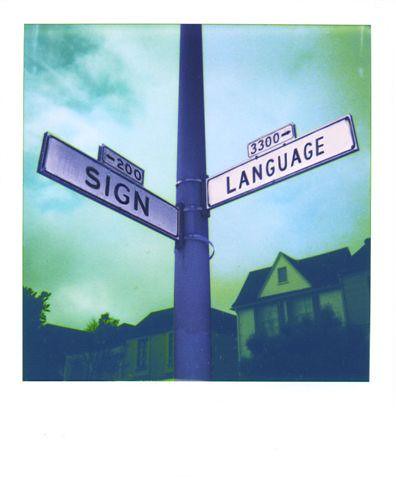
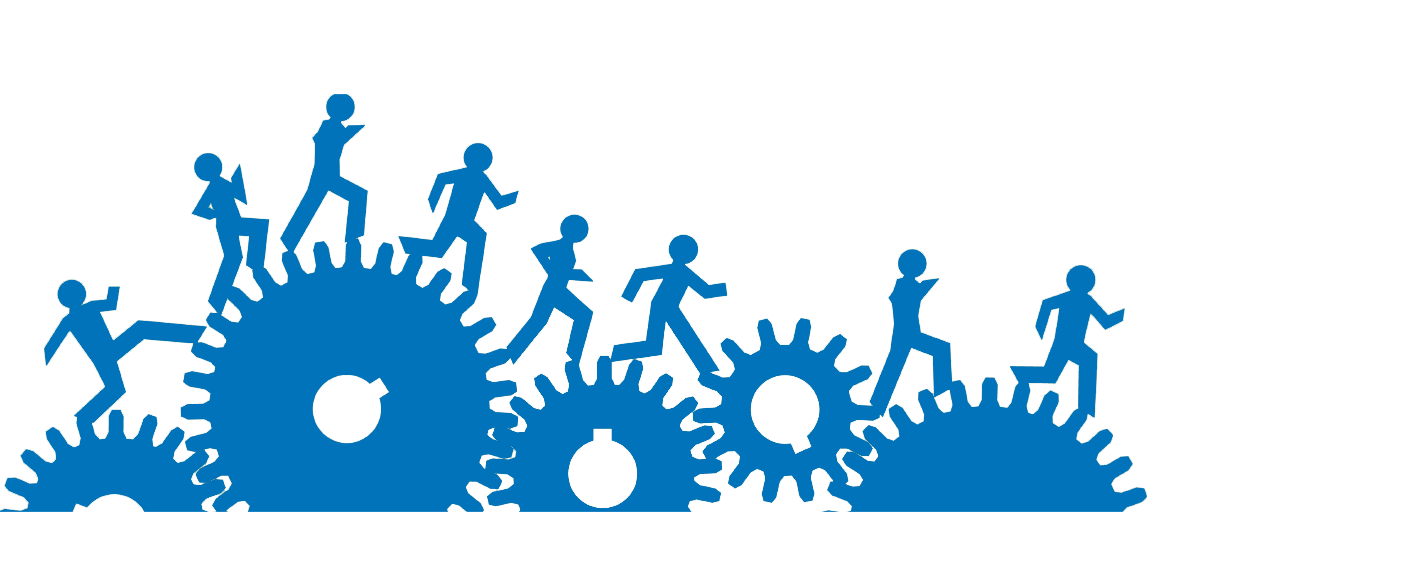
|  |
| --- |
| PROJECTREPORT |

SIGN LANGUAGE RECOGNITION

KHUSHI GUPTA

BTECH CS 3rd Year

201500342



Contents

ABSTRACT2

DECLARATION………………………………………………………………….3

CERTIFICATION………………………………………………………………..4

INTRODUCTION5

PROBLEM DEFINITION AND ALGORITHM6

DATASET6

TASK DEFINITION7

ALGORITHM DEFINITION8

EXPERIMENTAL EVALUATION10

METHODOLOGY

RESULTS

FUTURE WORK

CONCLUSION

DECLARATION

I Khushi Gupta of B.Tech (CSE) in 3rd Year hereby declare that the work submitted for the project of SIGN LANGUAGE RECOGNITION is my original work. I have not copied from any other student’s work or any other sources except where due reference or acknowledgment is made explicitly, and the work has been carried out under the guidance of Mr. Akash Kumar Choudhary, Department of Computer Science, GLA University. I further declare that the work on this project in this project has not been submitted and will not be submitted, either in part or in full, for any other degree in this institute or any other institute or university.

Candidate Name:

Khushi Gupta

Uni Roll No.- 201500342

Certification

Certified that this project report “SIGN LANGUAGE RECOGNITION” is the bonafide work of “KHUSHI GUPTA” who carried out the project work under Mr. Akash Kumar Choudhary’s supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

ABSTRACT

A group of established languages known as sign languages employ a visual-manual modality to communicate. We look at the issue of real-time finger-spelling recognition in Indian Sign Language (ISL). For the purpose of identifying 36 different gestures, we gathered a dataset of depth-based segmented RGB images using the Microsoft XBOX360 Kinect Camera (alphabets and numerals). The device receives a hand motion as input and instantly displays the appropriate identified character on the monitor. With the Deep Convolutional Neural Network, we were able to classify data with an accuracy of 89.30%.

*Keywords:* Deep convolutional neural network, gestures, and sign language, RGB.

GITHUB LINK

<https://github.com/khushigupta7902/Sign-Language-Recognition>

The main reason sign languages are created is to help the deaf and dumb. To express certain information, they use a coordinated and precise combination of hand motions, hand forms, and hand orientation. Indian Sign Language (ISL), which is primarily utilized in south Asian nations, is one such system of languages. ISL has some characteristics that set it apart from other sign languages, including the use of both hands and the absence of any temporal intonations in its finger pronunciation chart.

INTRODUCTION

Healthcare, robotics, automated self-driving cars, human computer interaction (HCI), and other fields have experienced rapid growth as a result of the development of artificially intelligent algorithms, enormous data, and large computational resources.

Systems for augmented reality, facial recognition, and hand-gesture recognition all use HCI in some capacity. The goal of this HCI-related project is to identify different ISL family alphabets (a-z) and digits (0-9). Recognizing hand gestures is a difficult task, and ISL identification is hard because it uses both hands. Many studies in this area have been conducted in the past employing sensors (such as glove sensors) and various image processing methods (such as edge detection technique, Hough Transform, etc.), but the outcomes were never adequate. However, the performance in this area has substantially improved thanks to new deep learning techniques like CNN, opening up a wide range of potential future opportunities.

Many people in India have speech or hearing impairments, therefore they communicate with others by using hand gestures. Apart from a small group of individuals, not everyone is conversant with this sign language, thus they might need an interpreter, which can be both inconvenient and expensive. By creating algorithms that can instantly predict the ISL alphanumeric hand motions, our effort seeks to close this communication gap.

1. Overview of the Problem

PROBLEM STATEMENT

To create a software system capable of recognizing ISL hand gestures in real-time using deep learning methods. The goal of this project is to foresee the ISL system's "alphanumeric" gesture.



1. Goals for the Research

* Using the Microsoft Kinect (v1) camera to produce a great amount of useful data.
* Using the proper picture pre-processing methods to get the ROI and eliminate the noise.
* Creating a model and architecture for CNN that will allow it to train on pre-processed images and attain the highest level of accuracy.
* Create an algorithm that can instantly predict the gesture.

In this project, I implemented various machine-learning algorithms for detecting the onset for sign language. The algorithms were trained on data from people who were using sign language as a mode for communication during the follow-up period (6 months).

1. Litreature Review

As the dataset that I acquired was labeled, so opted to work with supervised learning algorithms. I implemented multiple classification algorithms to predict whether the person is using sign language or not as a mode of communication.

Two networks, each with three layers of convolution and three layers of max-pooling, were utilized by Lionel Pigou and colleagues. Both of the CNNs were trained to recognise features on the upper body and the hand. Concatenated outputs from the two CNNs were fed into a fully linked layer. They made use of the CLAP14 dataset, which included 27 participants and 20 Italian gestures. They made advantage of both the colour and depth images. They were accurate 91.7% on cross-validation, which included users with various backgrounds from the training set, and 95.68% on testing, which also included users with training-set backgrounds.

1. **Dataset**

The dataset used in this project: the Sign Language MNIST dataset.

**Sign Language MNIST**

The initial MNIST image dataset of handwritten digits serves as a prominent benchmark for image-based machine learning techniques, but academics are working hard to update it and create drop-in replacements that are both more difficult for computer vision and unique for practical applications. The Zalando researchers cited the shocking assertion that "Most pairings of MNIST digits (784 total pixels per sample) can be discriminated quite well by just one pixel," as was mentioned in one recent replacement known as the Fashion-MNIST dataset. The Sign Language MNIST is offered here and uses the same CSV format with labels and pixel values in single rows to encourage the community to create more drop-in replacements.

With 24 classes of letters, the American Sign Language letter database of hand gestures is a multi-class problem (excluding J and Z which require motion).

The dataset format roughly resembles the traditional MNIST. There are no cases for 9=J or 25=Z due to gesture motions; each training and test case represents a label (0–25) as a one-to-one map for each alphabetic letter A–Z. The training data (27,455 cases) and test data (7172 cases) are roughly half the size of the standard MNIST but otherwise similar, both having a header row of labels that read "pixel1, pixel2" through "pixel784," each of which represents a single 28x28 pixel image with grayscale values ranging from 0-255. Multiple users repeating the move across various backgrounds were represented by the original hand gesture image data.

The MNIST data for sign language was obtained by considerably expanding the scan of the few (1704) colour photos that were not clipped around the hand region of interest. An ImageMagick-based image pipeline was utilised to generate fresh data, and it includes hand-only cropping, gray-scaling, resizing, and at least 50+ variants to increase the quantity. Filters (Mitchell, Robidoux, Catrom, Spline, and Hermite), 5% random pixelation, +/- 15% brightness/contrast, and finally 3 degrees rotation made up the modification and expansion strategy.

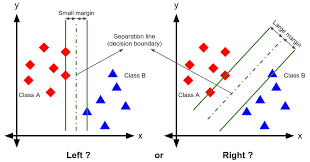
1. **Algorithm Used**

For supervised learning, which involves categorising the dataset before putting it into the algorithm for training, classification machine learning algorithms like SVM and k-NN are utilised. SVM, k-NN, and CNN are among the classification methods employed in this research.

The algorithms used are as follow:

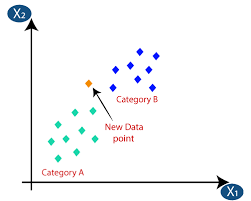
* **Support Vector Machine (SVM)**

Each data point in SVM is plotted in an n-dimensional space (n being the number of features), with each feature's value being the value of a certain coordinate. Finding the hyper-plane that best distinguishes the classes allows for classification.



* **k-NN (k-Nearest Neighbours)**

An object is classified in k-NN classification by the majority vote of its neighbours, with the object being allocated to the class that is most prevalent among its k-nearest neighbours, where k is a positive integer, often small. The algorithm's output is a class membership.



* **CNN**

Deep neural networks called convolutional neural networks (CNN) are used to analyse data with a structure that resembles a grid, such as photographs that can be represented as a 2-D array of pixels. Four primary processes make up a CNN model: convolution, nonlinearity (Relu), pooling, and classification (Fully-connected layer ).

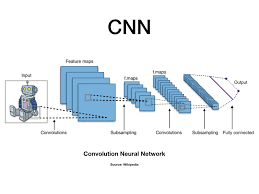
**Convolution:** Convolution is used to extract features from an image's input. Through the use of small squares of input data to learn image attributes, the spatial link between pixels is preserved. Relu generally comes after it.

**Relu**: All negative pixel values in the feature map are replaced by zero using Relu, an element-wise procedure. Its goal is to give a convolution network non-linearity.

**Pooling:** Pooling, also known as downsampling, reduces the detail of each feature map while keeping crucial information.

**Fully-connected layer:** A multi-layer perceptron with a fully linked layer that employs the softmax function in the output layer. Its goal is to classify the input image into distinct classes using training data and characteristics from previous layers.

In order to build a CNN model, these layers are combined. A fully connected layer makes up the final layer.

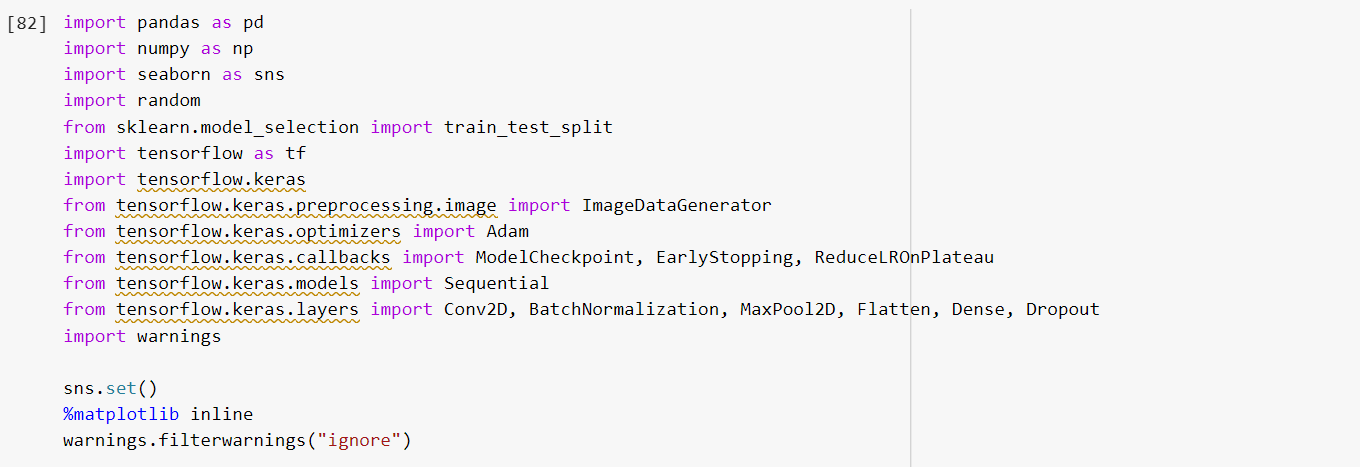


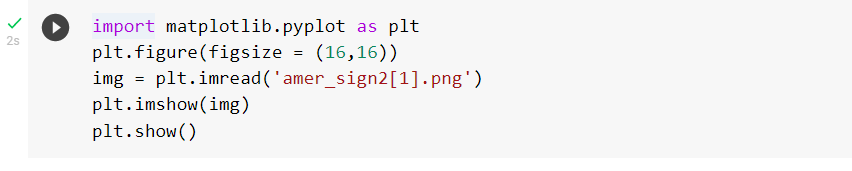
1. **METHODOLOGY**

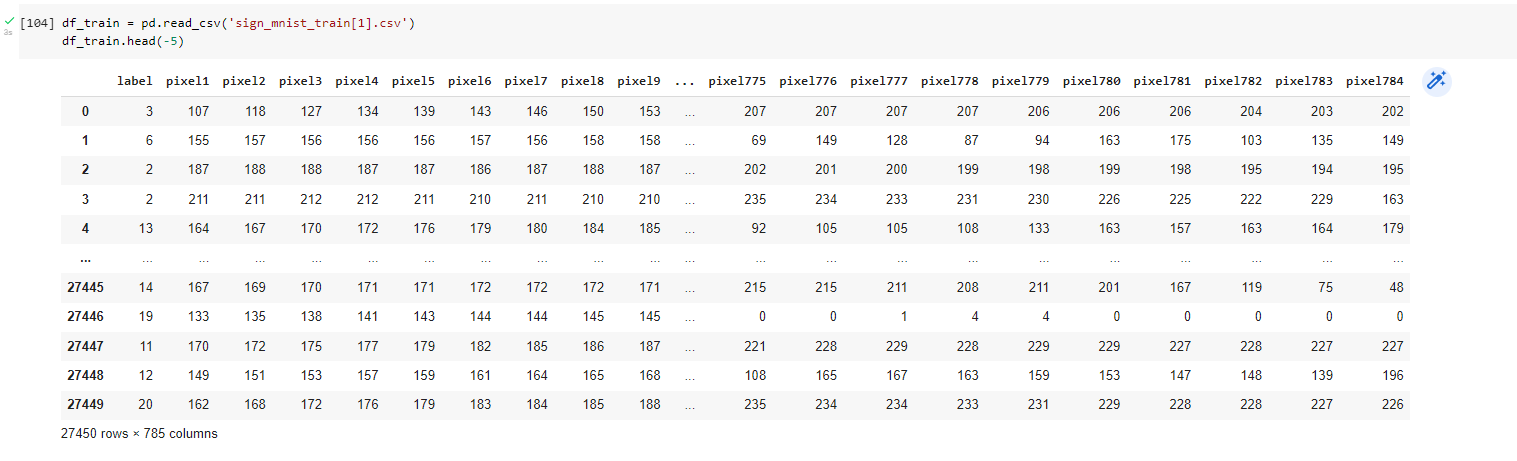
EXPERIMENTAL

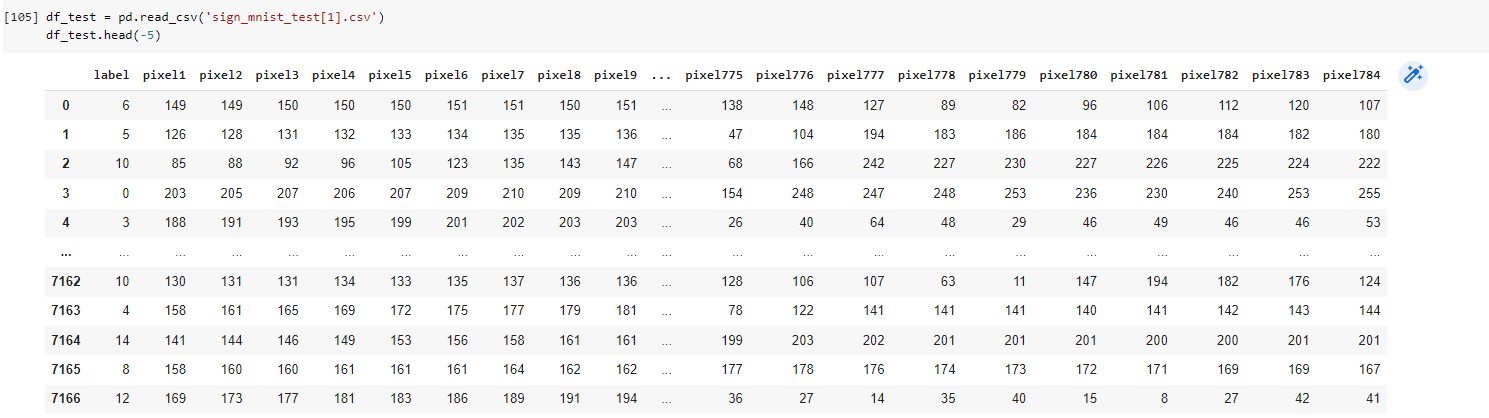
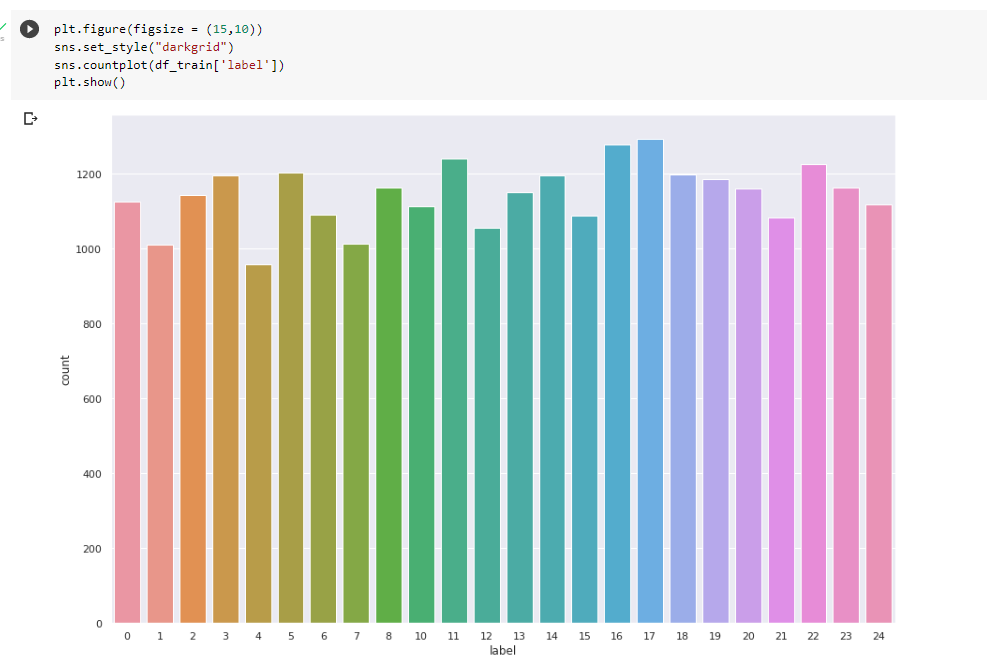
EVALUATION

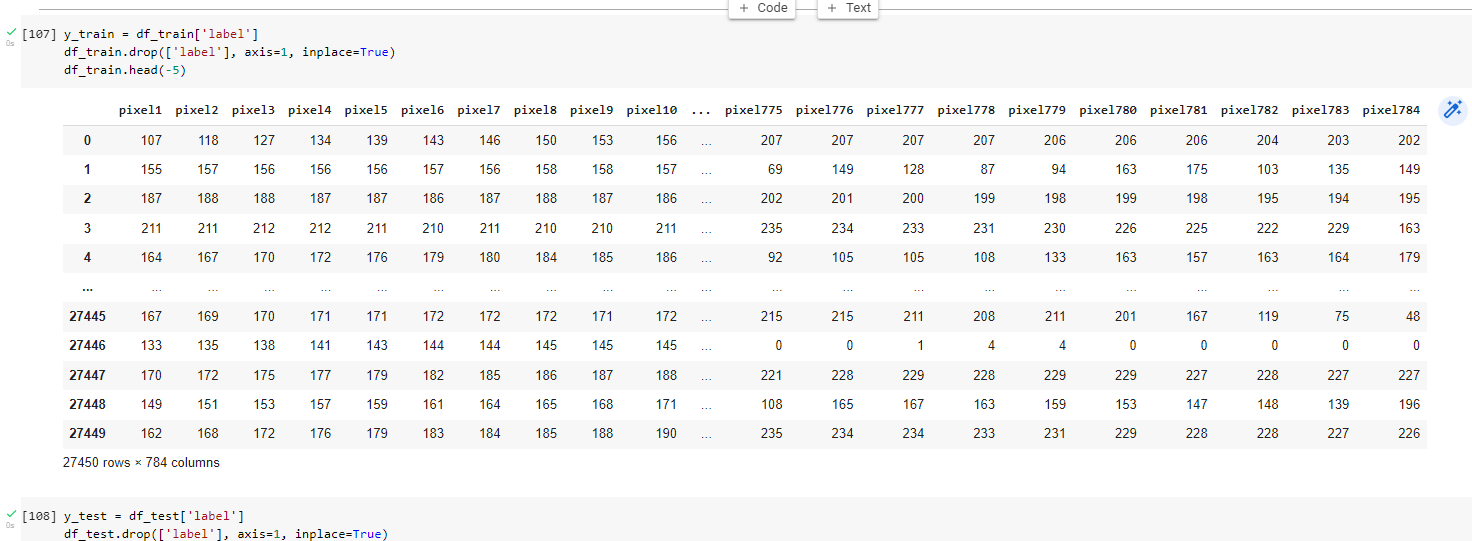
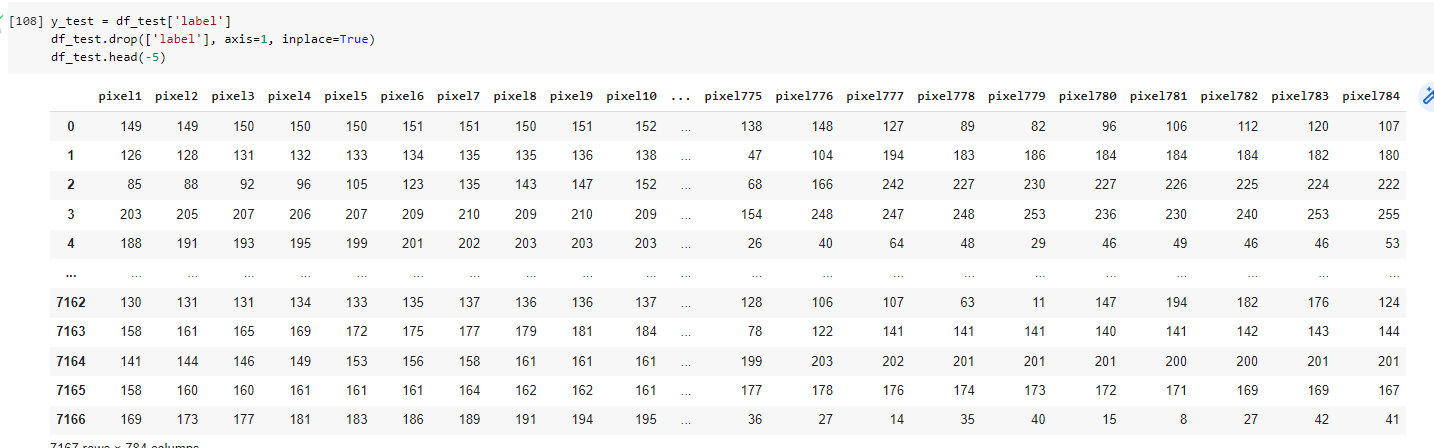
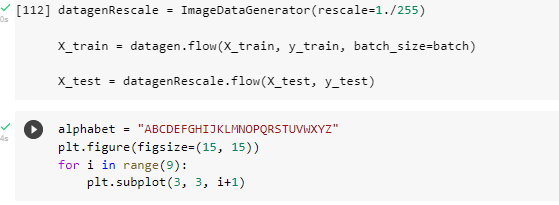
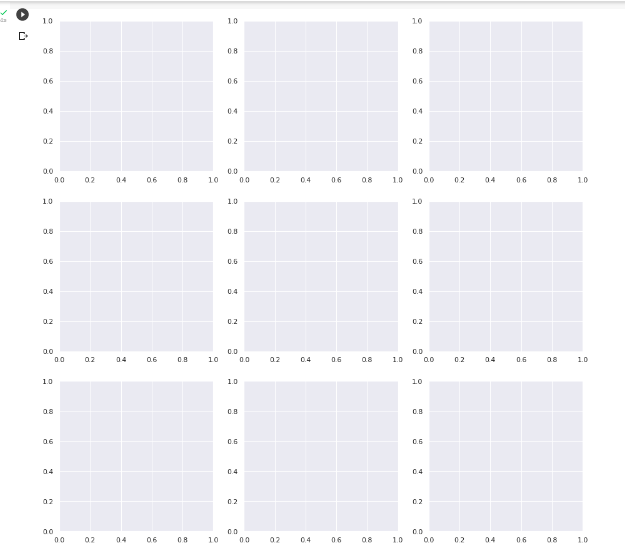
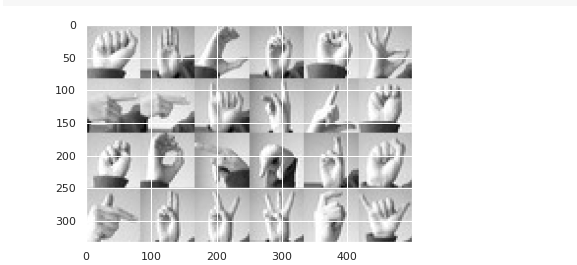
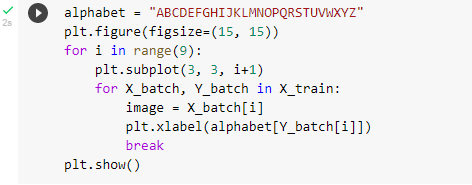
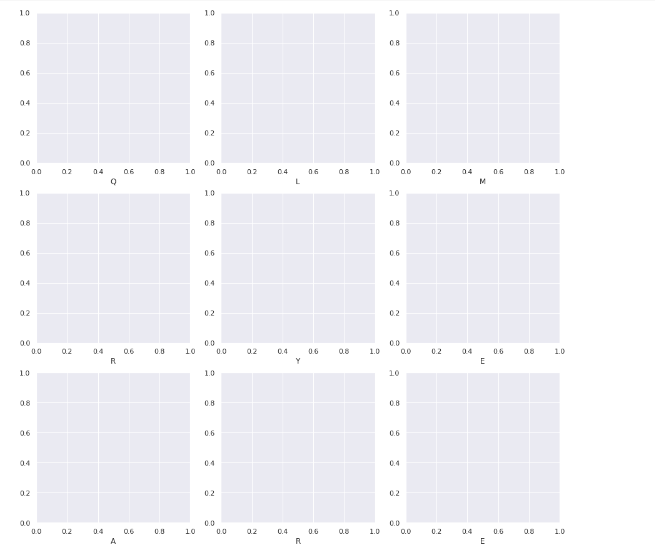
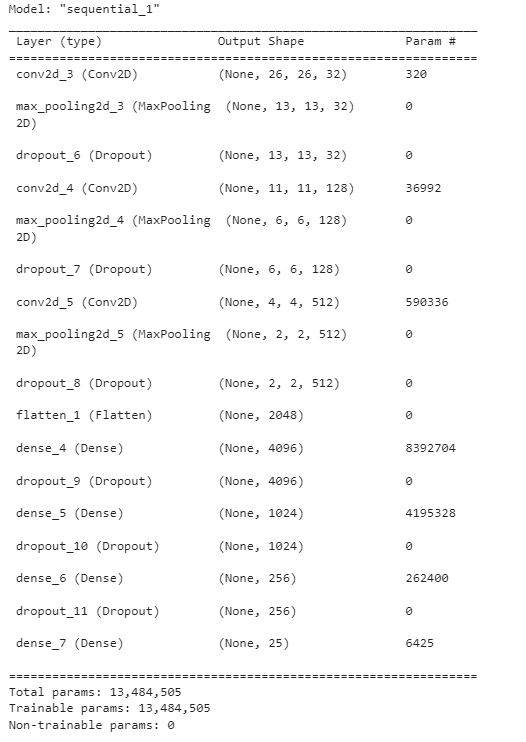
Our work attached with snapshots of the respective codes is systematically listed below:



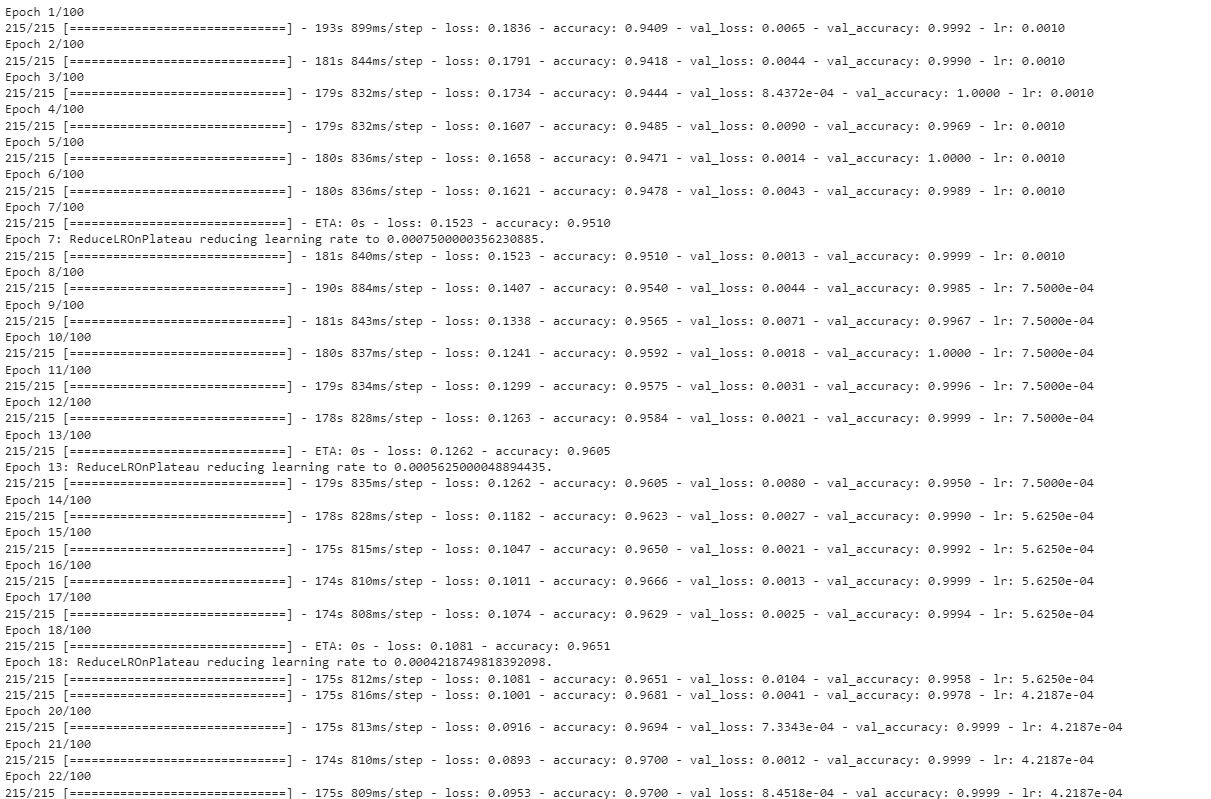
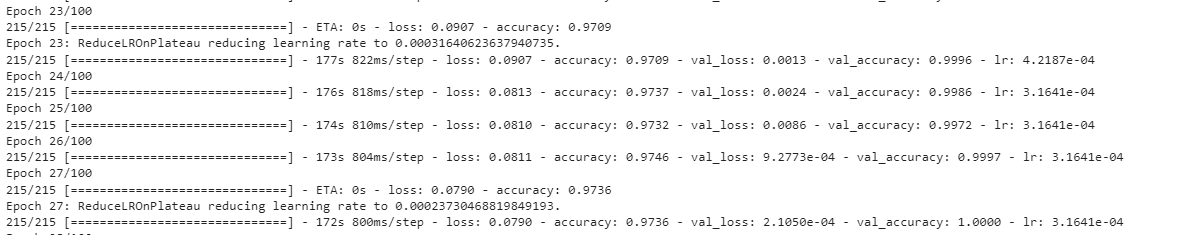
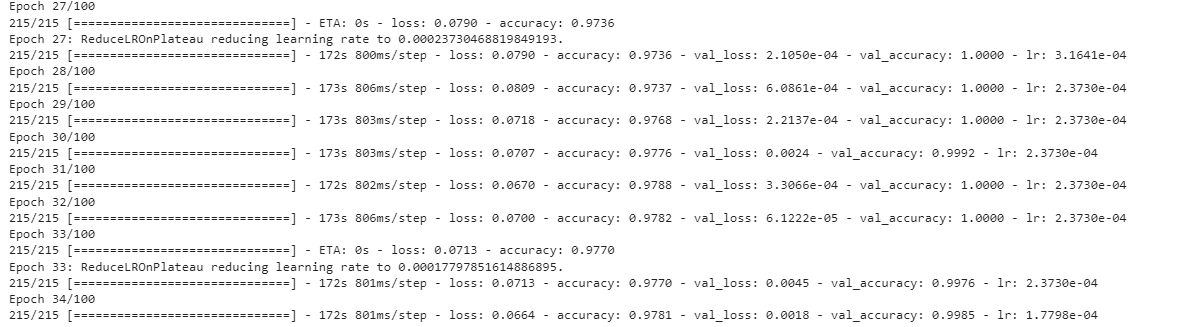
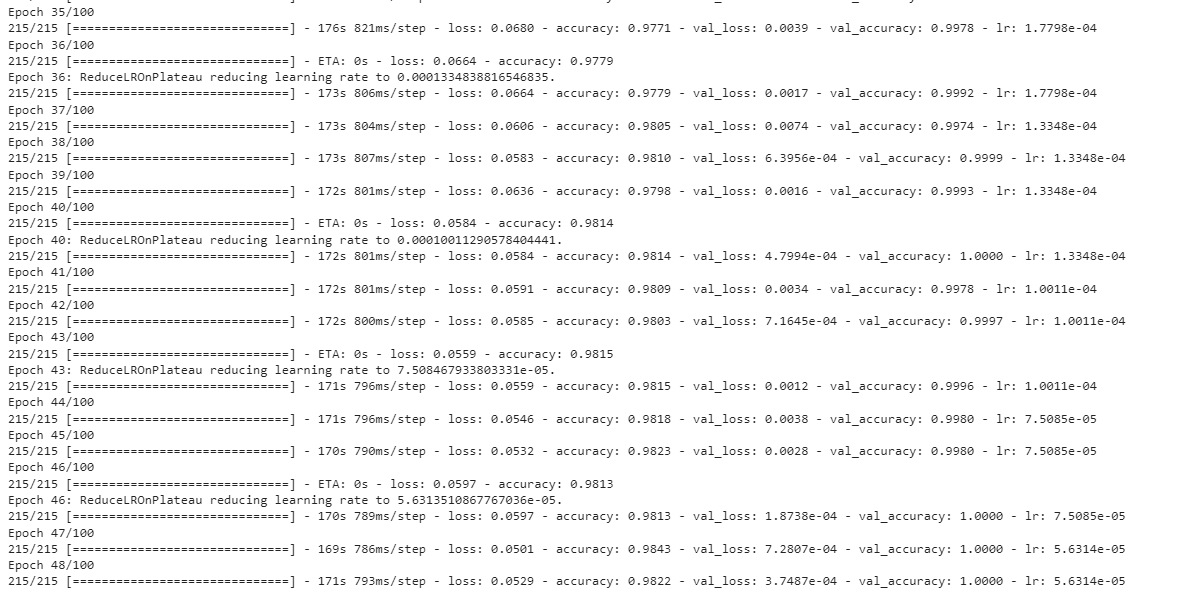
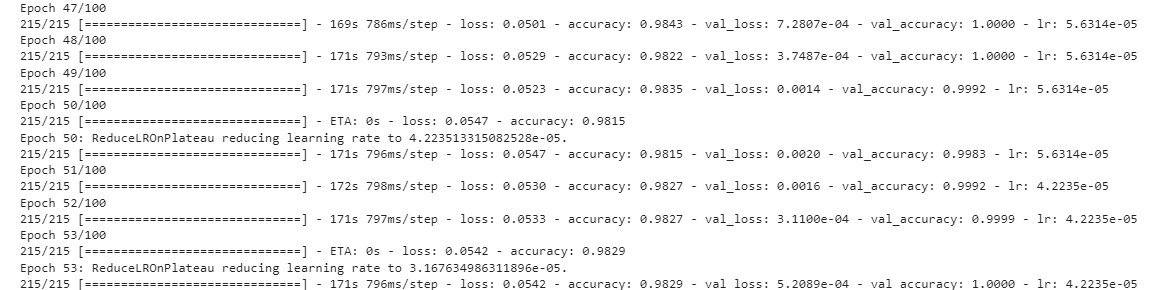
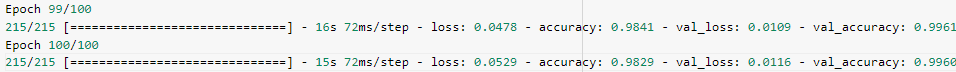
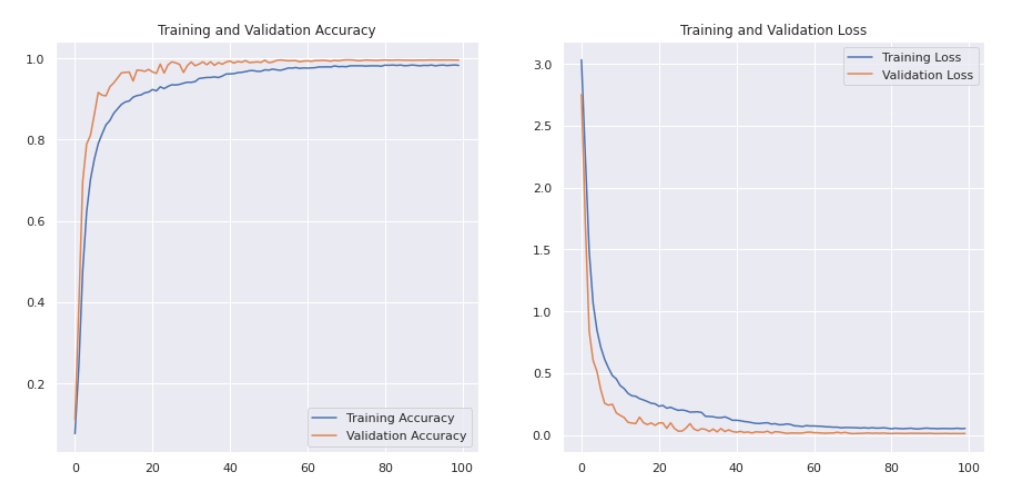
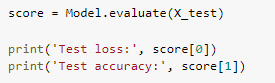
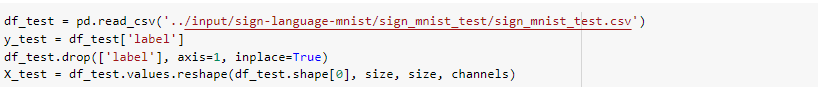
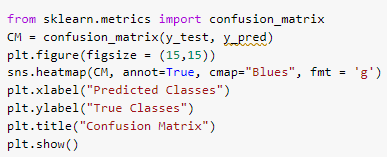
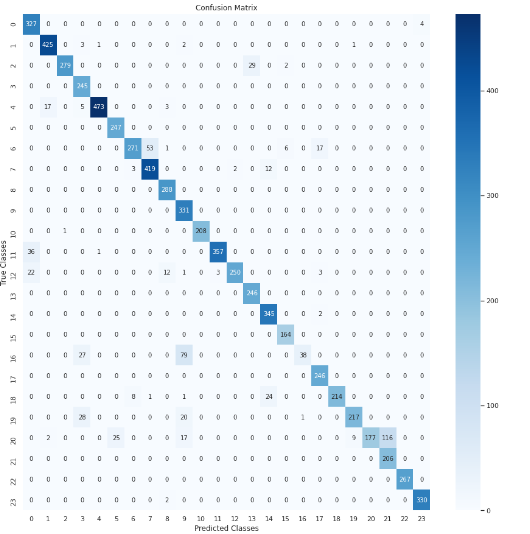


**** ****

**** ****

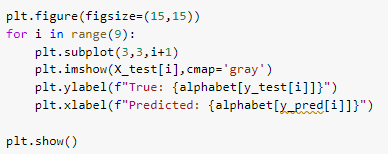
**** ****         

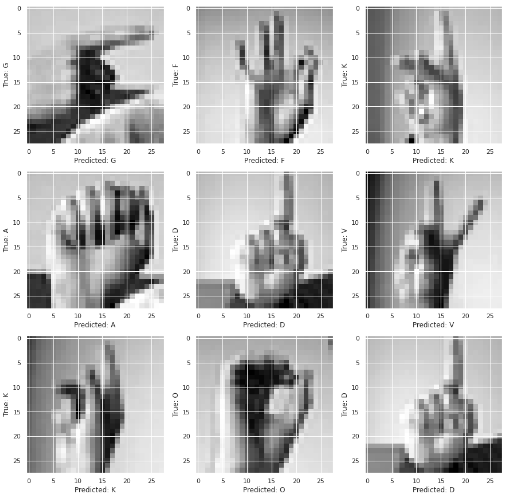


** **          

1. **RESULTS**

*The results we obtained are listed below:*

**

**

Pre-training with a user-dependent model:

FUTURE

WORK

Pre-training the model on a bigger dataset, such as the approximately 14,000-class ILSRVC, and fine-tuning it with the ISL dataset will enable the model to perform well even when trained on a small dataset. In the case of user-dependent models, the user will provide a set of photos to the model for training so that it may get to know the user. In this manner, the model will function effectively for a specific user.

CONCLUSION

We come to the conclusion that classification techniques for sign language recognition can be employed with SVM and convolutional neural networks. However, in order to demonstrate an increase in accuracy, pre-training must be carried out with a larger dataset. We were able to surpass the accuracy reported in earlier literatures by 71.88% using SVM for the ISL dataset using depth pictures dataset when 4 subjects were utilised for training and a different subject for testing.