Sign Language Recognition

CSE 6363 – Machine learning

-Gandhali Shastri

1001548562

## Introduction

Sign Language is the primary language used by deaf individuals to communicate. The language is as rich as spoken languages and employs signs made with the hand, along with facial gestures and bodily postures.

The project aims at building a machine learning model that will be able to classify various hand gestures used for fingerspelling in sign language. In this user independent model, classification algorithms are trained using a set of image data and testing is done on a completely different set of data. This way we can judge the accuracy of the model.

## Problem description

Fingerspelling is a vital tool in sign language, as it enables communication. Despite this, fingerspelling is not widely used as it is challenging to understand and difficult to use. Moreover, there is no universal sign language and very few people know it, which makes it an inadequate alternative for communication.

A system for sign language recognition that classifies finger spelling can solve this problem. Various machine learning algorithms are used, and their accuracies are recorded and compared in this report.

## Data description

Dataset consists of images for each alphabet and each image is clicked in different light condition with different hand orientation.

The dataset has a total of 4,972 Files in 24 Folders. Each folder represents an alphabet. There are 24 folders because the dataset doesn’t include ‘J’ and ‘Z’.

It does not include J and Z as they are finger spelled using hand movement which is not possible in an image.

## PreProcessing

### Image Segmentation

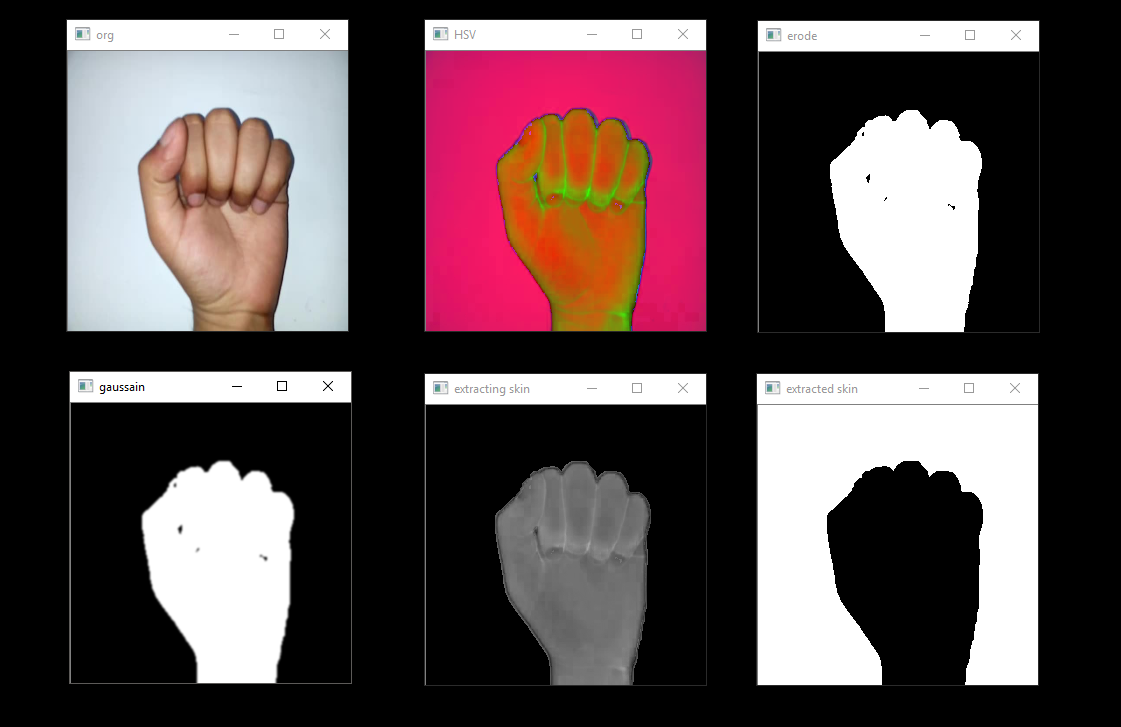
In the segmentation phase, the main objective is to remove the background and noises from the images, so that we can easily analyze the image and get only the useful information.

#### Skin masking

For getting the useful information, the technique of Skin Masking has been used. It is done by defining a threshold on RGB schema and then converting RGB to grayscale image.

#### Canny edge detection

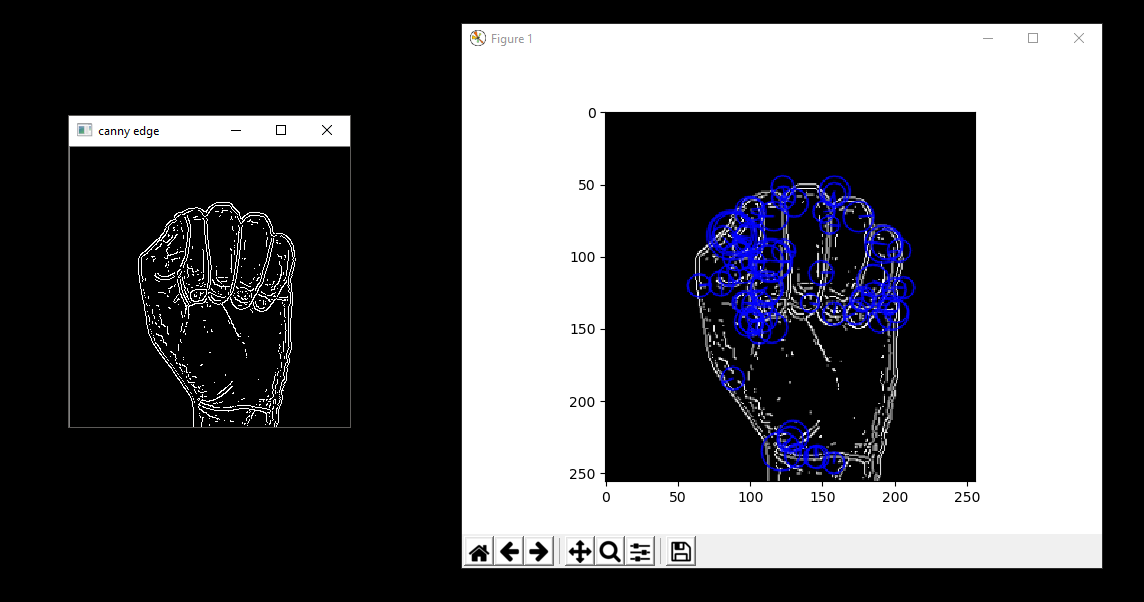
Lastly, Canny Edge detection technique is used to identify and detect the presence of sharp discontinuities in an image, thereby detecting the edges in the image. (leaving only the Region of Interest (ROI))



### Feature Extraction

#### SURF technique

The Speeded Up Robust Feature (SURF) technique is used to extract features from the region of interest that we found after image processing in the last step. SURF is similar to the SIFT technique, but it is faster and robust against rotation, scaling, occlusion and variation in viewpoint. The SURF technique returns key points and descriptors from a given image, which are used for feature matching.



## Processing

In this stage, we need to split the data into training data and test data, so that we can process the data using the classifiers. This will give us the prediction ie. the accuracy of our model.

The data is split into 60:40 ratio and then different classifiers are used on the training data.

### Bag of Features Model:

### What is a Bag of Words?

In the world of natural language processing (NLP), we often want to compare multiple documents. Documents each have a bunch of different words in a certain order. We will ignore the order and just throw the words into a bag. Then we can simply count the occurrences of each word in the corpus of all words. Finally, each document is converted to a histogram of word counts, and that can be used as features for machine learning.

Steps for Bag of Features:

1. The descriptors that we extracted in the previous step are 64-dimensional vectors, so we can simply make a matrix with every SURF descriptor in our training set.
2. Then plug that matrix to a clustering algorithm ie. in this case K Means, then get the descriptors clustered into K different clusters. (K=150 here)
3. Then go through each individual image, and assign all of its descriptors to the bin they belong in.
4. We can feed the matrix of M observations and K features to our classifiers.

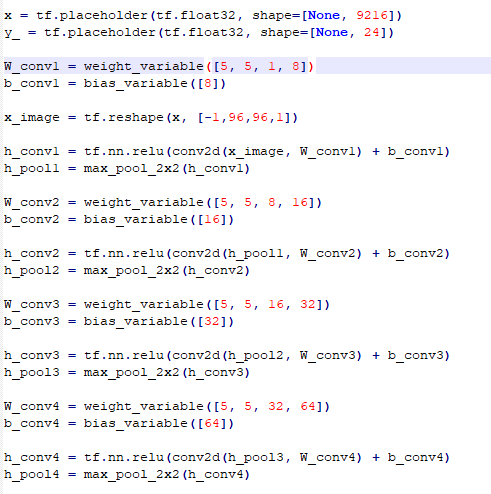
## Classification

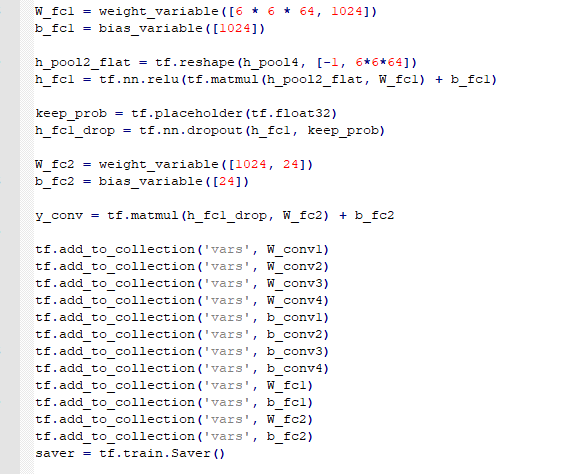
The following classifiers have been used:

* Naive Bayes
* Logistic Regression classifier
* K-Nearest Neighbors
* Support Vector Machines
* Convolution Neural Network

### Convolution Neural Network

This is the code for CNN architecture of layers.





Output:

starts

step 0, training accuracy 0

step 100, training accuracy 0.24

step 200, training accuracy 0.5

step 300, training accuracy 0.82

step 400, training accuracy 0.86

step 500, training accuracy 0.94

step 600, training accuracy 0.94

step 700, training accuracy 0.94

step 800, training accuracy 0.98

step 900, training accuracy 1

step 1000, training accuracy 1

## Results

## Analysis of approach/ Discussion

Initially when I started working on the project, I found that there were many different approaches to perform image recognition. So, initially I tried the basic approach using raw image pixels (pixel intensity values) as input to the classifier. The results I got were drastic. It was not at all stable. For example, sometimes KNN would give me 99% accuracy whereas Naïve Bayes would give just 45% accuracy.

Therefore, later I decided to implement more classifiers and compare their results. Also, instead of pixel intensity values as input to the classifier; after doing a bit research I decided to use the SURF technique for feature detection and extraction.

A multiclass SVM requires uniform dimensions of feature vector as its input. Bag of Features (BoF) is therefore implemented to represent the features in histogram of visual vocabulary rather than the features as proposed. The descriptors extracted are first quantized into 150 clusters using K-means clustering. Given a set of descriptors, where K-means clustering categorizes numbers of descriptors into K numbers of cluster center.

The clustered features then form the visual vocabulary where each feature corresponds to an individual sign language gesture. With the visual vocabulary, each image is represented by the frequency of occurrence of all clustered features. BoF represents each image as a histogram of features, in this case the histogram of 24 classes of sign languages gestures.After obtaining the BoF of features model, we can pass the features as input to the classifiers.

I observed that, using BoF and SURF gave a very stable prediction compared to earlier.

## Conclusion

In conclusion, after comparing these classifiers I can say that SVM and CNN gives the best accuracy. However, pre-processing must be performed on the dataset, which extracts the skin correctly to gain higher accuracy. Also, to get the best accuracy I have combined the approaches used in 2 papers. First approach gave me best results for skin detection and the other helped me comparing the classifiers.

## References

<http://cs231n.stanford.edu/reports/2016/pdfs/214_Report.pdf>

<https://edu.authorcafe.com/academies/6813/sign-language-recognition>

<https://ianlondon.github.io/blog/visual-bag-of-words/>

<https://www.datacamp.com/community/tutorials/cnn-tensorflow-python>