# Sign Language Recognition

**Project Phase I (CSD 401)**
**End Semester Examination (November 18, 2021 , Thursday)**

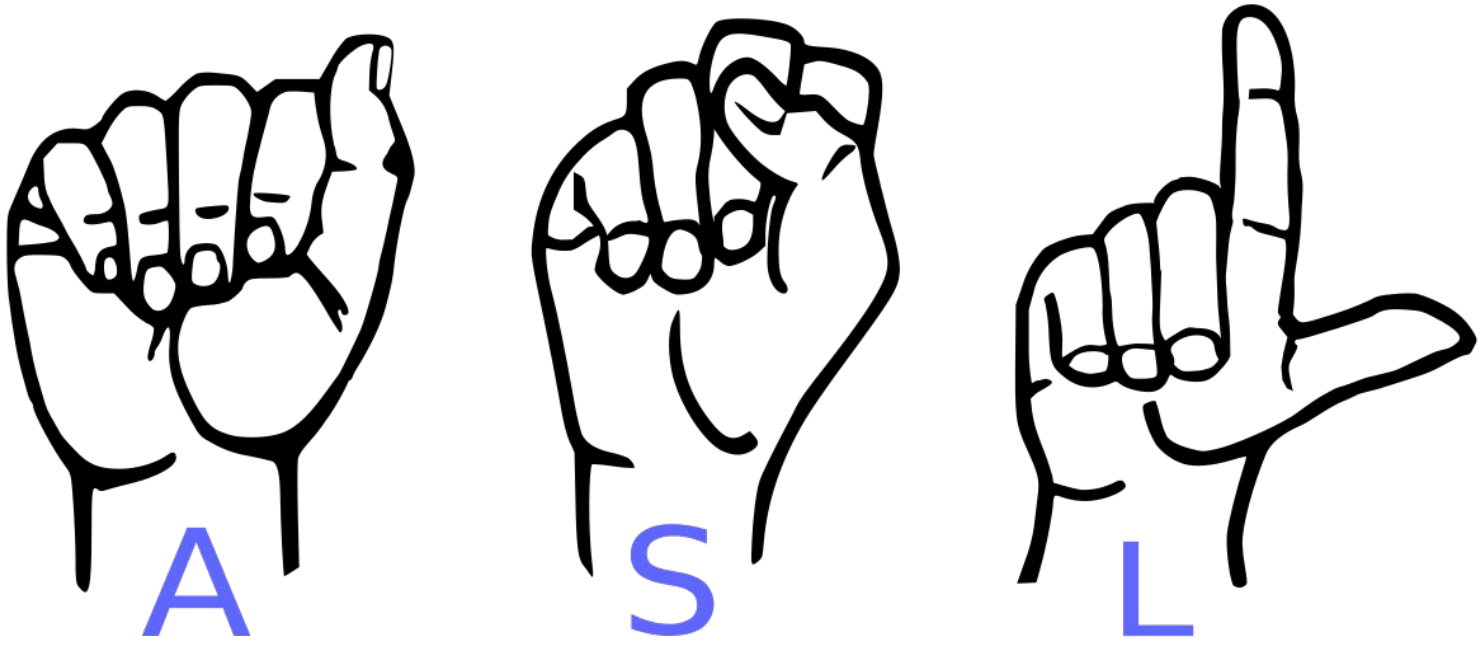

**PROJECT TEAM MEMBERS**

1. Sanskruti Nakhale BT18CSE015
2. Niraj Agrawal BT18CSE035
3. Tanmay Ganvir BT18CSE036
4. Nisarg Gogate BT18CSE040
5. Ketan Sarode BT18CSE044

**NAME OF SUPERVISOR**

Dr. Umesh Deshpande

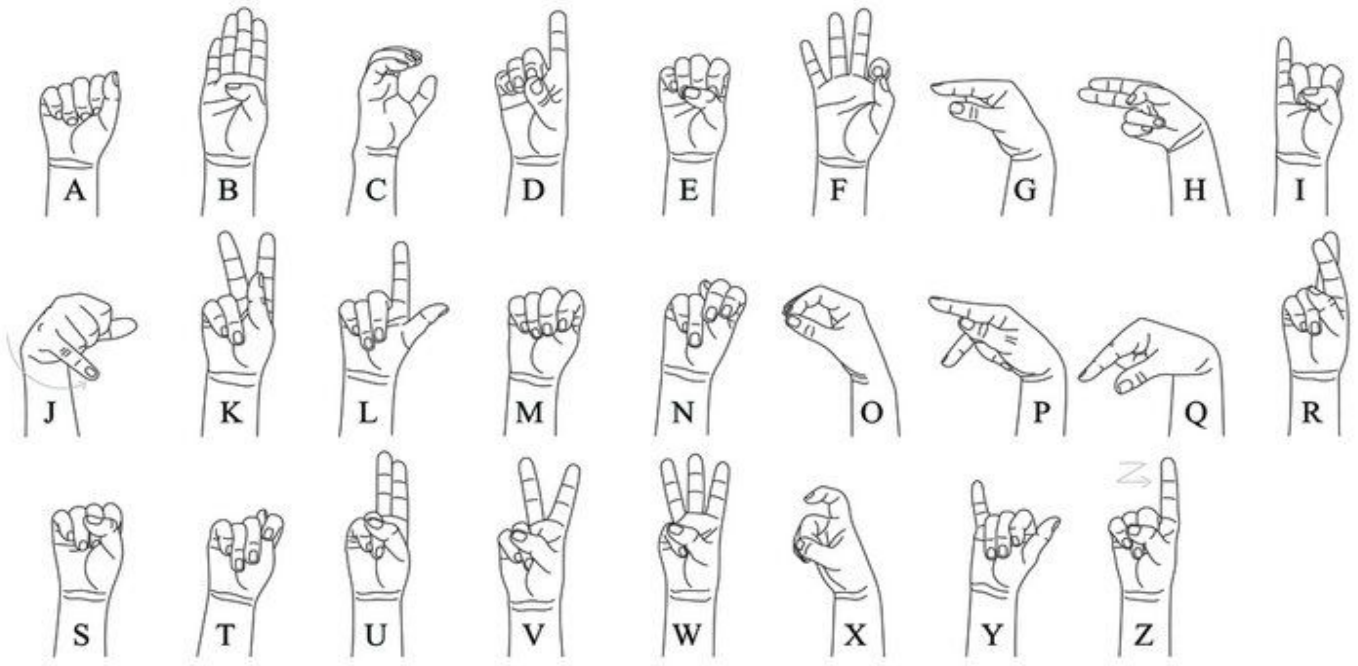
# Conversion of Sign Language to Text for deaf and dumb



# Aim

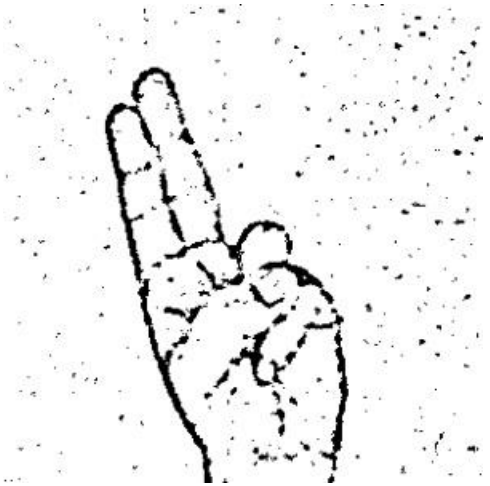
Our project aims to create a computer application and train a model which when shown a real time video of hand gestures of **American Sign Language** shows the output for that particular sign in text format on the screen.

# Our model Classifies ASL into 27 categories (A-Z + Blank).



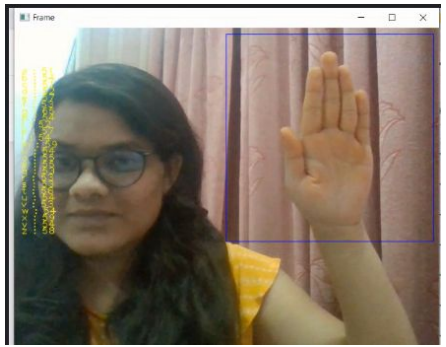
# Why we Created our own Dataset ?

- Existing Dataset was little skewed.
- We tried adding low light images to our training and testing dataset. (But low accuracy)



# Preprocessing

**Capturing Raw Image**




**Gray Scale Image**

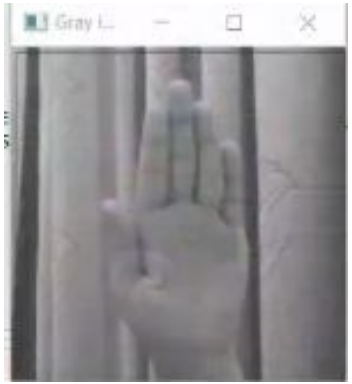
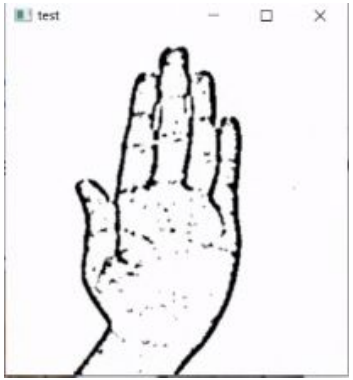
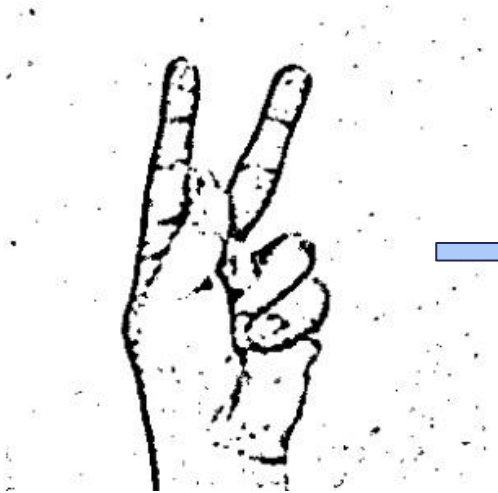



**Image Post Gaussian Blur and thresholding**



# Gesture Classification



**Classifier 1**

Classify between
27 Symbols

**Classifier 2**

Classify between
Similar Symbols

Classifier 1: Classified from [A,B...Z,Blank]
Classifier 2: Classified from [S,M,N]

# OverView

Input

Classifier 1

**M**

**L**

Predicted Symbol

Classifier 2

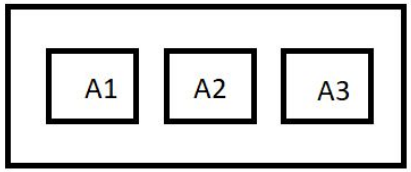
**S**

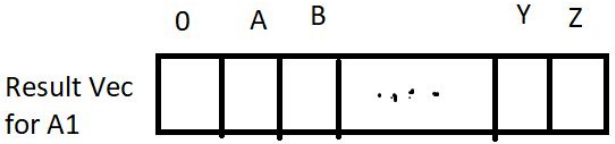
Predicted Symbol

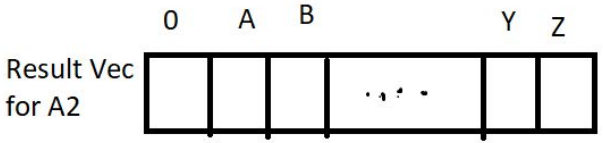
Classifier 2

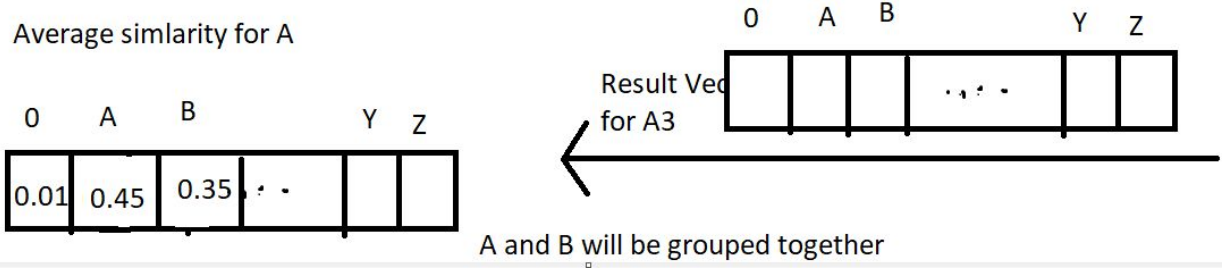
**Note:**
Thumb inside the fist for M
Thumb outside the fist for S.

# Working of grouping?



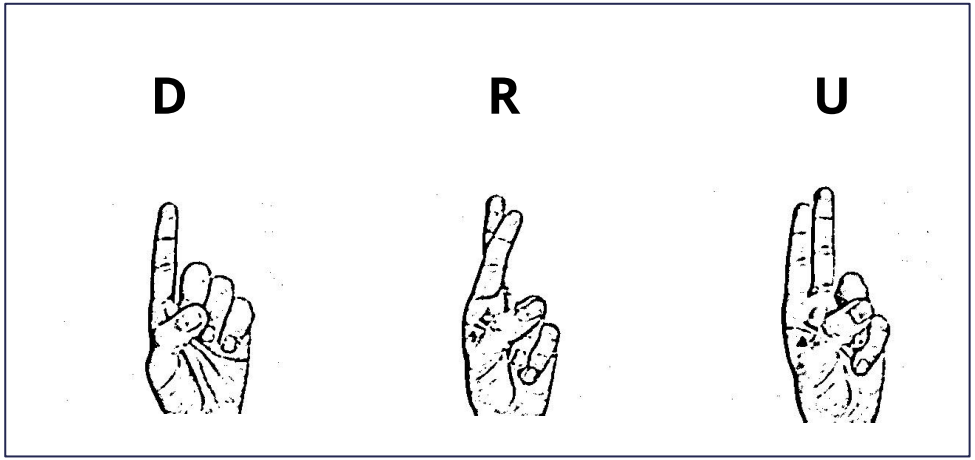



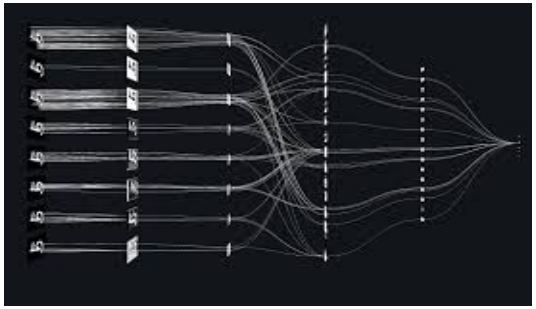
# Why Sub-Grouping

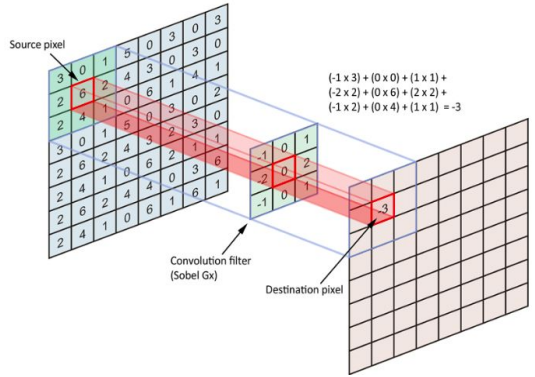
- We tried with single layer CNN but it was not accurate enough.
- So we decided to make Layer 2 model which classified groups of similar signs based on some threshold.
- Few groups observed are : **[K D I], [D R U], [S M N]**



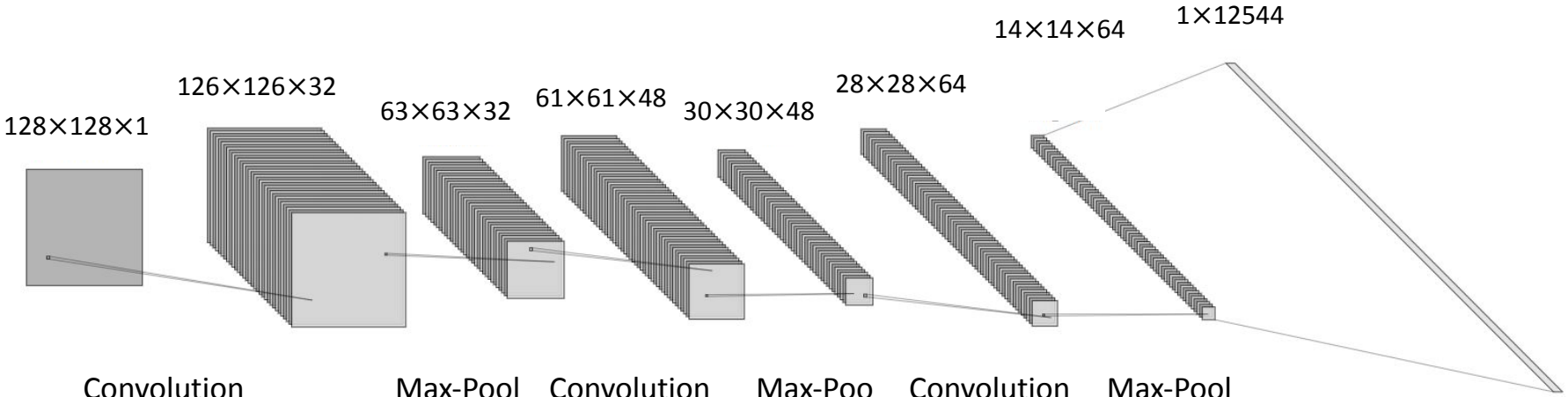
# Convolutional Neural Networks

- CNNs consist of multiple convolutional layers each layer containing numerous “filters” which perform feature extraction.
- Initially these “filters” are random and by training, the feature extraction gets better by better.
- It’s primarily used for image classification.





# CNN - Layer 1




Convolution
Filter:-32
Kernel:-(3,3)
Activation:-ReLU

Max-Pool
(2,2)

Convolution
F:- 48
K:-(3,3)
Act :-ReLU

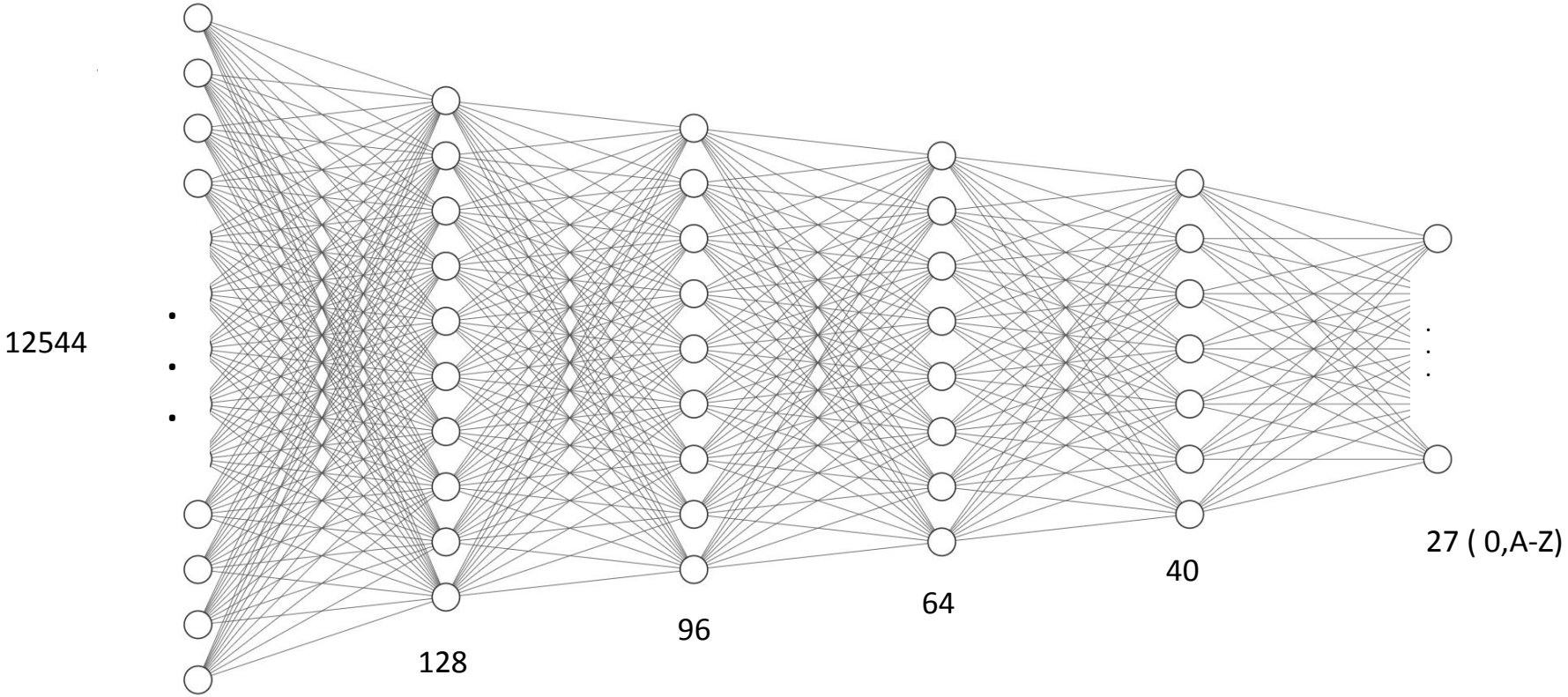
Max-Poo
(2,2)l

Convolution
F:- 64
K:- (3,3)
Act :-ReLU

Max-Pool
(2,2)

Flattened

# CNN - Layer 2 (Dense & Dropout Layers)



# Model Parameters:

- Max pooling to the input image with a pool size of (2, 2) with relu in each of the layers(convolutional as well as fully connected neurons).
- To avoid overfitting , we applied dropout of 40%.
- Adam optimizer for updating the model in response to the output of the loss function.
- Kernel size = (3,3) , Strides=(1, 1) and pool_size=(2, 2) is used
- Used Softmax func in last layer which classify our final 27 units ( 0 , A-Z )

| Layer (type) | Output Shape | Param # |
|---|---|---|
| conv2d_8 (Conv2D) | (None, 126, 126, 32) | 320 |
| max_pooling2d_8 (MaxPooling 2D) | (None, 63, 63, 32) | 0 |
| conv2d_9 (Conv2D) | (None, 61, 61, 48) | 13872 |
| max_pooling2d_9 (MaxPooling 2D) | (None, 30, 30, 48) | 0 |
| conv2d_10 (Conv2D) | (None, 28, 28, 64) | 27712 |
| max_pooling2d_10 (MaxPoolin g2D) | (None, 14, 14, 64) | 0 |
| flatten_3 (Flatten) | (None, 12544) | 0 |
| dense_15 (Dense) | (None, 128) | 1605760 |
| dropout_6 (Dropout) | (None, 128) | 0 |
| dense_16 (Dense) | (None, 96) | 12384 |
| dropout_7 (Dropout) | (None, 96) | 0 |
| dense_17 (Dense) | (None, 64) | 6208 |
| dense_18 (Dense) | (None, 40) | 2600 |
| dense_19 (Dense) | (None, 27) | 1107 |

Total params: 1,669,963
Trainable params: 1,669,963
Non-trainable params: 0

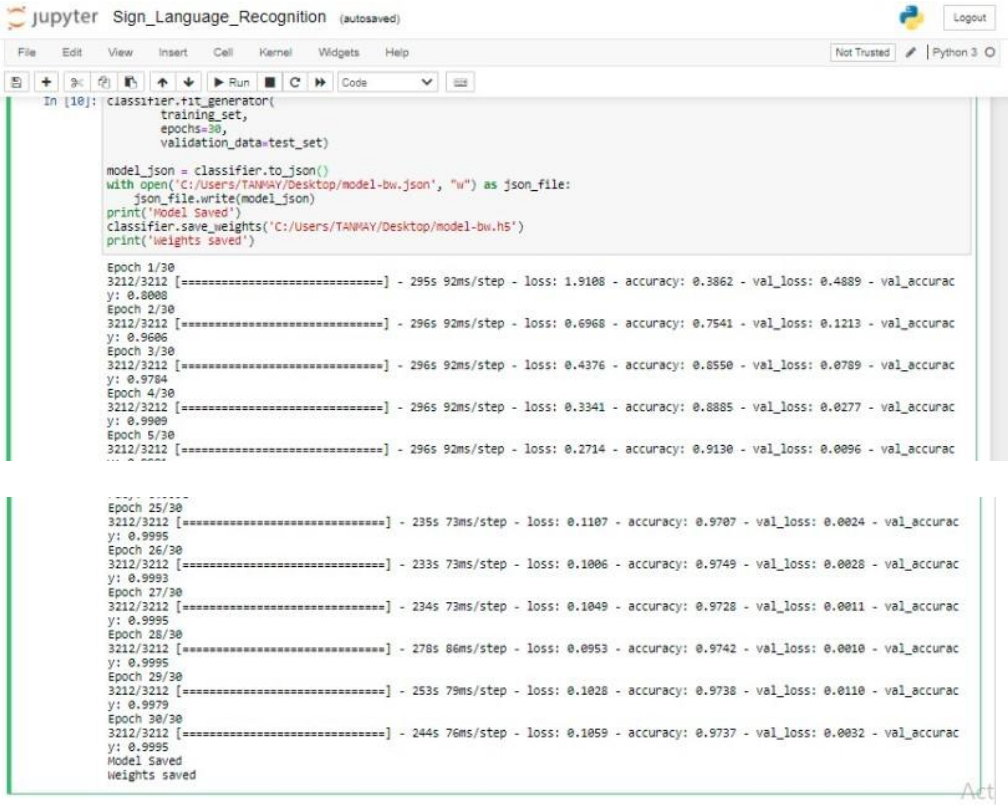
# Results:

**Layer 1 (After 30 epoch)**

- Accuracy : 0.973
- Step-Loss : 0.105
- Val-Loss : 0.003

**Layer 2**(Sub-groups):

- [K D I] : 0.992
- [D R U] : 0.994
- [S M N] : 0.990



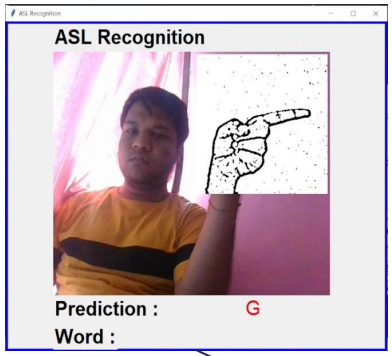
# Real Time Application:

- Apply gaussian blur filter and threshold to the frame taken with opencv to get the processed image.

- This processed image is then passed to the CNN model for prediction and if letter is detected for more than 25 frames then the letter is selected and taken into consideration for forming the word.

- Completion of  words are considered using the blank symbol.

# Application Demo




G

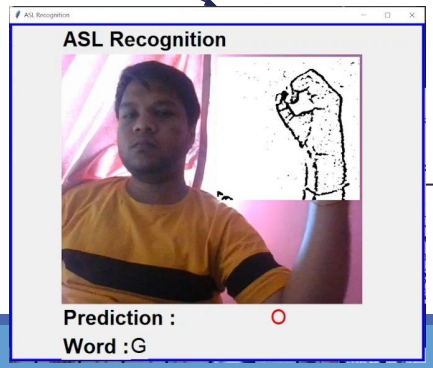
O

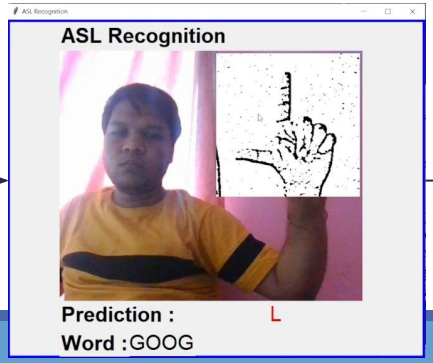
O

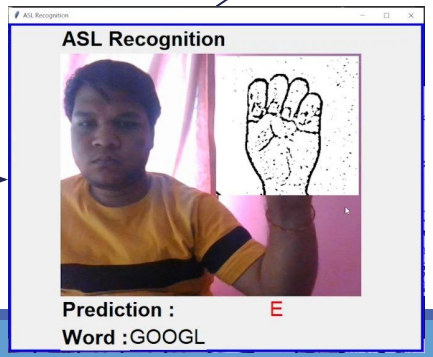
G

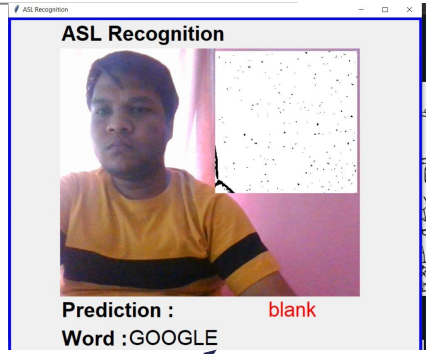
L

E



# Problems Faced

- We couldn’t find a dataset with raw images of all the ASL characters so we made our own dataset.
- Selection of filter for feature extraction. We tried various filter including binary threshold, canny edge detection, gaussian blur etc. ,of which **gaussian blur** filter was giving better results.
- We made our data set with low light images, but found our images too noisy so our model couldn’t give a good accuracy.
- We tried to find out the hand gestures for which accuracy was low and group those symbols together. But our algorithm was not giving a good enough grouping. So, we made groups by our observations.
- We didn’t rotate the images in data augmentation as it was leading to confusion within gestures.

# Future Work

- Our project is giving good accuracy with static images, we are planning to improve real time video input application.

- Currently, after getting similar groups we manually train for every group , we have to automate this.

- We are planning to work on other Sign Languages too like ISL with Higher Models like Yolo.


VISVESVARAYA NATIONAL INSTITUTE OF TECHNOLOGY
NAGPUR
ॐ: कर्मसु कौशलम्

# Thank You !!!

# References


[1]Manuel Eleazar Martínez-Gutiérrez, José Rafael Rojano-Cáceres, Edgard Benítez-Guerrero, Héctor Eduardo Sánchez-Barrera,"Data Acquisition Software for Sign Language Recognition" in Universidad Veracruzana, Facultad de Estadística e informática, Mexico

[2]Mathavan Suresh Anand, Nagarajan Mohan Kumar, Angappan Kumaresan, "An Efficient Framework for Indian SignLanguageRecognition Using Wavelet Transform" Circuits and Systems, Volume 7, pp 1874-1883, 2016

[3]Suharjitoa, Ricky Andersonb , Fanny Wiryanab , Meita Chandra Ariestab , Gede Putra Kusumaa ,"Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process-Output" in 2nd International Conference on Computer Science and Computational Intelligence 2017, ICCSCI 2017, 13-14 October 2017, Bali, Indonesia

[4]T. Yang, Y. Xu, and "A. , Hidden Markov Model for Gesture Recognition", CMU-RI-TR-94 10, Robotics Institute, Carnegie Mellon Univ.,Pittsburgh,PA, May 1994

# References

[5]Pigou L., Dieleman S., Kindermans PJ., Schrauwen B. (2015) “Sign Language Recognition Using Convolutional Neural Networks”. In: Agapito L., Bronstein M., Rother C. (eds) Computer Vision - ECCV 2014 Workshops. ECCV 2014. Lecture Notes in Computer Science, vol 8925. Springer, Cham

[6]Byeongkeun Kang , Subarna Tripathi , Truong Q. Nguyen ”Real-time sign language fingerspelling recognition using convolutional neural networks from depth map” 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)

[7]Zaki, M.M., Shaheen, S.I.: “Sign language recognition using a combination of new vision based features”. Pattern Recognition Letters 32(4), 572–577 (2011)