



**Department of Computer Science and Engineering**  
**Motihari College of Engineering, Motihari**

# Conversion of Sign Language to Sentence And Audio

For Dumb , Deaf And Blind



*PRESENTED BY:-*

**NITESH KUMAR**      19105113029

**SANISH KUMAR**      20105113906

**RAJ ARYAN**      19105113059

**ASHUTOSH KUMAR**      19105113006

*UNDER THE SUPERVISION  
OF:*

**Mrs. Abha Kumari**  
[Assistant Professor]




# Contents

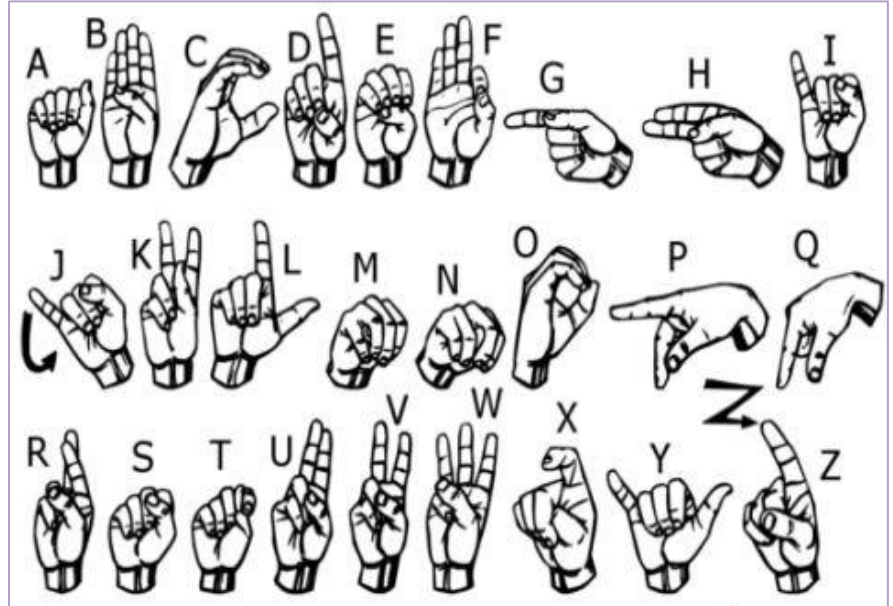
1. Abstract
2. Sign language is a visual language and consists of 3 major components
3. We implemented 26 symbols(A-Z, blank) of ASL in our project.
4. Software Requirements
5. How we generated data set and did Data Preprocessing ?
6. Why we Created our own Dataset ?
7. Gesture Classification
8. Convolutional Neural Networks
9. Our CNN Classifier Model
10. Finger Spelling Sentence Formation
11. Autocorrect feature
12. Challenges Faced
13. Limitations of our model
14. Conclusion
15. Future Scope

---

# Abstract

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.

  
We implemented  
26 symbols(A-Z,  
blank) of ASL in our  
project.





# Software Requirements

- Python 3.9.13
- Tensorflow 2.12.0
- OpenCV 4.7.0
- NumPy 1.23.5
- cyhunspell 2.0.2
- Keras 2.12.0
- PIL 9.2.0

**How we generated  
data set and did Data  
Preprocessing?**

---

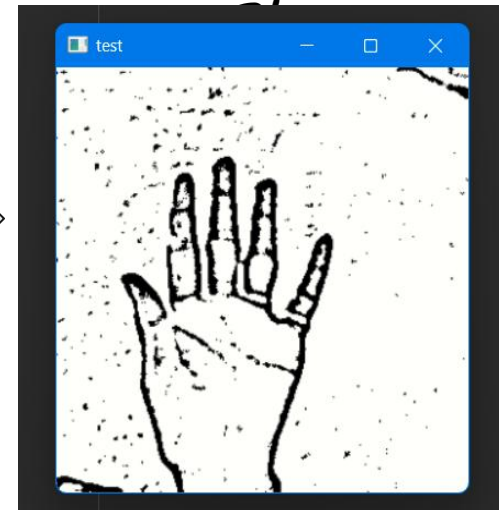
Capturing Raw  
Image



Gray Scale  
Image



Image Post  
Gaussian



# **Why we Created our own Dataset ?**

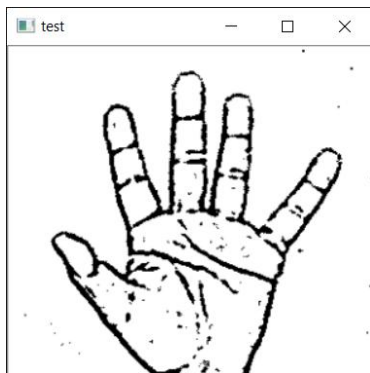




- For the project we tried to find already made datasets but we couldn't find dataset in the form of raw images that matched our requirements.
- All we could find were the datasets in the form of RGB values.
- Hence we decided to create our own data set.

# Gesture Classification





Layer  
1  
Classify  
between  
26  
Symbols



Layer 2  
Classify  
between  
Similar  
Symbols



## Algorithm Layer 1:

1. Apply gaussian blur filter and threshold to the frame taken with opencv to get the processed image after feature extraction.
2. This processed image is passed to the CNN model for prediction and if a letter is detected for more than 50 frames then the letter is printed and taken into consideration for forming the word.
3. Space between the words are considered using the blank symbol.

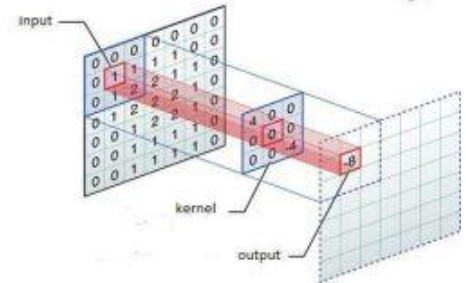
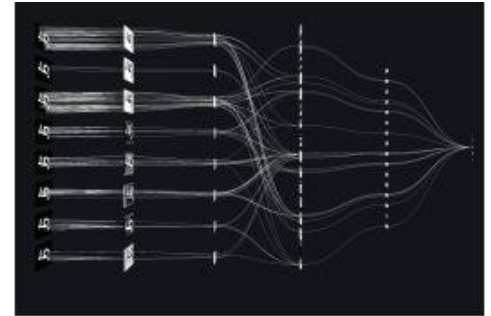


## Algorithm Layer 2:

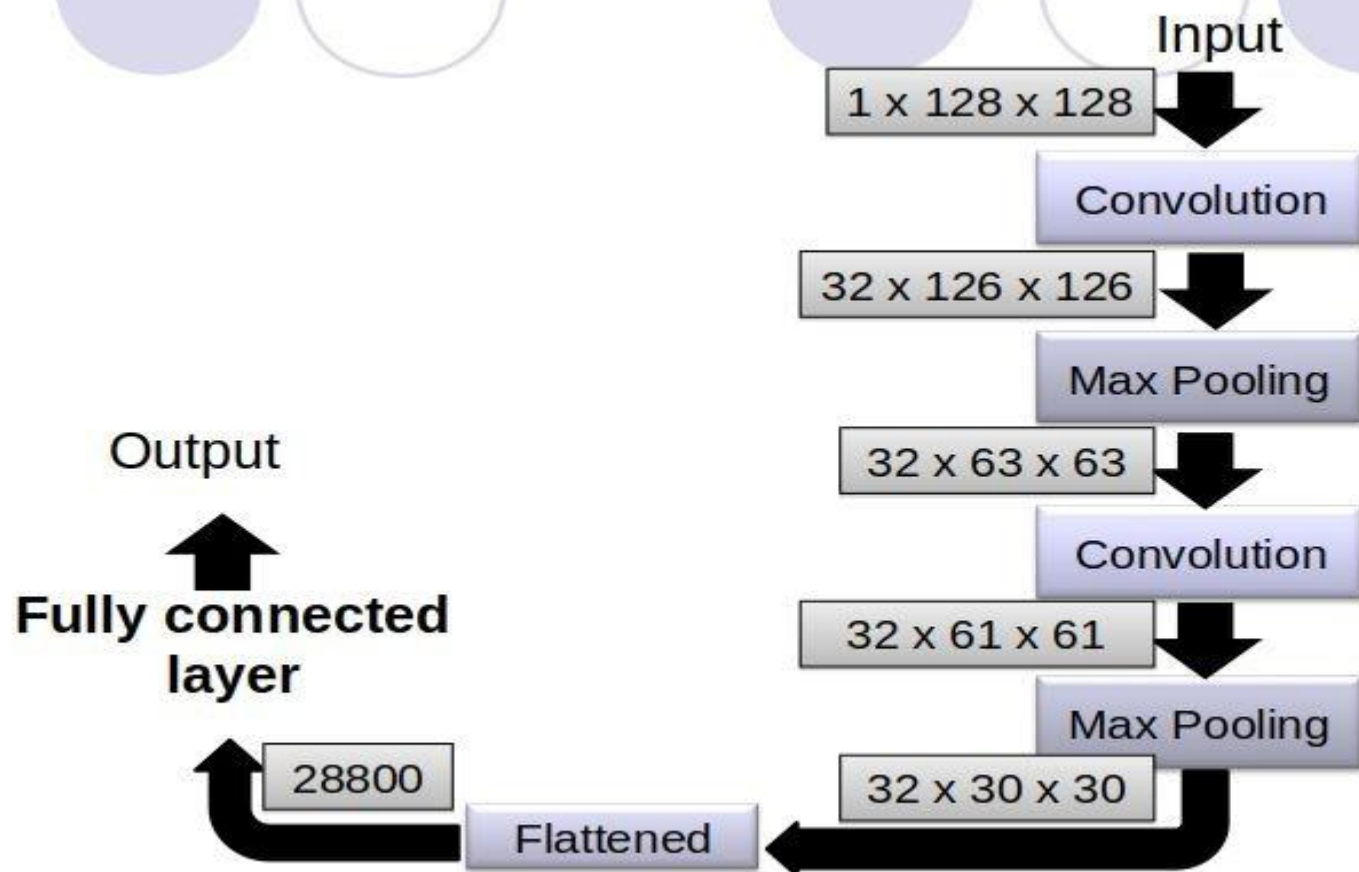
- We detect various sets of symbols which show similar results on getting detected.
- We then classify between those sets using classifiers made for those sets only.
- In our testing we found that following symbols were not showing properly and were giving other symbols also :
  1. For D : R and U
  2. For U : D and R
  3. For I : T, D, K and I
  4. For S : M and 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.



# **Our CNN Classifier Model**





# **Finger Spelling Sentence Formation**

# Autocorrect feature



A python library **Hunspell\_suggest** is used to suggest correct alternatives for each (incorrect) input word and we display a set of words matching the current word in which the user can select a word to append it to the current sentence. This helps in reducing mistakes committed in spellings and assists in predicting complex words.

# Challenges Faced

---

- We couldn't find a dataset with raw images of all the asl characters so we made our own dataset.
- Second issue was to select a 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.
- Issues were faced relating to the accuracy of the model we trained in earlier phases which we eventually improved by increasing the input image size and also by improving the dataset



## Limitations of our model

- *The model works well only in good lighting conditions.*
- *Plain background is needed for the model to detect with accuracy.*



# Conclusion

- In this report, a functional real time vision based american sign language recognition for dumb, deaf and blind people have been developed for asl alphabets.
- Prediction has been improved after implementing two layers of algorithms in which we verify and predict symbols which are more similar to each other.



# Future Scope

- ❖ We are planning to achieve higher accuracy even in case of complex backgrounds by trying out various background subtraction algorithms.
- ❖ We are also thinking of improving the preprocessing to predict gestures in low light conditions with a higher accuracy.

# Thank You !

---