the core building block of a convolutional Neural Network that does most of the computational work.
* The convolutional layer parameters consist of a set of filters. Every filter is small, but it extends through the full depth of the input volume.

**2D CNN and 3D CNN**

* In 2d cnn we use a kernel of two dimensions for performing operations on image kernel scan pixels of image by moving only in 2 dimensions(x axis and y axis).
* In 3d CNN kernel moves in all three directions x y and z axis
* Third axis is for the time as we perform 3d cnn for classifying videos
* In **3D CNN**, the kernel moves in 3 directions. Input and output data of **3D CNN** is 4 dimensional. Mostly used on **3D** Image data (MRI, CT Scans, Video).

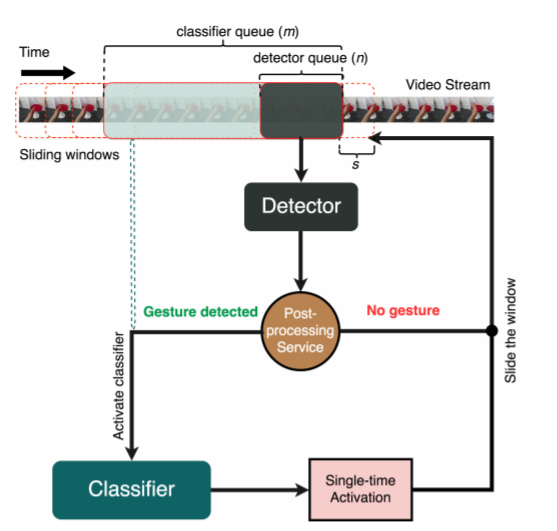
**The Pooling Layer**

* Pooling layers are meant to reduce the spatial size of the representation which helps reduce the amount of computation and parameters in the networks.
* The pooling layer resizes the depth slice using something called the Max operation (The MAX operation is basically using the maximum value from each cluster of neurons at the prior layer).

**Dense layer**

* The dense layer is a neural network layer that is connected deeply, which means each neuron in the dense layer receives input from all neurons of its previous layer.
* The dense layer is found to be the most commonly used layer in the models.

**Classification**



* Once the image is put through a Convolutional Neural Network, the computer recognizes what the gesture is. Now, all it has to do is repeat this process until it has learned about the different patterns about the object.
* So using the example in this gesture, the computer would use CNNs to recognize a gesture of the hand and when given a file of gestures, should be able to classify and group all the hand gestures correctly!

**Tools Used:**

* **Programming Language**:

Python

* **Topic:**

Deep learning, 3D Convolutional Neural Network, Computer Vision

* **Modules of python:**

Tensorflow, Keras, CV2, matplotlib, Sklearn, OS, Numpy

* **Ide Used**:

Pycharm, Google Colaboratory

**Dataset:**

20BN-jester Dataset V1 taken from <https://20bn.com/datasets/jester>

**Python:**

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

**Deep Learning:**

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

**Convolutional Neural Network:**

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

**Computer Vision:**

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

**Tensorflow:**

* TensorFlow is a free and open-source software library for machine learning.
* It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.
* Tensorflow is a symbolic math library based on dataflow and differentiable programming.

**Keras:**

* Keras is an open-source software library that provides a Python interface for artificial neural networks.
* Keras acts as an interface for the TensorFlow library.

**OpenCV:**

* OpenCV is a library of programming functions mainly aimed at real-time computer vision.
* Originally developed by Intel, it was later supported by Willow Garage then Itseez.
* The library is cross-platform and free for use under the open-source Apache 2 License.

**Matplotlib:**

* Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.
* It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+.

**Numpy:**

* NumPy is the fundamental package for scientific computing in Python.
* NumPy arrays facilitate advanced mathematical and other types of operations on large numbers of data.
* In this operations are executed more efficiently and with less code than is possible using Python's built-in sequences.

**Sklearn:**

* Scikit-learn is probably the most useful library for machine learning in Python.
* The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.
* Sklearn is a free software machine learning library for the Python programming language.

**OS:**

* The OS module in Python provides functions for interacting with the operating system.
* This module provides a portable way of using operating system dependent functionality.
* The os and os. path\*modules include many functions to interact with the file system.

**Pycharm:**

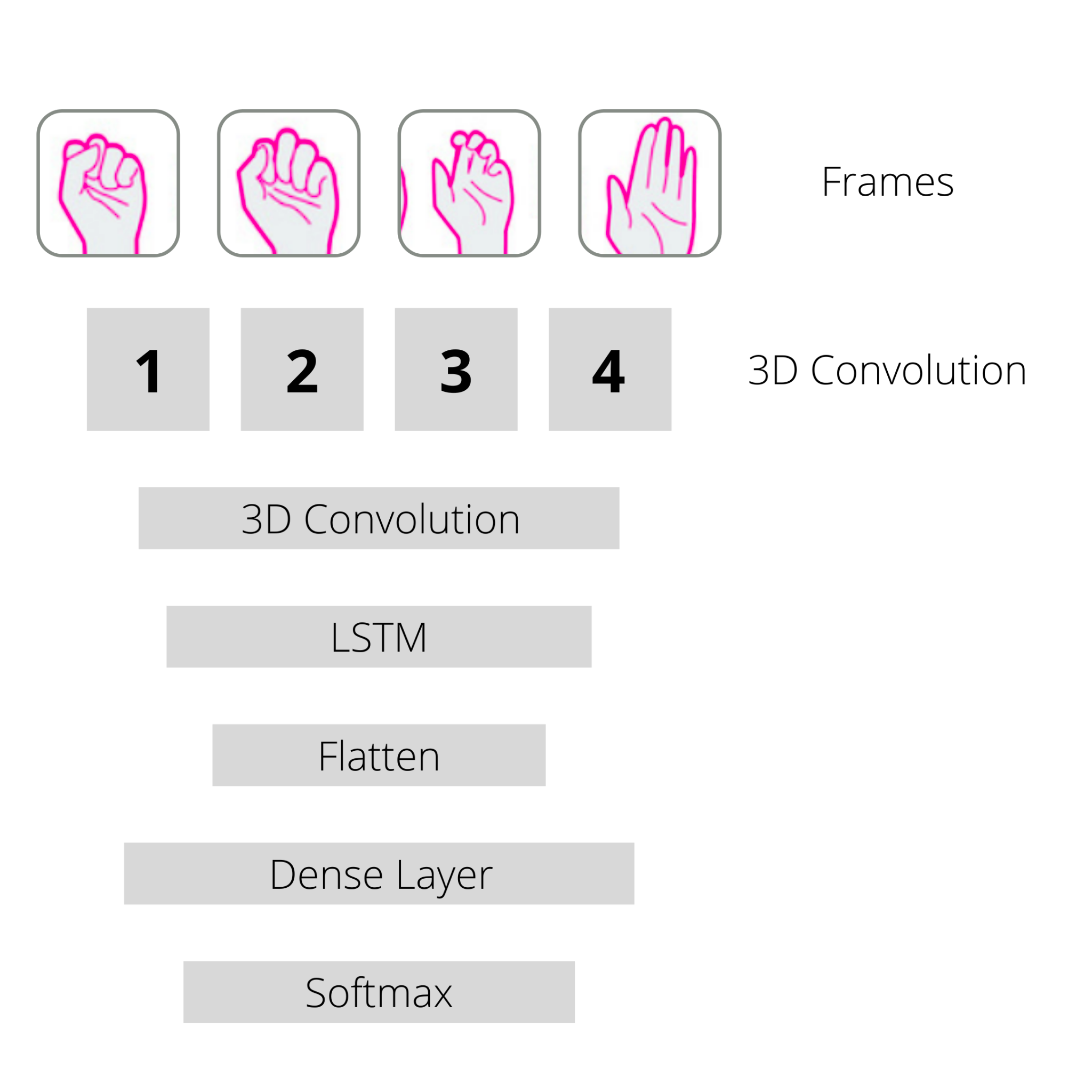
* PyCharm is an extremely popular Python IDE. An Integrated Development Environment or IDE features a code editor and a compiler for writing and compiling programs in one or many programming languages.
* Furthermore, an IDE comes with a galore of features that facilitate comprehensive software development.

**Google Colaboratory:**

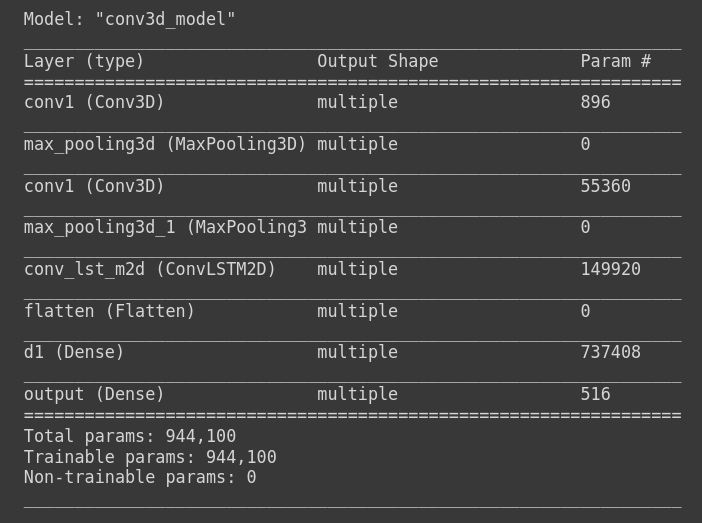
* Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.
* More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs.

**Structure Of Our Model**

The following figure represents the structure of our model



Following figure shows the model that we have used in our code



* Two 3D CNN layers each 3D CNN is followed by max pooling layer for extracting feature and performing operations on that extracted feature
* This is followed by LSTM 2D layer
* After this we have flattened our data so that we can perform simple ANN to the data for getting proper result
* This is followed by the Dense layer of 128 neurons contains relu as an activation function which is followed by the Dense layer of 4 neurons contains softmax activation function.
* The accuracy of our model is 60% as size of our data is 160 for each of the four gesture

**CHAPTER 2 LITERATURE REVIEW**

This Project can be divided into following stages:

* Importing Libraries
* Loading Dataset
* Assigning Variable Name with path for the Training and Testing dataset
* Setting Model Parameters
* Defining Model
* Training Model
* Predictions and Result
* Adding GUI

**CHAPTER 3 PROPOSED METHODOLOGY**

1. **Importing Libraries :**

For this project we have used following libraries

* Numpy
* tensorflow
* cv2
* os
* matplotlib
* Sklearn

1. **Loading dataset:**

**About dataset**

* The dataset that we have used here is The 20BN-jester Dataset V1 taken from <https://20bn.com/datasets/jester>
* In dataset there are total 27 gestures out of which we have taken four gestures for creating our project
* Swiping right, swiping up, swiping down, no gesture
* In the dataset there is folder containing 30-40 images for each gesture, Each gesture has different folders for their images
* From this huge dataset we have taken four gestures each gesture containing 160 folders(contains framewise images in order)
* For each gesture we have taken 10 folders for testing

**Adjusting dataset**

* As our dataset does not contain equal images for each gesture also we have to send equal numbers for images for each gesture so first we need to decide number of ordered images for training
* We took 30 frames for training each gesture
* So for those folders that have less than 30 images we have added a copy of last image to make it 30
* Also for those folders who have greater than 40 we remain those folders as it is as we need only 30 framewise images

1. **Setting model parameters :**

* setting parameters like dimension of frames, epoch
* After this we have resized every frame to (64, 64)

1. **Defining Model :**

defining model, adding convolution to model, adding required function to each convolution (each convolution contains 3dCNN layer and Pooling function)

1. **Training :**

Training or Compiling model using provided dataset

1. **Prediction :**

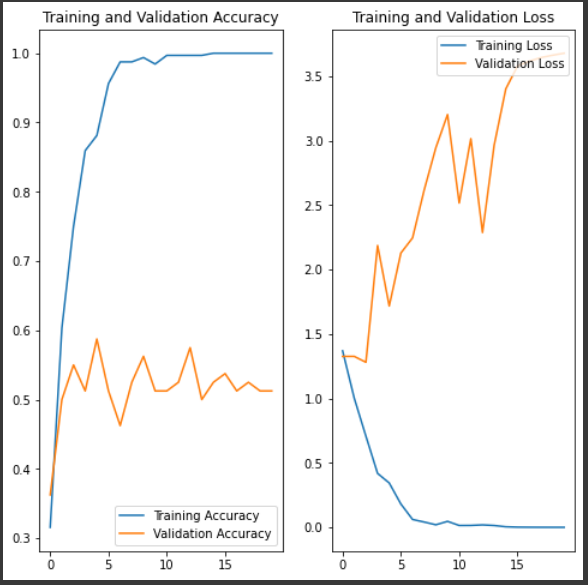
Checking result by taking input as frames of video (video can be taken from webcam or we can take pre recorded video)

1. **Adding GUI :**

Adding graphics using Tkinter module, making user friendly

**CHAPTER 4 RESULTS AND DISCUSSION**

* We have trained models using Hand gestures dataset that contain 4 different training gestures each gesture contains 160 data for training .
* Following is the graph for comparing training data and testing data



* We evaluated the performance of our dynamic hand gesture recognition system using leave-one-subject-out cross validation on the JESTER dataset.
* We used data from the 8 subjects.In which 4 are used for testing and trained the classifier with data from the 4 remaining subjects; we repeated this process for each of the 8 subjects and averaged the accuracy.
* Running the script will open new windows which will show captured video from our web camera with text of detected gesture written in the corner of the window.
* **Total training data** = 4\*160
* **Total testing data** = 4\*10
* **Accuracy** = 60%

**CHAPTER 5 CONCLUSION AND FUTURE DIRECTION**

**Conclusion:**

* Accuracy of the model directly depends on the dataset and numbers of layers that we have used in the model.
* If we increase the number of images on the dataset then accuracy will be increased.
* We developed an effective method for dynamic hand gesture recognition with 3D convolutional neural networks.
* The proposed classifier uses a fused motion volume of normalized depth and image gradient values, and utilizes spatiotemporal data augmentation to avoid overfitting.
* By means of extensive evaluation, we demonstrated that the combination of low and high resolution sub-networks improves classification accuracy considerably.
* We further demonstrated that the proposed data augmentation technique plays an important role in achieving superior performance.
* For the challenging Jester dataset, our proposed system achieved a classification rate of 60%.
* Our future work will include more adaptive selection of the optimal hyperparameters of the CNNs, and investigating robust classifiers that can classify higher level dynamic gestures including activities and motion contexts.

**Applications:**

* 3D CNN is mostly used for Medical purposes like MRI scans etc.,
* It can be used to detect activity detection of human
* A 3D Convolution can be used to find patterns across 3 spatial dimensions; i.e. depth, height and width.

**Future Direction:**

This project can be further modified

1. We can increase the size of our dataset so that we can guess more different gestures with better accuracy.
2. We can predict human pose using different dataset so that it can be used for human pose estimation.

**CHAPTER 6 REFERENCES**

**Link for dataset** : <https://20bn.com/datasets/jester>