



# SIGN LANGUAGE RECOGNITION

ML PROJECT PROPOSAL

**Group Members:**

Omkar Satupe (9232)

Ryan Valiaparambil (9237)

Mahek Intwala (9423)

# AGENDA



**1**

Introduction

**2**

Problem Statement

**3**

Scope & Dataset

**4**

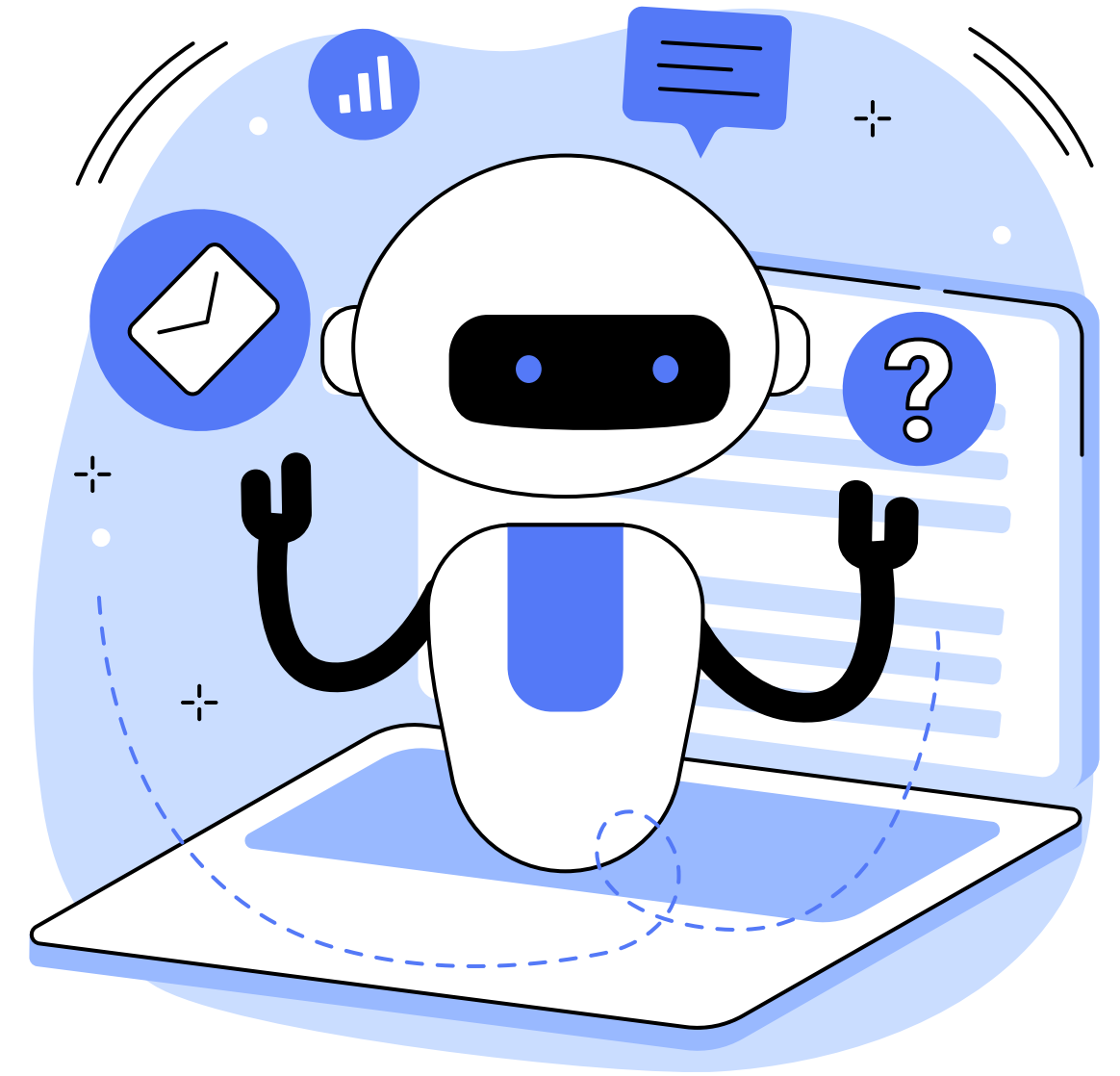
Literature Review

**5**

Methodology

# INTRODUCTION

- Sign Language is used by the deaf and speech-impaired to communicate through hand gestures.
- Sign Language Recognition (SLR) aims to identify these gestures and convert them into text or speech.
- Deep Learning, particularly Convolutional Neural Networks (CNNs), can be employed for hand gesture recognition.
- The model learns to identify gestures from images and translates them into English alphabets or numbers.
- This approach enhances communication for the deaf and speech-impaired, improving efficiency and accessibility.



# PROBLEM STATEMENT

To create a machine learning model that will be able to recognize the hand signs from images in the American Sign Language format

## Objectives:

- To find a dataset having sufficient images following the ASL format.
- To study about Neural Networks and how it can be used.
- To split the dataset into training and testing sets.
- To learn about Transfer Learning.
- To create a model which gives a good accuracy.
- To test the model's accuracy on unseen(test) data

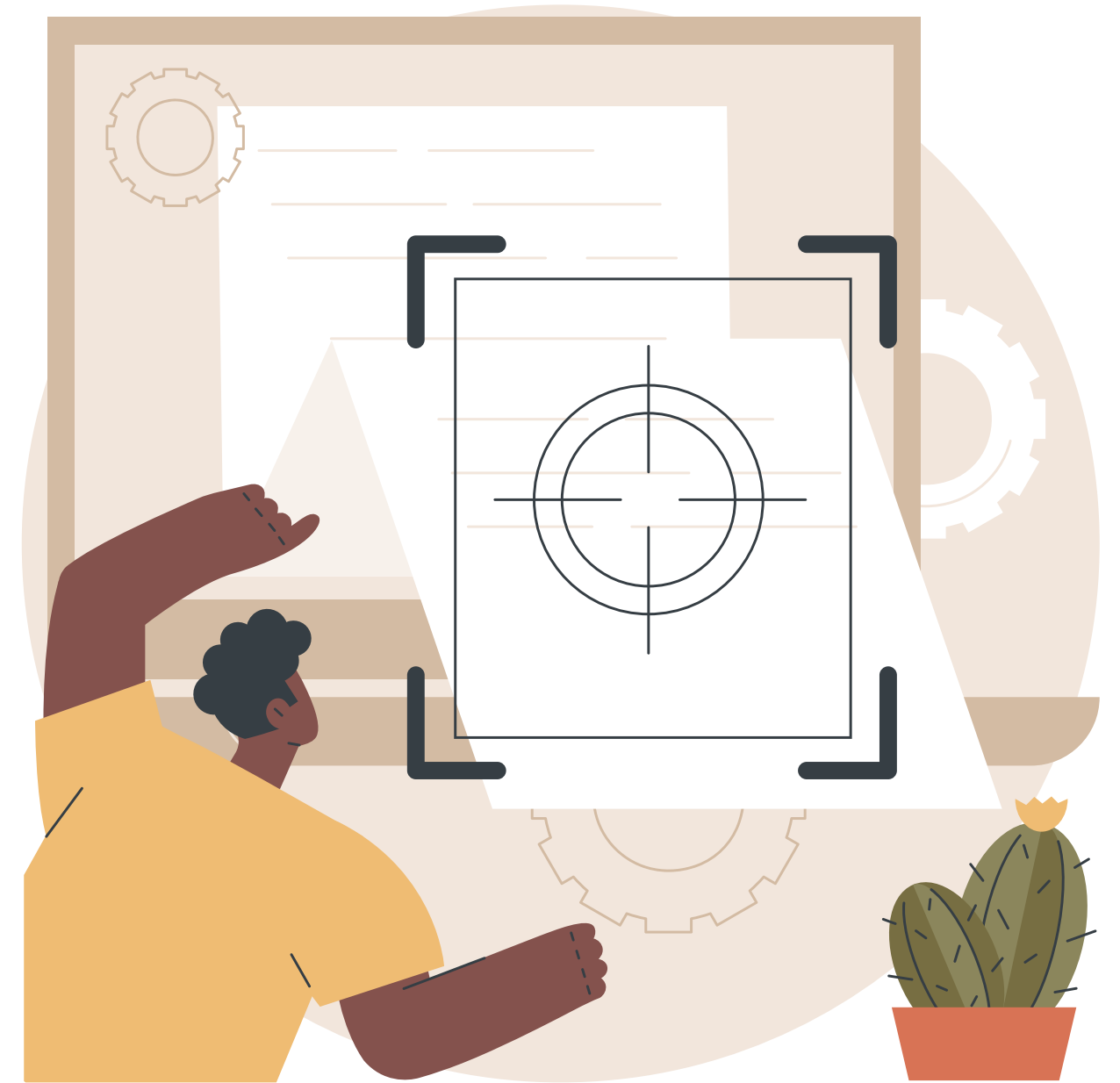


# SCOPE

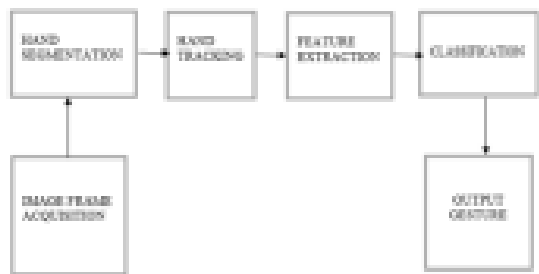
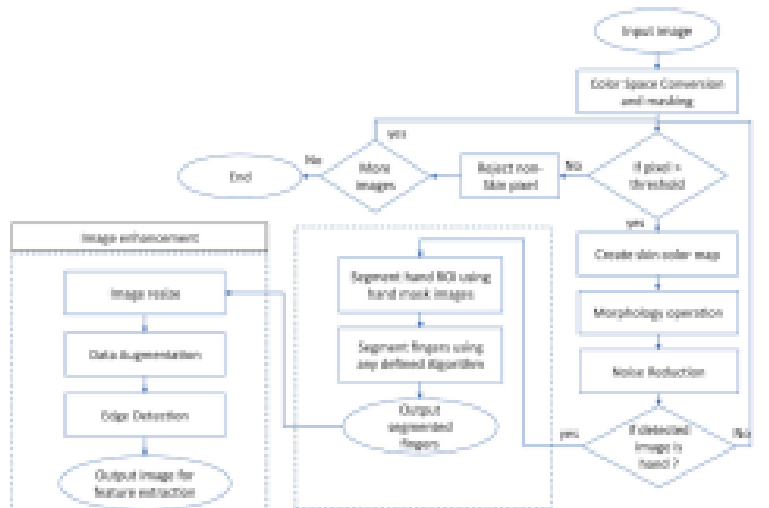
- This project has the potential to make a significant impact on the lives of individuals who rely on ASL for communication.
- It can be extended further by adding features like real-time sign language recognition using webcams or smartphone cameras, and integrating it into assistive technologies.

# DATASET

[For Dataset Click here](#)

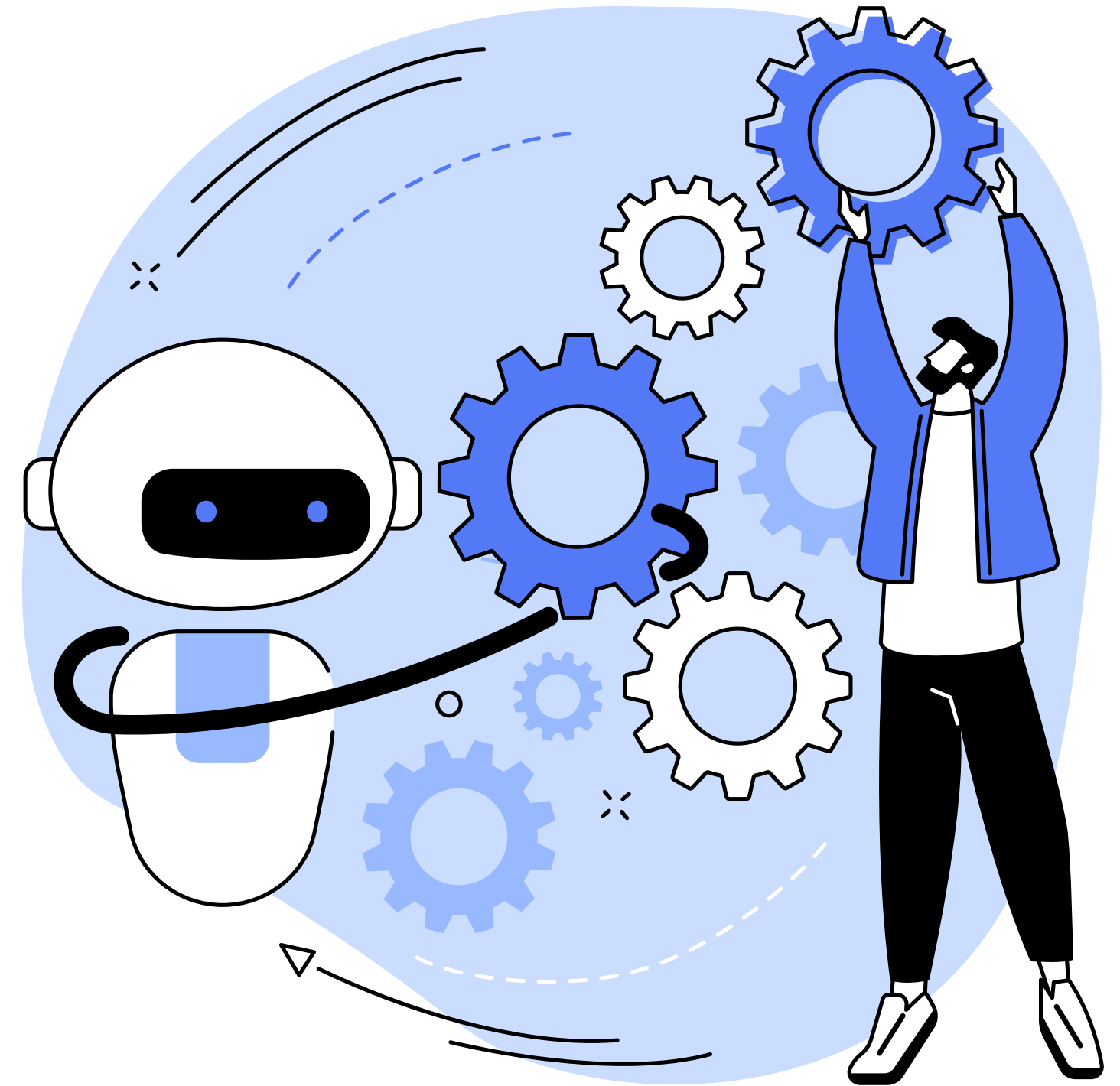


# LITERATURE SURVEY

Parameters \ Papers	Sign Language Recognition 1 Satwik Ram Kodandaram, 2N Pavan Kumar 3 Sunil G L	Real Time Sign Language Detection	Sign Language Recognition System using TensorFlow Object Detection API	A Review Paper on Sign Language Recognition for The Deaf and Dumb	Sign language identification and recognition: A comparative study
Publication	Research Gate	Research Gate	International Conference on Advanced Network Technologies and Intelligent Computing	IJERT	De Gruyter
Year of Publication	2021	2022	2021	2021	2022
Language	American Sign Language	Sign Language Gestures	Indian Sign Language	American Sign Language	American Sign Language
Algorithm	CNN & Transfer Learning	Transfer Learning & ConvNets	Transfer Learning	CNN	KNN, CNN, SVM
Transfer Learning Model used	LeNet-5, MobineNetV2,Ensemble	SSD Mobile net V2	SSD Mobile net V2	-	-
Tools Used	OpenCV, Django	Object Detection API, OpenCV, Labelling	Object Detection API, OpenCV, TF Record		
Methodology	Get the dataset having images of same size. Preprocess the image by removing noise and masking the hand. Give the input image to CNN model or Transfer Learning model.	Used Object Detection API to find the objects (hands) in an image and then labelled them as to what sign that gesture is using Labelling. Then similar Transfer Learning process was done.	Own dataset created and similar process followed.		
Flowchart					
Accuracies Achieved	MobileNetV2 98.9% LeNet-5 97% Own Model 98% Ensemble 99.8%	70-80%	Average Confidence Rate of 85.45%.	92%	accuracy of 97.62% for CNN, 78.95% for KNN, and 70.25% for SVM
Links	<a href="https://www.researchgate.net/publication/354066737_Sign_Language_Recognition">https://www.researchgate.net/publication/354066737_Sign_Language_Recognition</a>	<a href="https://www.researchgate.net/publication/357622360_Real_Time_Sign_Language_Detection">https://www.researchgate.net/publication/357622360_Real_Time_Sign_Language_Detection</a>	<a href="https://arxiv.org/ftp/arxiv/papers/2201/2201.01486.pdf">https://arxiv.org/ftp/arxiv/papers/2201/2201.01486.pdf</a>	<a href="https://www.ijert.org/a-review-paper-on-sign-language-recognition-for-the-deaf-and-dumb">https://www.ijert.org/a-review-paper-on-sign-language-recognition-for-the-deaf-and-dumb</a>	<a href="https://www.degruyter.com/document/doi/10.1515/comp-2022-0240/html?lang=en">https://www.degruyter.com/document/doi/10.1515/comp-2022-0240/html?lang=en</a>

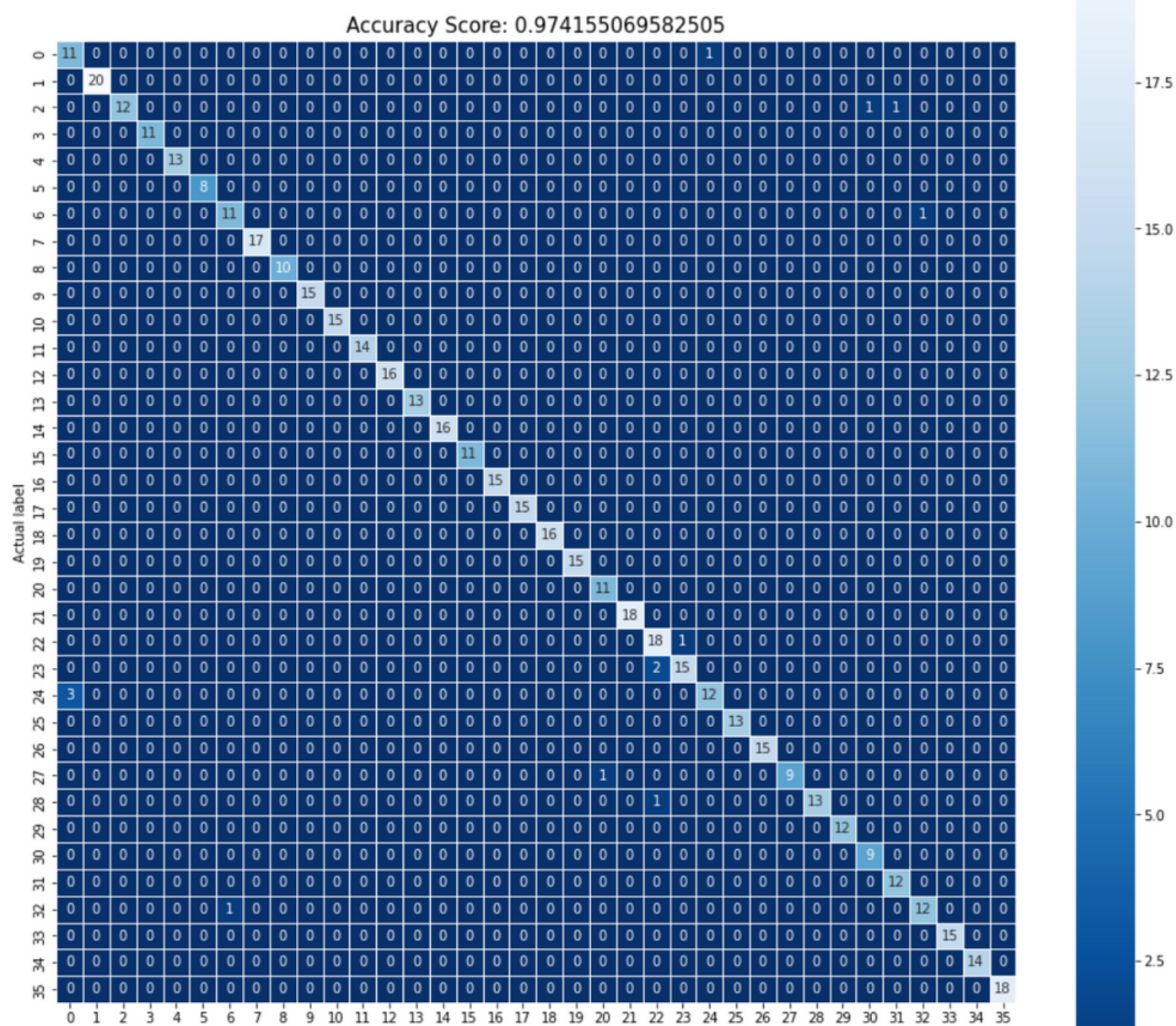
# METHODOLOGY

- Find a suitable dataset
- Preprocess the dataset.
- Split it into training and testing.
- Image Augmentation.
- Create a CNN Architecture.
- Fit the model on the training data.
- Try Transfer Learning and fit the data on that model.
- Check accuracy of the model on testing data.

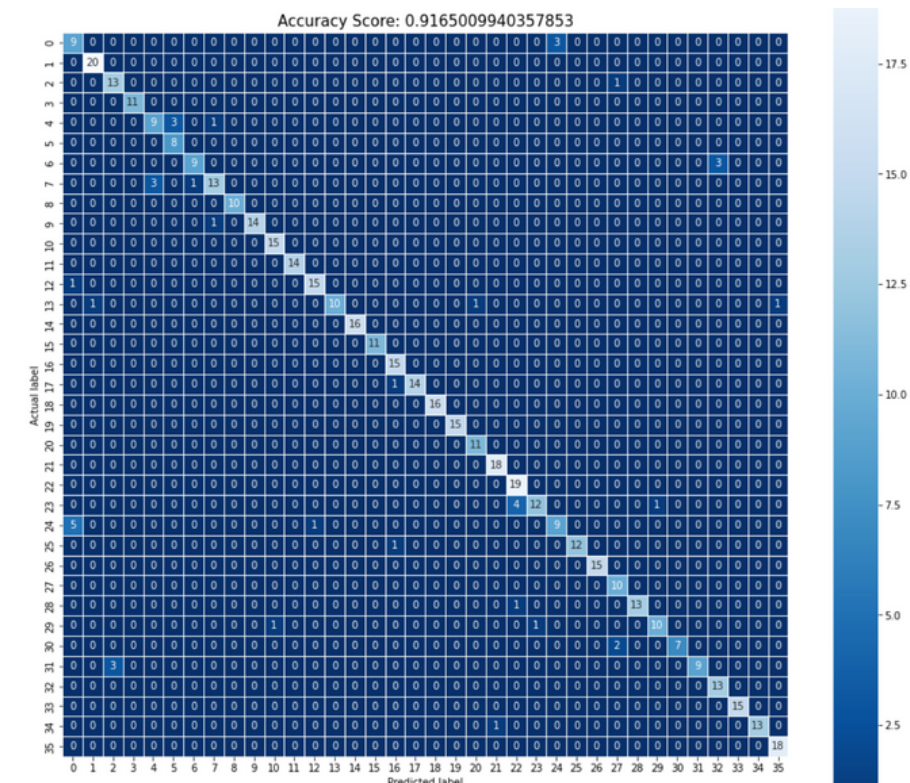




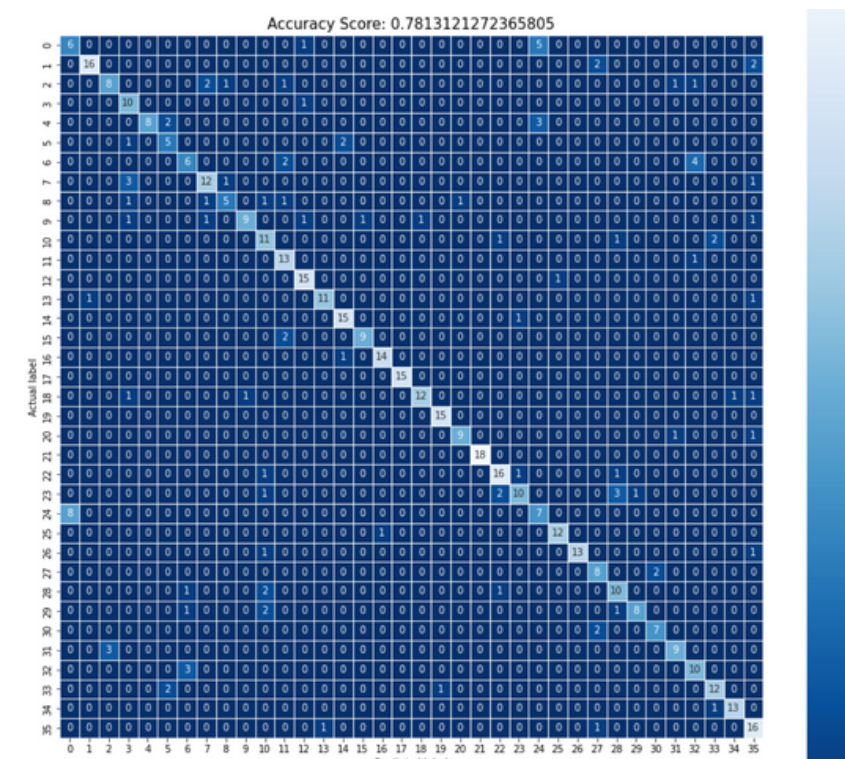
# IMPLEMENTATION ( ASL USING ML )



Logistic Regression



K nearest Neighbor



Decision tree



# IMPLEMENTATION ( ASL USING ML )

## Summary of the Models

```
: print(f"The Accuracy of KNN is {accuracy_score(y_test,predicted)}")  
  print(f"The Accuracy of Logistic Regression is {accuracy_score(y_test,y_pred)}")  
  print(f"The Accuracy of Decision Tree is {accuracy_score(y_test,ypred)}")
```

The Accuracy of KNN is 0.9165009940357853

The Accuracy of Logistic Regression is 0.974155069582505

The Accuracy of Decision Tree is 0.7813121272365805

---

# IMPLEMENTATION (ASL USING DL)

```
In [11]: imgs, labels = next(iter(train_set))
         counter = 1
         for img, label in zip(imgs, labels):
             plt.subplot(5,5,counter)
             plt.subplots_adjust(right=5, top=5, wspace=0.5, hspace=0.5)
             value=np.argmax(label)
             labelname=label_names[value]
             plt.imshow(img)
             plt.title("Image of: "+labelname, fontdict={'fontsize': 25})
             counter+=1
             plt.axis("off")
             if(counter>10):
                 break

         plt.show()
```

Image of: S



Image of: O



Image of: E

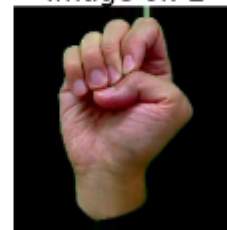


Image of: L

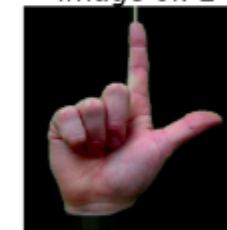


Image of: U



Image of: 2



Image of: P



Image of: G



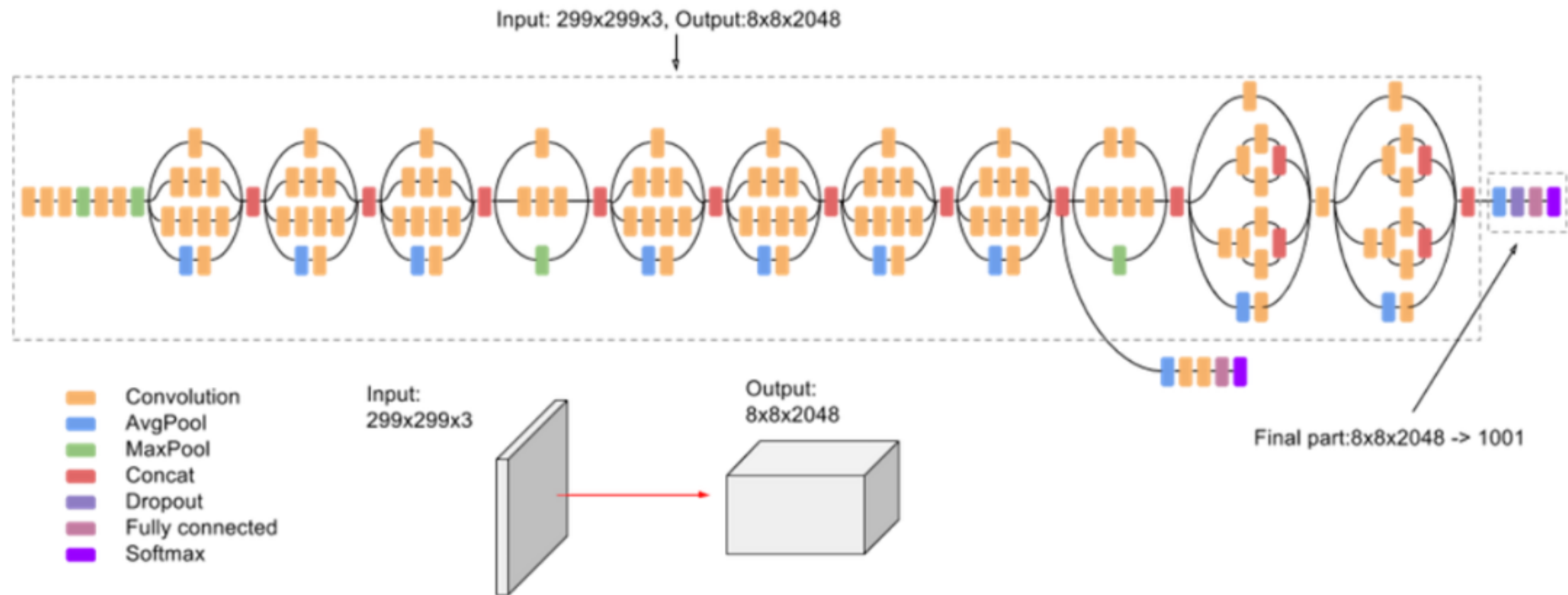
Image of: N



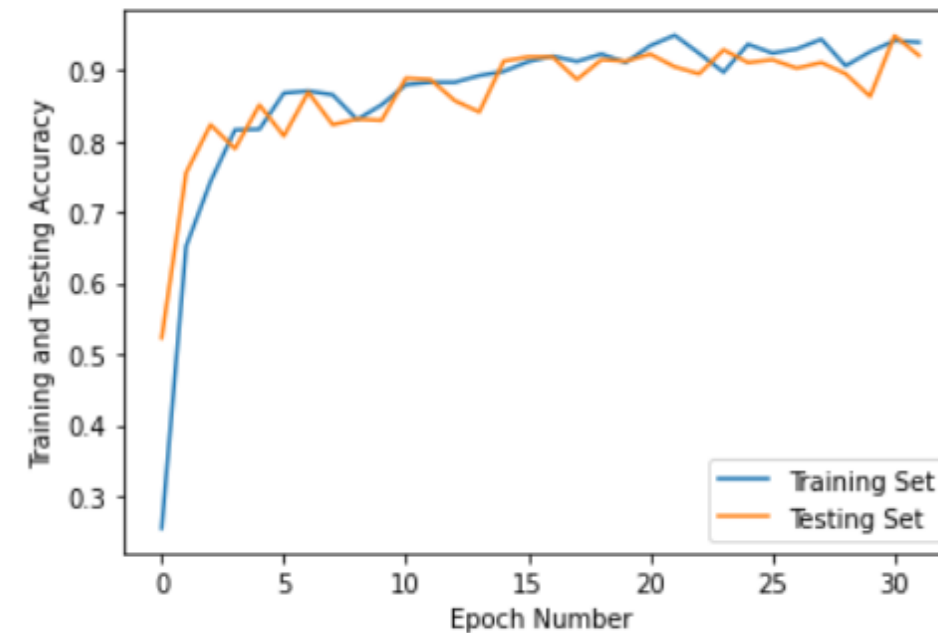
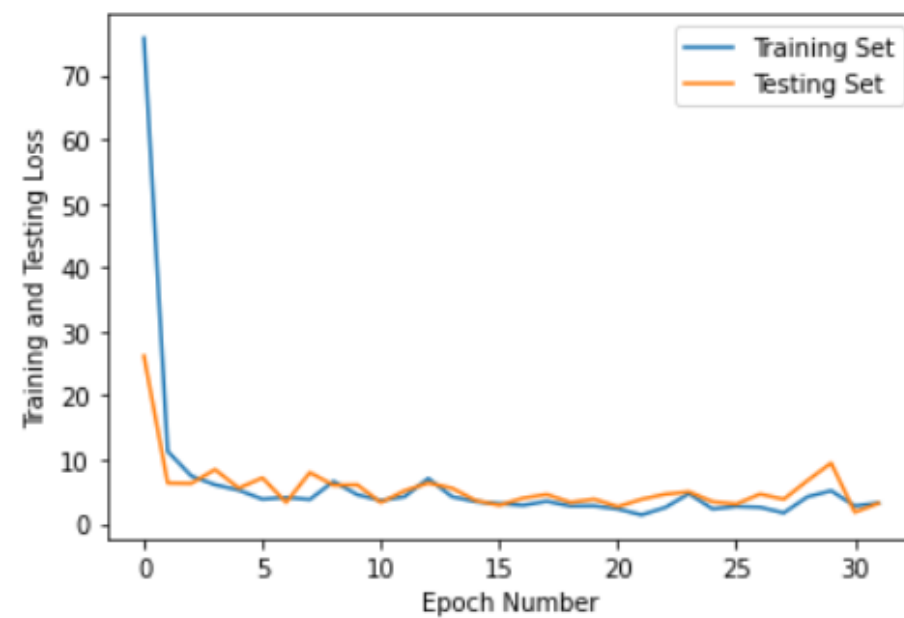
Image of: R



# IMPLEMENTATION (ASL USING DL)



# IMPLEMENTATION (ASL USING DL)



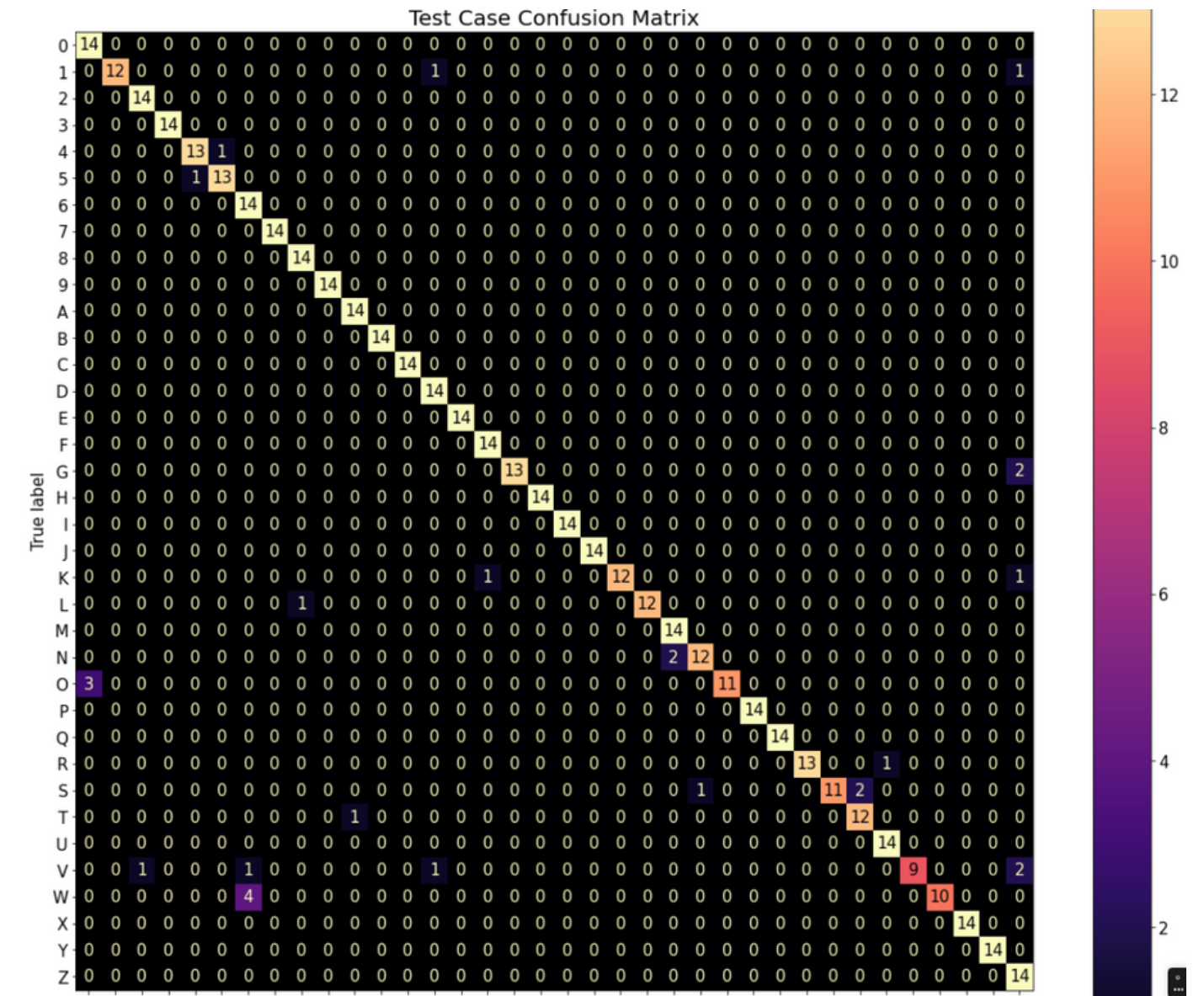
## Evaluating the model

```
] model.evaluate(test_set)
```

```
16/16 [=====] - 60s 4s/step - loss: 3.9102 - categorical_accuracy: 0.9264
```

```
] [3.9102275371551514, 0.9264413714408875]
```

Thus we have created a model that recognizes the hand signs based on the ASL with an accuracy of 92.64% and a loss of 3.91





**THANK YOU !**