Birla Institute of Technology, Mesra

Project Thesis

CS8014

Indian Sign Language Recognition by Feature Extraction Using SURF

Author:
Aman Aryan
(BE/10227/2014)
Subham Sanghai
(BE/10257/2014)

Supervisor:
Dr. Subrajeet Mohapatra

Declaration Certificate

This is to certify that the contents of the project entitled "INDIAN SIGN LANGUAGE RECOGNITION BY FEATURE EXTRACTION USING SURF" is a bonafide work carried out by **Subham Sanghai, Aman Aryan** under my supervision and guidance in partial fulfilment of the requirements for the degree of Bachelor of Engineering in Computer Science of Birla Institute of Technology, Mesra, Ranchi.

The contents of this project report have not been submitted earlier for the award of any other degree or certificate. I hereby commend this work.

Date:

Dr. Subrajeet Mohapatra
Assistant Professor,
Dept. of CSE
Birla Institute of Technology
Mesra,Ranchi-835215

Dr. Vandana Bhattacharjee
Head of Department,
Dept. of CSE
Birla Institute of Technology
Mesra, Ranchi-835215

Certificate of Approval

The project work entitled "INDIAN SIGN LANGUAGE RECOGNITION BY FEATURE EXTRACTION USING SURF" , is carried out and presented in a manner satisfactory to warrant its acceptance as a pre-requisite to the degree for which it has been submitted. It is understood that by this approval, the undersigned do not necessarily endorse any conclusion drawn or opinion expressed therein, but approve the project report for the purpose for which it is submitted.

Internal Examiner

External Examiner

Dr. Vandana Bhattacharjee
Head of Department,
Dept. of CSE
Birla Institute of Technology
Mesra, Ranchi-835215

Acknowledgement

It is not possible to prepare a project without the assistance and encouragement of other people. This one is certainly no exception.

On the very outset of this report, We would like to extend our sincere and heartfelt obligation towards all the personages who have helped us in this endeavour. Without their active guidance, help, cooperation and encouragement, We would not have made headway in the project.

We are ineffably indebted to Dr. Subrajeet Mohapatra for conscientious guidance and encouragement to accomplish this project.We also acknowledge with a deep sense of reverence, our gratitude towards our parents and members of my family, who have always supported us morally as well as economically.At last but not the least gratitude goes to all of our friends who directly or indirectly helped us in completing this project report.

Abstract

This project aims at identifying alphabets in Indian Sign Language from the corresponding gestures. Gesture recognition and sign language recognition has been a well researched topic for American Sign Language(ASL), but few research works have been published regarding Indian Sign Language(ISL) and all of them require use of technologies like Accelorometer, Kinect Sensors, gyroscopes, etc. Here instead of using high-end technology like gloves or kinect, we aim to solve this problem using state of the art computer vision and machine learning algorithms. A data-set of around 767 segmented images was collected for 20 letters of the Indian Sign Language giving each letter around 35 images. SURF was firstly used to detect key points and describe them because the SURF features were invariant to image scale and rotation and were robust to changes in the viewpoint and illumination. SURF method is used to extract all the features of certain types of signs, then formed theirs code book from all the features by using K means clustering and finally classified using KNN, SVM and other supervised models to train the model.

1 Introduction

Sign language is a language that uses manual communication to convey meaning. An example can implicate simultaneously fusing hand geometry, movement, or orientation of the hands, arms or body, and facial expressions to convey a speaker's ideas. Sign languages often share significant similarities with their respective spoken language.

Wherever communities of deaf people exist, sign languages have developed, and are at the cores of local deaf cultures. Although signing is used primarily by the deaf and hard of hearing, it is also used by hearing individuals, such as people who can hear but cannot physically speak, or have trouble with spoken language due to some other disability.

For a native signer, sign perception influences how the mind makes sense of their visual language experience. For example, a hand shape may vary based on the other signs made before or after it, but these variations are arranged in perceptual categories during its development. The mind detects hand shape contrasts but groups similar hand shapes together in one category.Different hand shapes are stored in other categories. The mind ignores some of the similarities between different perceptual categories, at the same time preserving the visual information within each perceptual category of hand shape variation.

Indian Sign Language (ISL) is the predominant sign language in South Asia, used by at least several hundred thousand deaf signers. As with many sign languages, it is difficult to estimate numbers with any certainty, as the Census of India does not list sign languages and most studies have focused on the north and on urban areas.

In this project we have developed an intelligent sign language recognition system for the Indian Sign Language by extrapolating the methods used by various researchers on American, Chinese and Australian Sign Languages.Section 2 describes the Background.Section 3 describes the Motivation. Section 4 describes the Challenges. Some of the existing literature has been reviewed in Section 5. Section 6 describes the Problem Statement formally and is followed by Section 7 which describes the methodology. Sections 8,9 and 10 describe Results Conclusion and Future Work respectively.

2 Background

One of the earliest written records of a sign language is from the fifth century BC, in Plato's Cratylus, where Socrates says: "If we hadn't a voice or a tongue, and wanted to express things to one another, wouldn't we try to make signs by moving our hands, head, and the rest of our body, just as dumb people do at present?".

Until the 19^{th} century, most of what we know about historical sign languages is limited to the manual alphabets (finger spelling systems) that were invented to facilitate transfer of words from an oral to a sign language, rather than documentation of the sign language itself. Many sign languages have developed independently throughout the world, and no

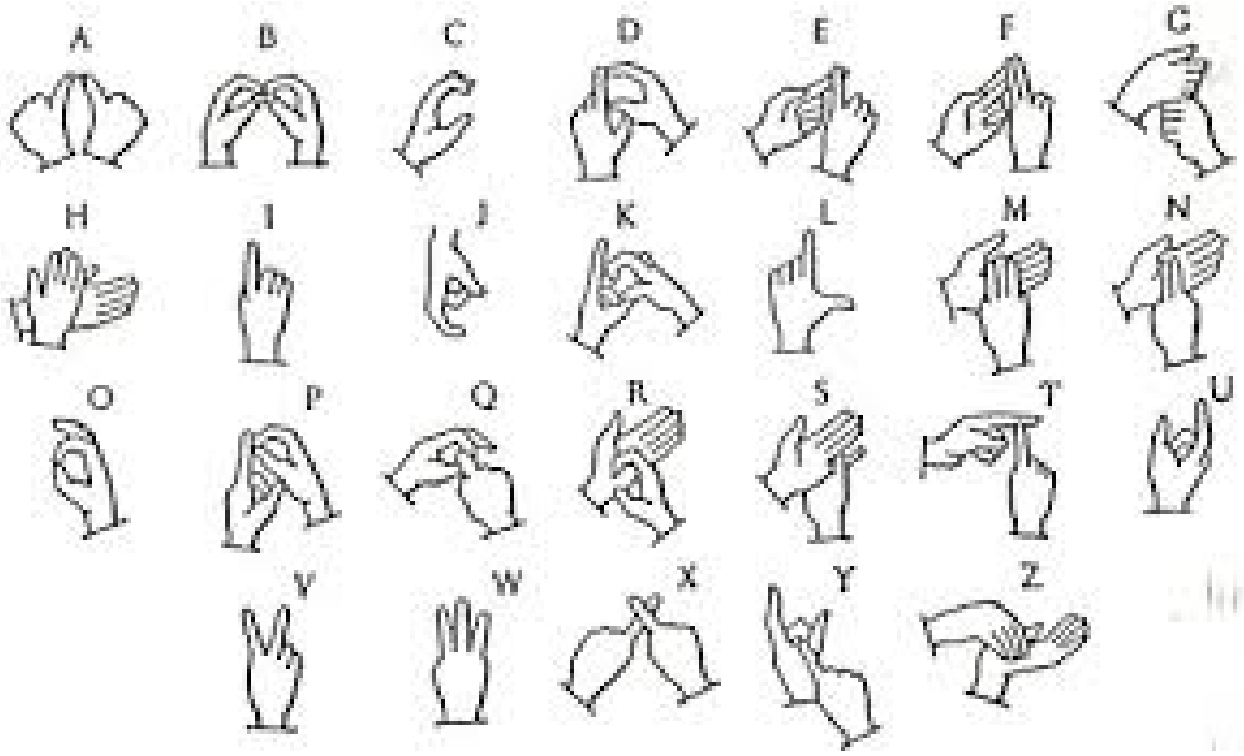

Figure 1: Indian Sign Language.

first sign language can be identified. Both signed systems and manual alphabets were found worldwide, and, though most recorded instances of sign languages seem to occur in Europe in the 17^{th} century, it is possible that popular European ideals have overshadowed much of the attention earlier signed systems may have otherwise received.

Indo-Pakistani Sign Language (IPSL) is the predominant sign language in South Asia, used by at least several hundred thousand deaf signers (2003). As with many sign languages, it is difficult to estimate numbers with any certainty, as the Census of India does not list sign languages and most studies have focused on the north and on urban areas.While the sign system in ISL appears to be largely indigenous, elements in ISL are derived from British Sign Language. For example, most ISL signers nowadays use fingerspelling based on British Sign Language fingerspelling, with only isolated groups using an indigenous devanagari-based fingerspelling system (for example, Deaf students and graduates of the school for the deaf in Vadodara/Baroda, Gujarat). In addition, more recently contact with foreign Deaf has resulted in rather extensive borrowing from International Signs and (either directly or via International Signs) from American Sign Language.

3 Motivation

Communication is one of the basic requirement for survival in society.The Indian deaf population of 1.1 million is 98 % illiterate.In line with oralist philosophy,deaf schools attempt

early intervention with hearing aids etc. ,but these are largely dysfunctional in an improvished society. As of 1986,only 2% of deaf children attended school. Deaf and dumb people communicate among themselves using sign language but normal people find it difficult to understand their language. Extensive work has been done on American sign language recognition but Indian sign language differs significantly from American sign language.ISL uses two hands for communicating(20 out of 26) whereas ASL uses single hand for com- municating. Using both hands often leads to obscurity of features due to overlapping of hands. In addition to this, lack of datasets along with variance in sign language with local- ity has resulted in restrained efforts in ISL gesture detection. Our project aims at taking the basic step in bridging the communication gap between normal people and deaf and dumb people using Indian sign language. Effective extension of this project to words and com- mon expressions may not only make the deaf and dumb people communicate faster and easier with outer world, but also provide a boost in developing autonomous systems for understanding and aiding them

4 Challenges

Extensive work has been done on American sign language recognition but Indian sign language differs significantly from American sign language.ISL uses two hands for communicating(20 out of 26) whereas ASL uses single hand for com- municating. Using both hands often leads to obscurity of features due to overlapping of hands. In addition to this, lack of datasets along with variance in sign language with local- ity has resulted in restrained efforts in ISL gesture detection. The Indian Sign Language lags behind its American Counterpart as the research in this field is hampered by the lack of standard datasets. Unlike American Sign Language, it uses both hands for making gestures which leads to occlusion of features. ISL is also subject to variance in locality and the existence of multiple signs for the same character. Also some character share the same alphabet(E.g V and 2 have the same sign, similarly W and 3 have the same sign) and the resolution of the sign is context dependent.

5 Literature Review

5.1 Sign Language Learning System with Image Sampling and CNN

This paper proposed a sign language recognition technique using 2D image sampling by constructing the training data from a sign language demonstration video at a certain sampling rate. The learning process was implemented using Convolutional Neural Network.This network consisted of three convolution layers and two full-connect layers, like the network commonly used in the MNIST problem. As a result, high accuracy was obtained using only 2D images from a low-cost camera with much less data size than previous studies.The overall technique of the paper is as given in the figures:

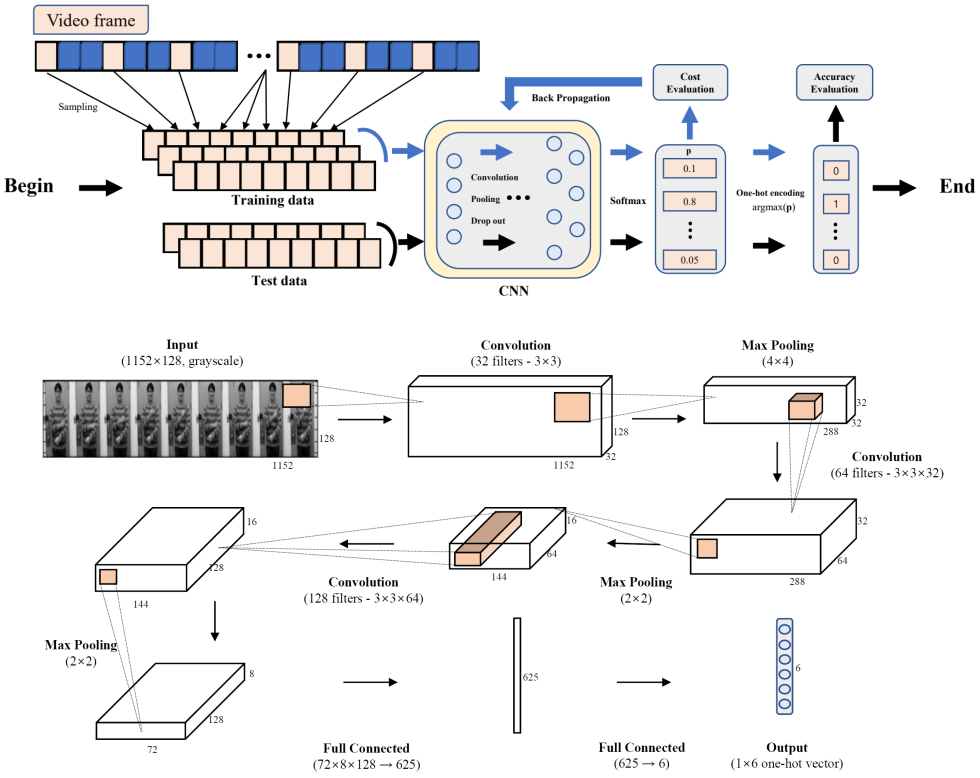

Figure 2: Convolutional Neural Network

5.2 Research and Implementation of Sign Language Recognition Method Based on Kinect

HOG and SVM algorithms with the Kinect Software Libraries are used to recognize sign language by recognizing the hand position, hand shape and hand action features. Histogram of oriented gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting).In order to realize this method a special 3D design language dataset containing 72 words is collected with Kinect and experiments are conducted.It is shown in the experimental results that the use of HOG and SVM algorithms significantly increase the recognition accuracy of the Kinect and is insensitive to background and other factors.

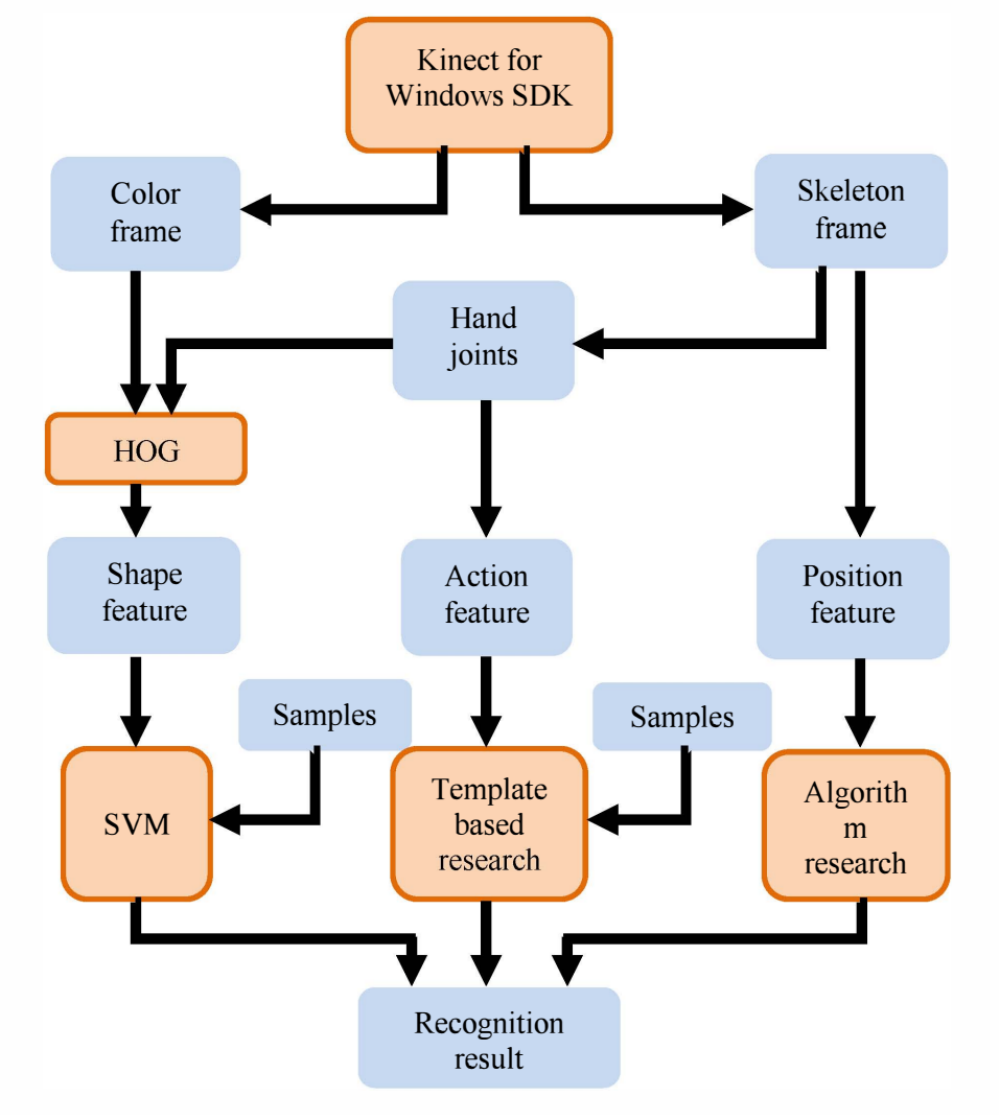

Figure 3: Overall Methodology

5.3 Talking Hands – An Indian Sign Language to Speech Translating Gloves

An approach that gives a technique for improving Sign Language Recognition system.Use of sensors which are incorporated on a glove to detect the gestures and are converted to speech with the help of a Bluetooth module and an Android Smart phone.The gloves track three kinds of movements –

- Finger bends using flex sensors: The flex sensors calculate the resistance for bend in each finger. The lesser the radius of bend, higher the resistance.
- Angular movement using gyroscope: The gyroscope calculates the angular movement in space by checking rate of change of angle along each axis. We require three measurements –

  - GyroZeroVoltage: Voltage reading when the gyroscope is still.
  - Voltage received from the gyroscope.
  - GyroSensitivity: Voltage sensitivity of the gyroscope.

- Orientation using accelerometer: The accelerometer calculates the orientation of the hands in space by determining the axes readings. Again, the axes reads are detected in voltage.

The proposed gesture recognition system converts Indian Sign Language to speech with the help of variety of sensors like flex sensor, gyroscope and accelerometer in order to successfully determine the position and orientation of the hand gesture.This system also aims at integrating the results of the sensor with a smart phone that map the sensor reading to a corresponding sign which is stored in a database.The output is the form of speech which can be easily understood by others.

Figure 4: Glove

5.4 Real-Time Recognition of Sign Language Gestures and Air-Writing using Leap Motion

Recognizes manual signs and finger spellings using Leap motion sensor.The sensor comes with the associated Application Programming Interface that provides an easy access to capture the 3D position of fingertips with a sampling rate of 120fps.Raw data captured through

the API of the device are then preprocessed and relevant features are extracted.BLSTM-NN is a sequence modeling classifier that has been popularly used in gesture and handwriting recognition problems. The classifier is able to process the input sequences in both directions, i.e. forward as well as backward with the help of two hidden layers. Both the layers are connected to a common output layer. The framework facilitates a signer to communicate using modalities in real-time, i.e. manual and finger-spelling. The recognition process has been done in two stages. Firstly, SVM classifier has been used to distinguish input gestures into two classes corresponding to manual and finger-spelling. In the second stage, two BLSTM-NN classifiers have been trained for recognition of distinguished gestures using sequence classification and sequence transcription based approaches. A dataset of 2240 gestures is prepared using the proposed framework.

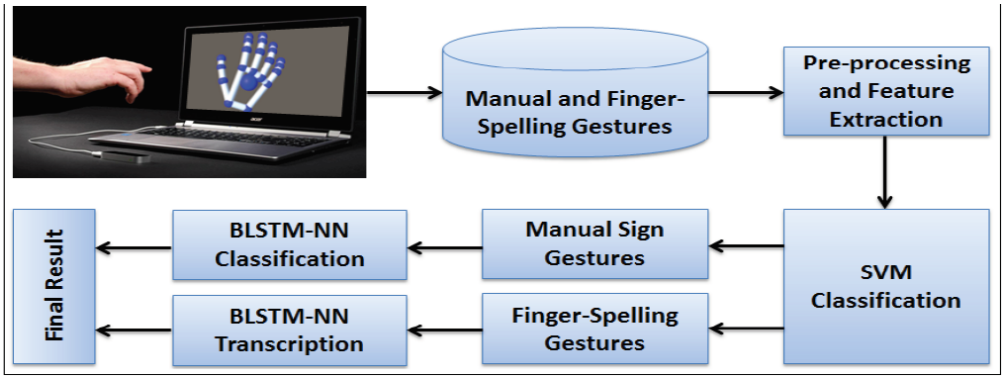

Figure 5: Overall Methodology

6 Problem Statement

It has been observed from the above literature study that Extensive work has been done on American sign language recognition but Indian sign language differs significantly from American sign language and this field has not been explored.

Again from the various studies it has been observed that there are innumerable hearing and speech impaired Indians whose primary mode of communication is sign language.For them communicating with non-signers is a daily struggle and they are often at a disadvantage when it comes to finding jobs , accessing healtcare etc.

So our main focus in this project is, implementing techniques in machine learning and image processing , where we hope to obtain a high level accuracy in distinguishing in between letters in the English alphabets of the Indian Sign Language.Our System receives input from webcam and classifies them based on the features defined by pre processing the images.

7 Methodology

7.1 Image Data Collection

A dataset of around 767 segmented images were collected for 20 different Indian Alphabets giving each letter around 35 images.

7.2 Image Preprocessing

Image processing is often viewed as arbitrarily manipulating an image to achieve an aesthetic standard or to support a preferred reality. The human visual system does not perceive the world in the same manner as digital detectors , with display devices imposing additional noise and bandwidth restrictions. Using pre-processing steps all the images were converted to a form that would allow a general algorithm to solve it and increase the accuracy of the applied algorithm.The following pre-processing steps were applied to the segmented image collected.

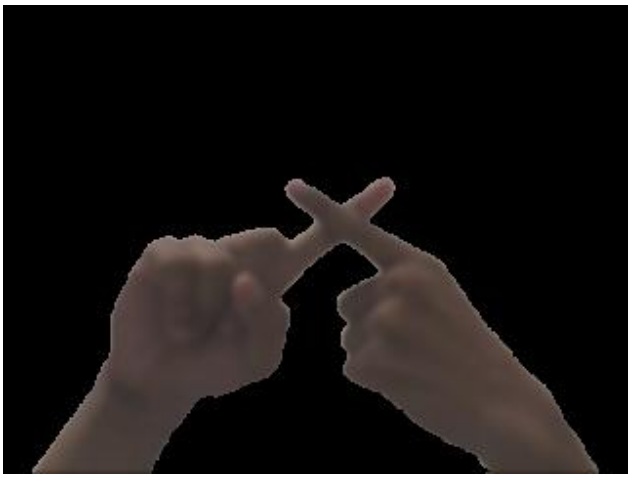

Figure 6: Original Image.

7.2.1 Resizing

The image was resized into a 100X100 image to facilitate faster computation of the preprocessing steps.

7.2.2 Skin Masking

Apart from the main object of interest, everything was made black. The RGB model consisting of Red, Blue and Green Pixel values was converted to HSV image consisting of Hue, Saturation and Intensity where Hue defines the dominant color present. The colorfulness was measured using saturation component. The intensity component was used to measure the brightness. The RGB color space doesn't separate luminance and chrominance hence R,G and B components were found to be highly correlated. So, HSV was found to be more

Figure 7: Resized image.

suitable color space for color based skin segmentation which made making the background black accurate.

To detect colors in images, the first thing we did is defined the upper and lower limits for our pixel values.Once we have defined our upper and lower limits, then make a call to the cv2.inRange method which returns a mask, specifying which pixels fall into your specified upper and lower range.

After that we detected many small false-positive skin regions in the image. To remove these small regions ,we created an elliptical structuring kernel . Then, we used this kernel to perform two iterations of erosions and dilations, respectively. These erosions and dilations helped to remove the small false-positive skin regions in the image. This smoothing step, while not critical to the skin detection process, produces a much cleaner mask.

Figure 8: Making Background Black.

7.2.3 Making the skin white

Figure 9: Making Skin White.

Image processing in grayscale is much easier than HSV because it involves simple scalar algebraic operations. So, the image was converted from HSV to grayscale and the threshold value was chosen for segregating the pixels into black and white tone. This was done to make sure that the trained algorithm does not get biased towards particular skin color.

7.2.4 Removing the arm

Arm does not play any significant role in detecting what a particular sign means. It is totally dependent on the orientation of the hands. So we cropped down 5% of the image for removing the arm so that the learning algorithm can focus only detecting patterns from hand movement.

Figure 10: Removing the arm.

7.2.5 Contour Detection

Figure 11: Contour Detection.

Contours are curves joining all the continuous points (along the boundary) having same color or intensity. It is a useful tool for shape analysis and object detection and recognition. In opencv contour image is like finding white object from black background. The largest contour was detected to find the main object we have to focus on.

The perimeter of the largest contour was found and whitened.

7.2.6 Centering and Resizing

The main object found after contouring was centered using dimensions of the largest contour obtained from the previous step. The image obtained was resized into a 30X30 image for faster training.

Figure 12: Final Image.

7.3 Machine Learning on Image

We explored the following Machine Learning algorithms on the image obtained.

- **Gaussian Naïve Bayes:** We started off with the most basic approach in Machine Learning which is naive Bayes classifiers , which are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Using Naïve Bayes classifier we got a combined of about 35%. This was because the training input was images and so we moved other classification algorithm for better results.

- **K- Nearest Neighbors:** We started simple by using K-Nearest Neighbors to train our model. We tried different values of k but could not achieve an accuracy greater than 53%. This result motivated us to go for an advanced algorithm like SVM.

- **Support Vector Machine:** Multiclass SVM using different kernels like polynomial, rbf and linear along with different values for maximum margin (C) was tried on a flattened out vector of the images. The best result was given by polynomial kernel with C=0.1 which was around 50% accuracy. This was a significant improvement on the previous result .

- **Logistic Regression:** Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome.The outcome is measured with a dichotomous variable (in which there are only two possible outcomes).From this classifier we achieved a combined accuracy of about 52%. This was because Logistic regression gives best results for binary classification problem.

- **Decision Tree:** A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences,including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.We achieved a combined accuracy of 55

- **Bagging:** Bagging refers to bootstrap aggregator which is a method of generating multiple versions of a predictor and using these to get a aggregated predictor. It helps to reduce variance from models. The three bagging models we have used are:

  - **Bagged decision tree:** Using this we achieved a combined accuracy of about 65%, which was an improvement over all the models used till now, which motivated us to use other bagging algorithms like random forest.
  - **Random Forest:**Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes. It is slightly different because it randomizes the algorithm and not the training dataset.With this we got an accuracy of about 67% which was close to the previously used bagged decision tree algorithm.

- **Boosting:** Boosting is another ensemble technique to create a collection of predictors where learners are learned sequentially with early learners fitting models to the data and then analyzing the data of errors.. The two bagging models we have used are:

  - **Ada Boosting :** da Boost is best used to boost the performance of decision trees on binary classification problems. Ada Boost can be used to boost the performance of any machine learning algorithm. It is best used with weak learners. These are models that achieve accuracy just above random chance on a classification problem. We got a combined accuracy of 45%.
  - **Gradient Boosting:**It is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.With gradient boosting we achieved a combined accuracy of 60%.

- **Voting:** The idea behind voting classifier is to combine conceptually different machine learning classifier and use a majority vote or average predicted probabilities. In this project we have combined three previously used models using a voting classifier to get a better result, which are:

  - Bagging classifier
  - Decision Tree classifier
  - Extra Tree classifier

    With this combination we got an accuracy of 75%, which was the best we achieved using various supervised learning algorithms.

7.4 Deep Learning on Image

7.4.1 Single Layer Perceptron

A single-layer perceptron network consists of one or more artificial neurons in parallel.Each neuron in the layer provides one network output, and is usually connected to all of the external (or environmental) inputs.The accuracy we achieved without adding any hidden layer was very less and it was around 54%, because the data was not linearly seperable.

7.4.2 Multi Layer Perceptron

A multilayer perceptron (MLP) is a class of feedforward artificial neural network. An MLP consists of at least three layers of nodes. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training.Its multiple layers and non-linear activation distinguish MLP from a llinear perceptron. It can distinguish data that is not linearly separable. Changing the parameter like the number of hidden layers and the nodes in the hidden layer, the best accuracy we could achieved was around 70%.

7.4.3 Convolutional Neural Networks

Convolutional Neural Networks , like other neural networks are made of neurons with learnable weights and biases. Each neuron receives several inputs , takes a weighted sum over them , pass it through an activation function and responds with an output.CNN's have a wide applications in image and video recognition , recommender systems and natural language processing. Here, the input vector is a multi-channelled image with the convolution layer is the main building block of the network which comprises of a set of independent filter initialized randomly which are learned by the network subsequently.

The architecture we used involved the following:

- Zero Padding
- Convolving
- ReLU Activation
- Max Pooling
- Flattening
- Fully Connected Neural Network
- Softmax Activation

The complete architecture can be shown as follows:

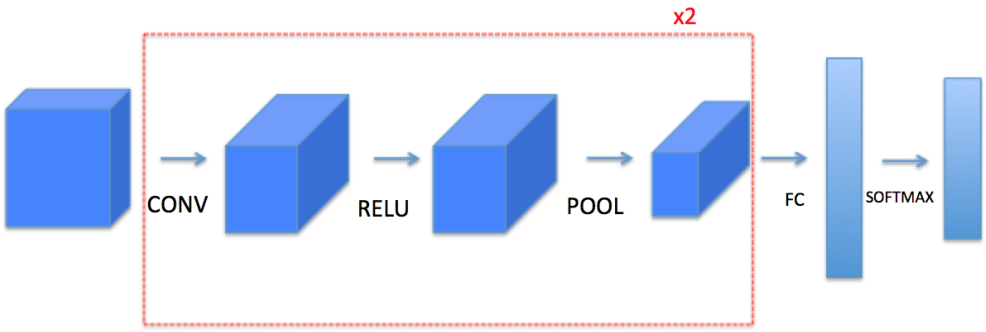

Figure 13: ConvNet Architecture.

7.4.4 Zero Padding

Image padding introduces new pixels around the edges of an image. The border provides space for annotations or acts as a boundary when using advanced filtering techniques. Zero-padding allows space for wrap-around to occur without contaminating actual output pixels.

In our case the input image of size 30X30 was padded with a single of zeros on all sides to give images of size 32X32.

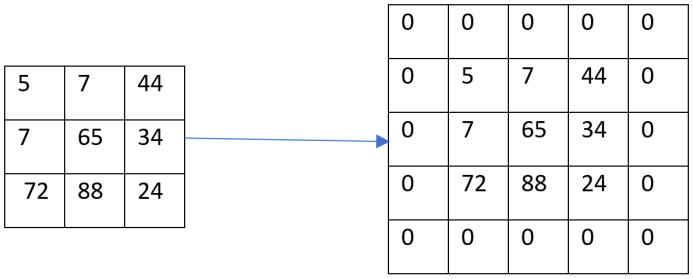

Figure 14: Padding

7.4.5 Convolving

The Conv layer is the core building block of a Convolutional Network that does most of the computational heavy lifting.The CONV layer's parameters consist of a set of learnable filters. Convolutional filtering is used to modify the spatial frequency characterstics of an image by the use of general purpose filter effect for images which are actually matrices comprised of integers. It works by determining the value of the central pixel by adding the weighted values of all its neighbors together. The output is a new modified filter image.

We used tensorflow for implementing the convolutional step. The built-in function

tf.nn.conv2d(X,W1,strides=[1,s,s1],padding = 'SAME')

where,

X=Input Image of size (613 x 30 x 30 x 1)

W1=Set of weight matrices which act as kernels each of size (4 x 4 x 1 x 8)

third input=[1,f,f,1] represents the strides for each dimension of input

7.4.6 ReLU Activation

The Convolution layer is followed by ReLU activation function which can be described as:
The graph for the same can be depicted in in the below figure as follows:
In tensorflow the function **tf.nn.relu(Z1)** computes the elementwise ReLU of Z1 .

7.4.7 Max Pooling

Max pooling is a sample-based discretization process. The objective is to down-sample an input representation (image, hidden-layer output matrix, etc.), reducing its dimensionality and allowing for assumptions to be made about features contained in the sub-regions binned. The pooling (POOL) layer reduces the height and width of the input. It helps reduce com-

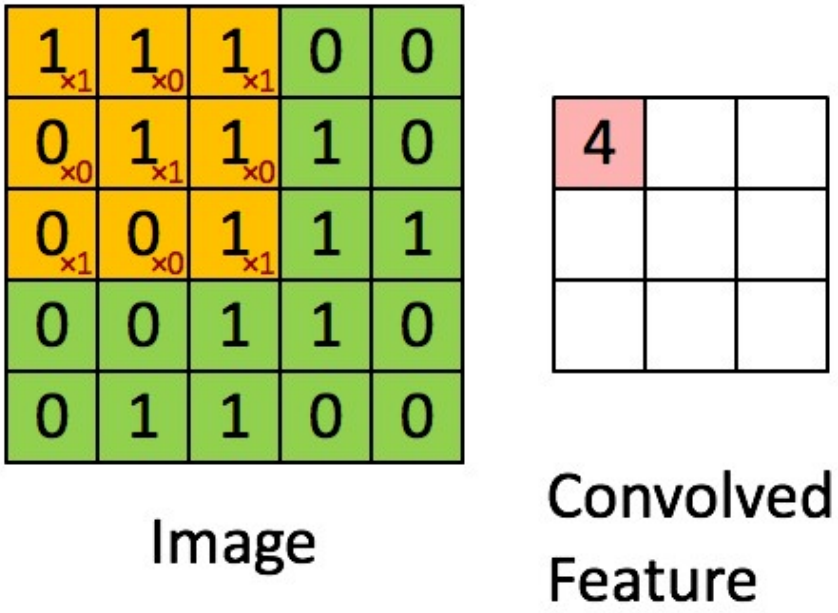

Figure 15: Convolving

$$f = \begin{cases} (x < 0) & f(x) = 0 \\ (x \geq 0) & f(x) = x \end{cases}$$

Figure 16: ReLU Equation

putation, as well as helps make feature detectors more invariant to its position in the input. It does not use padding and only uses hyperparameters instead of parameters so no back-propagation for learning is required.In Max-pooling we slide an (f,f) window over the input and stores the max value of the window in the output.

In tensorflow

(tf.nn.max_pool(A, ksize = [1,f,f,1],strides=[1,s,s1], padding = 'SAME')

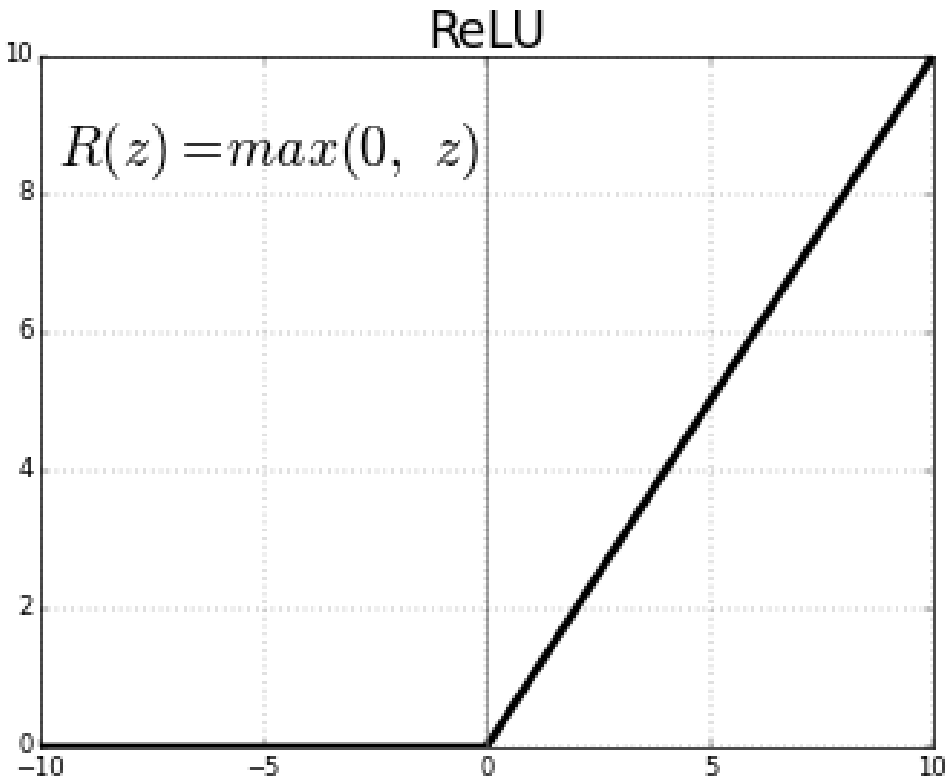

Figure 17: ReLU

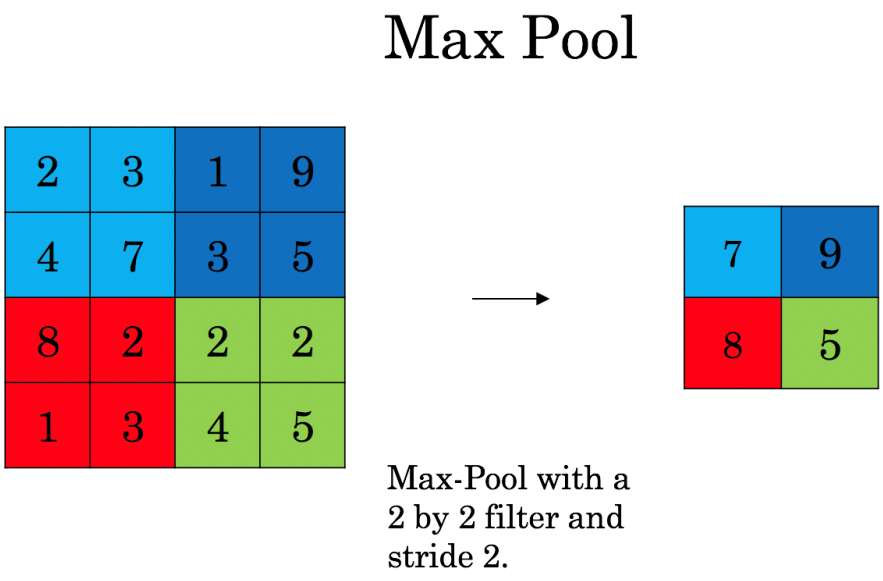

Figure 18: Max Pooling

helps us achieve the same which uses an input A and a window filter size f x f and a stride of size (s,s) to carry out max pooling over each window .

The convolution , RELU and max pooling was applied again on the consecutive inputs ,but this time the convolution was done on a new weight matrix(W2 of size (2 x 2 x 8 x 16)) whereas the window size of max pooling was of 4x4 instead of 8x8 with a stride of 4x4 .

7.4.8 Flattening

The output after applying Convolution,RELU and Max Pooling twice was flattened into a single vector to train the fully connected neural network.

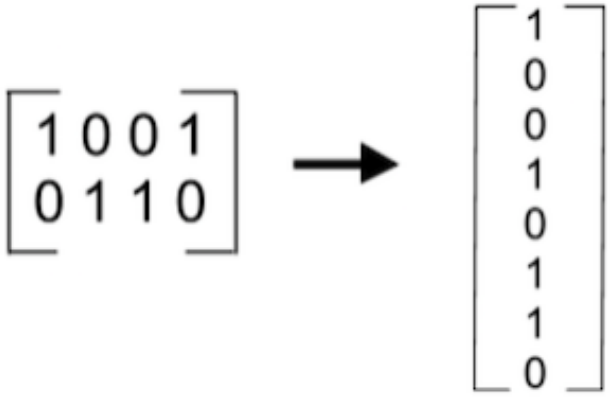

Figure 19: Flattening

7.4.9 Fully Connected Neural Network

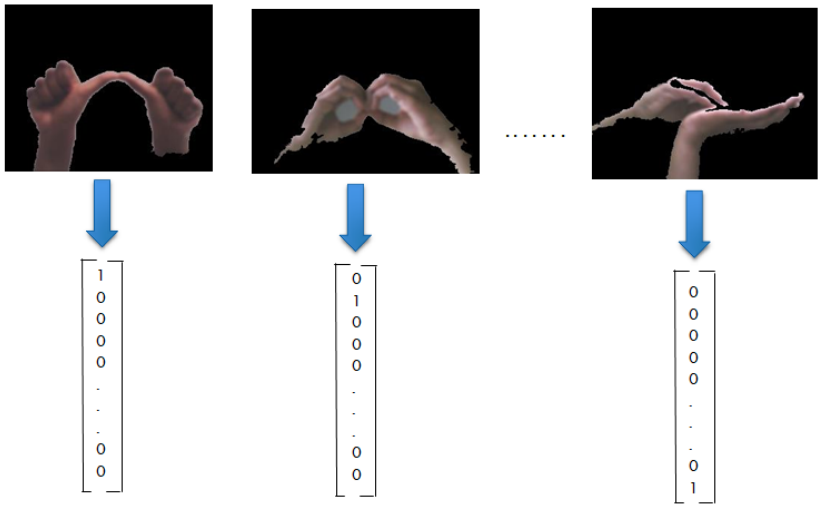

Figure 20: One-Hot Encoding

A fully connected neural network with twenty output layers and softmax activation was trained. The output layer dimension was taken as twenty because we had one-hot encoded

the letters.The fully connected neural network was trained using backpropagation and cross-entropy loss function. We tried out different optimizers and accuracies for each of them were reported.

7.5 Feature Extraction Using SURF

The Speeded Up Robust Feature (SURF) technique is used to extract descriptors from the segmented hand gesture images. SURF is a novel feature extraction method which is robust against rotation, scaling, occlusion and variation in viewpoint. For orientation assignment , SURF uses wavelet responses in horizontal and vertical direction for a neighbourhood of size 6s. Adequate Gaussian weights are also applied to it. Then they are plotted in a space. The dominant orientation is established by calculating the sum of all responses within a sliding orientation window of angle 60 degrees. Wavelet Response can be found out using integral images very easily at any scale. For many applications, rotation invariance is not required , so no need of finding this orientation , which speeds up the process.

After applying the above preprocessing steps , some more pre processing steps were used to extract feature efficiently including:

7.5.1 Canny Edge Detection

Canny Edge technique is employed to identify and detect the presence of sharp discontinuities in an image, thereby detecting the edges of the figure in focus. To decide which edges are really edges and which are not we need two threshold values minVal and maxVal. Any edges with intensity gradient more than maxVal are sure to be edges and those below minVal are sure to be non-edges, so discarded. Those which lie between the two thresholds are classified edges or non-edges based on their connectivity .If they are connected to "sure-edge" pixels , they are considered to be part of edges. Otherwise , they are also discarded.Here, for canny edge detection in case of Hand Gesture Recognition we select a minVal of & a maxVal of 60 .

After Canny edge detection we resize the frame and move on to the most important part which is Feature Extraction using SURF.

Figure 21: Canny Edge Detection

7.5.2 Feature Extraction

SURF.detect() finds the key-point in the images .Each Key point is a special structure of which has many attributes like its(x,y) coordinates ,size of the meaningful neighborhood, angle which specifies its orientation , response that specifies strength of keypoints etc. After finding the keypoints the descriptors computed from the key-points using SURF.compute() or detect and compute the descriptors from key points directly using surf.detectandCompute() as we have used .

The SURF descriptors extracted from each image are different in number with the same dimension (64). However, a supervised Machine Learning model requires uniform dimensions of feature vector as its input.So we applied Bag of Features (BoF).

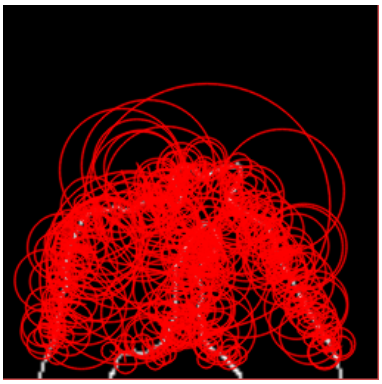

Figure 22: Key Point Extraction

7.6 Bag of Visual Words

Bag of Features (BoF) was therefore implemented to represent the features in histogram of visual vocabulary rather than the features as proposed. SURF descriptors are 128-dimensional vectors so we simply make a matrix with every SURF descriptor in our training set as its own row, and 128 columns for each of the dimensions of the SURF Features. The descriptors extracted were first quantized into k clusters using K-means clustering. Given a set of descriptors, where K-means clustering categorizes numbers of descriptors into K numbers of cluster center. Next we went through each individual image and assigned all of its SURF descriptors to the bin they belong in. All the SURF descriptors were converted from a 128-dimensional SURF vector to a bin label. Finally a histogram was made for each image by summing the number of features .

The clustered features then formed the visual vocabulary where combination of feature or visual word corresponded to an individual sign language gesture. With the visual vocabulary, each image is represented by the frequency of occurrence of all clustered features. BoF represented each image as a histogram of features, in this case the histogram of 20 classes of sign languages gestures.

K-means clustering technique categorized m numbers of descriptors into x number of cluster centre. The clustered features formed the basis for histogram i.e. Each image is

represented by frequency of occurrence of all clustered features.

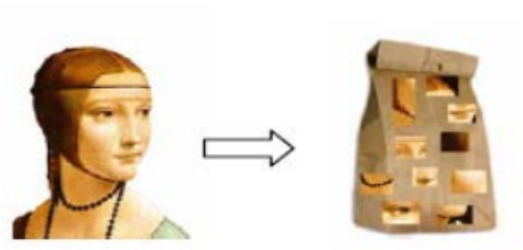

Figure 23: local Feature Extraction

Bag of words did not only map a collection of visual keywords and maintained local features of the image but also compressed effectively the description of images.

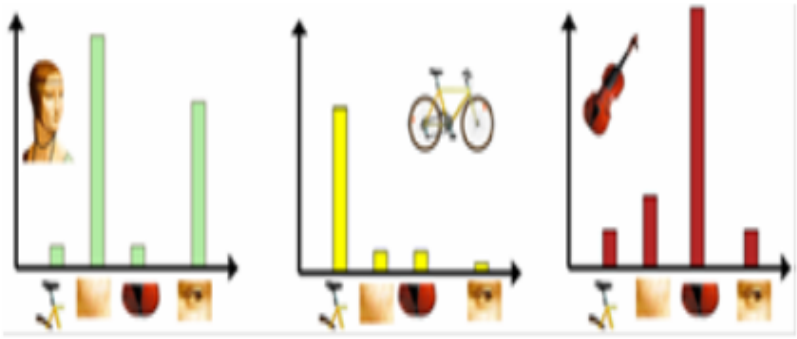

Figure 24: Conversion into a histogram of the number of occurrences of features descriptor in the given image

The complete WorkFlow can be explained in diagramatic form as follows:

7.7 Training The Model

After applying Bag of Visual Word and feature extraction , next and the final step was training the model on the image which was trained using the following models with a cross validation value of 6:

- **Gaussian Naïve Bayes:**Using Naïve Bayes classifier we got the best cross validation score [0.73584906 0.79591837 0.86666667 0.88636364 0.92682927 0.92105263] for k=150.
- **Logistic Regression:** From this classifier we achieved score of [0.98113208 0.97959184 0.97777778 0.95454545 1. 0.94736842] for k=250. This was because Logistic regression gives best results for classification problem.

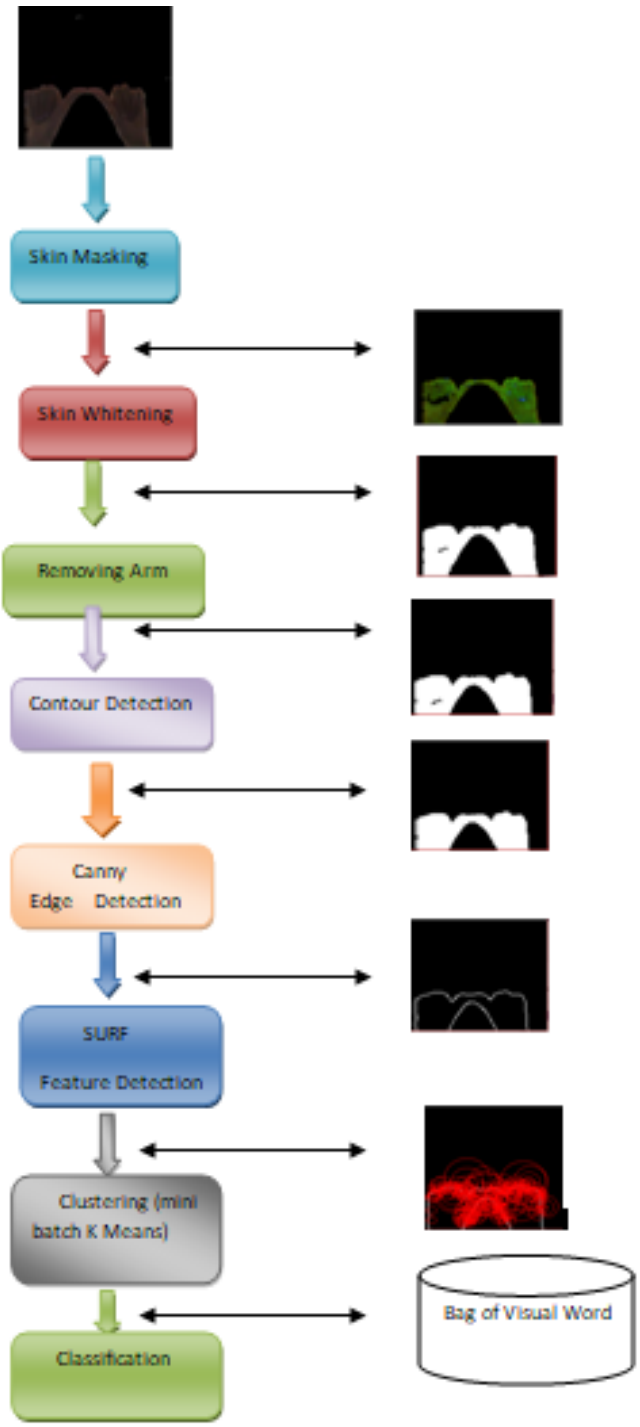

Figure 25: WorkFlow

- **K- Nearest Neighbors:** We started simple by using K-Nearest Neighbors to train our model. We tried different values of k and the best score was [0.94339623 0.97959184 0.95555556 0.93181818 0.95121951 0.94736842]. This result motivated us to go for an advanced algorithm like SVM.

- **Support Vector Machine:** Multiclass SVM using different kernels like polynomial, rbf and linear along with different values for maximum margin (C) was tried on a flattened out vector of the images. The best result was given by polynomial kernel with C=0.1 was [1. 0.95918367 1. 0.97727273 1. 0.92105263]. This was a significant improvement on the previous result .

  The results obtained were a significant improvement over all the above models implemented without applying Feature Extraction first.

8 Results

8.1 Witout Using SURF

Starting off with training our models without using Feature Extraction,the best accuracy achieved was using CNN. This section provides snippets of simulation results of some of the proposed techniques. In case of Convolution Neural Nets we had tried out various Optimizers and epochs which gave us different training and testing accuracies. The plots for the loss functions in the epochs and results for various optimizers are shown in Table 1.

SNo	Optimizer	Learning Rate	Epochs	Train Accuracy	Test Accuracy
1	Momentum	0.5	300	33.7	32.4
2	Momentum	0.5	265	94.12	77.27
3	Momentum	0.5	265	72.9	58.4
4	**Momentum**	**0.3**	**265**	**92.6**	**86.3**
5	Momentum	0.2	100	90.0	78.0
6	AdaGrad	0.5	300	91.2	75.9
7	AdaGrad	0.5	500	97.553	78.57
8	AdaGrad	0.3	500	98.2	82.57
9	AdaGrad	0.3	700	99.5	83.1
10	Gradient Descent	0.5	270	79.1	70.1

The cost vs epochs graphs obtained for the trained networks shown above are as follows:

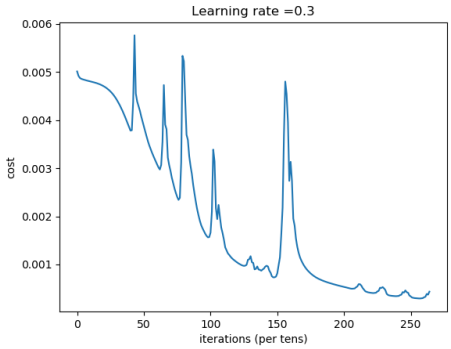

Figure 26: Momentum

The accuracy of the various models tried without Feature Extraction is been shown as follows:

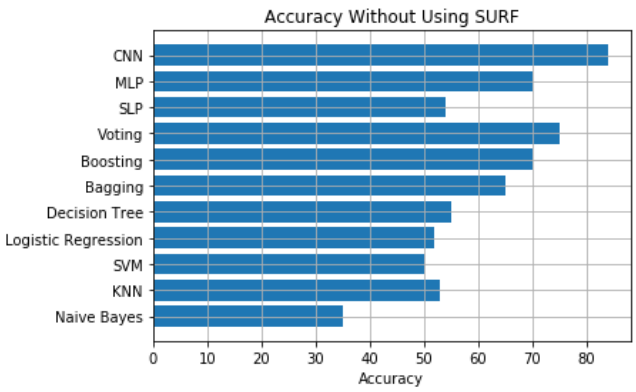

Figure 27: Accuracy Without Using SURF

8.2 Using SURF

SURF was firstly used to detect key points and describe them because the SURF features were invariant to image scale and rotation and were robust to changes in the viewpoint and illumination. SURF method used to extract all the features of certain types of signs, then formed theirs code book from all the features by using K means clustering and finally classified using KNN, SVM and other supervised models to train the model. The following figure shows the accuracy vs k value used in k means clustering in BOW algorithm

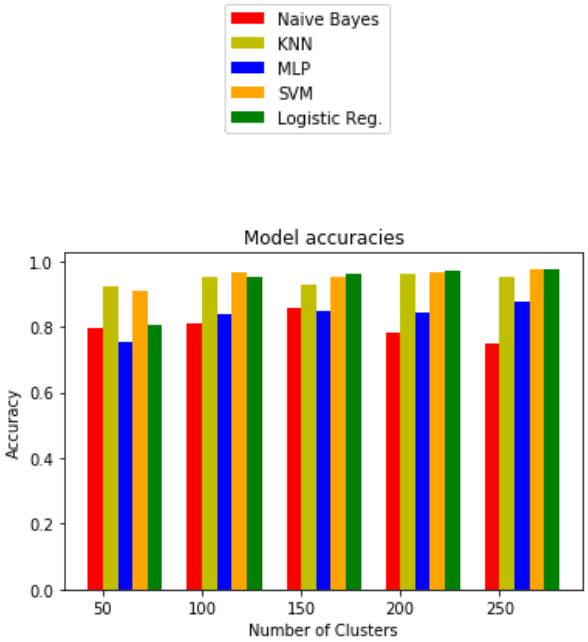

Figure 28: Accuracy vs No. of Clusters used

As we can see the best result obtained for almost all the models was when k was set to 250 as shown in the following figure.

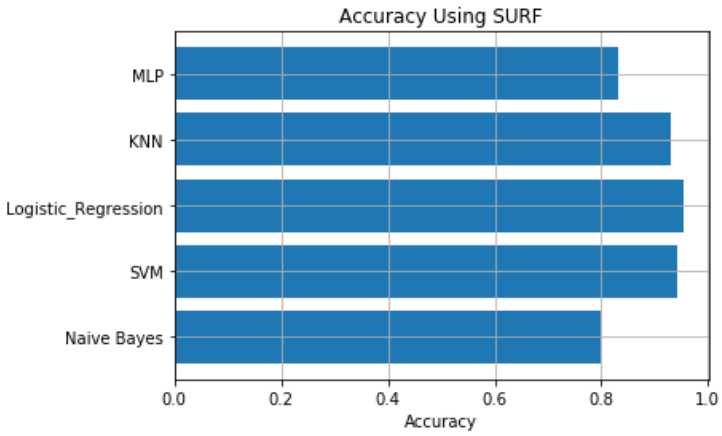

Figure 29: Accuracy Using SURF

9 Conclusion

In this project, attempts were made to achieve state of the art results for the Indian Sign Language like the ones that have been achieved for American Sign Language. The best accuracy was achieved by SVM after Feature Extraction using SURF which were invariant to to scaling,rotation .Codebooks were formed after feature extraction using SURF and K means .The accuracy reported here cannot be reported as a perfect representation of actual results because we are limited by data but can give us a direction as to which methods can be used when data is abundant.

10 Future Work

The results can further be improved by collecting more data from various schools for the specially abled. CNN architectures like VGG16, Le-Net5, etc. can be tried along with various optimizers and learning rates to achieve higher accuracies. We have right now seen methods developed only on American Sign Language and tried to extrapolate it. Various methods developed on European, German, Mexican and Ukranian Sign languages can also be applied to ISL. As future work it is also planned to add to the system a learning process for dynamic signs.

References

[1] Shihab Shahriar Hazari*, Asaduzzaman, Lamia Alam and Nasim Al Goni, “Designing a Sign Language Translation System using Kinect Motion Sensor Device”; Interna-

tional Conference on Electrical, Computer and Communication Engineering (ECCE), Bangladesh

[2] Ashish S. Nikam, Aarti G. Ambekar "Sign Language Recognition Using Image Based Hand Gesture Recognition Techniques"; 2016 Online International Conference on Green Engineering and Technologies (IC-GET).

[3] Yuqian Chen, Wenhui Zhang, "Research and Implementation of Sign Language Recognition Method Based on Kinect"; 2016 2nd IEEE International Conference on Computer and Communications: Beijing, China.

[4] Yangho Ji, Sunmok Kim, and Ki-Baek Lee, "Sign Language Learning System with Image Sampling and Convolutional Neural Network"; 2017 First IEEE International Conference on Robotic Computing

[5] Pham Quoc Thang, Nguyen Thanh Thuy, Hoang Thi Lam, "The SVM, SimpSVM and RVM on Sign Language Recognition Problem"; Seventh International Conference on Information Science and Technology, Vietnam, 2017.

[6] `https://en.wikipedia.org/wiki/Indo-Pakistani_Sign_Language`

[7] Pradeep Kumar, Rajkumar Saini, Santosh Kumar Behera, Debi Prasad Dogra and Partha Pritam Roy, "Real-Time Recognition of Sign Language Gestures and Air-Writing using Leap Motion"; 2017 Fifteenth IAPR International Conference on Machine Vision Applications, Nagoya University, Nagoya, Japan, 2017.

[8] S Yarisha Heera, Madhuri K Murthy, Sravanti VS and Sanket Salvi"; "Talking Hands - An Indian Sign Language to Speech Translation Gloves"; International Conference on Innovative Mechanisms for Industry Applications, 2017.

[9] Brunna Carolinne Rocha Silva, Geovanne Pereira Furriel, Wesley Calixto Pacheco and Junio Santos Bulhoes; "Methodology and Comparison of Devices for Recognition of Sign Language Characters"; IEEE, 2017

[10] Michael Grif, Yuliya Manueva; "Analyses of computer Russian sign language translation system with implemented semantic analyses unit"; 2017 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON),2017.

[11] Maksym Davydov and Olga Lozynska; "Information system for translation into ukrainian sign language on mobile devices"; 2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), 2017.

[12] F. Mandita, T. Anwar, Hermawan, G. Kusnanto, W. E. S. Yudha, A. Hermanto and Supangat; "An overview of searchings algorithm for Indonesian to German sign language statistical machine translation"; 2017 6th ICT International Student Project Conference (ICT-ISPC), 2017.

[13] Javier Jimenez, Anabel Martin, Victor Uc and Arturo Espinosa; "Mexican Sign Language Alphanumerical Gestures Recognition using 3D Haar-like Features" IEEE Latin America Transactions, 2017.

[14] Jian Hou, Jianxin Kang, and Naiming Qi; "On Vocabulary Size in Bag-of-Visual-Words Representation" IInternational Conference on Innovative Mechanisms for Industry Applications, 2015.

[15] https://ianlondon.github.io/blog/visual-bag-of-words/

[16] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool; SURF: Speeded Up Robust Features" Katholieke Universiteit Leuven.Seventh International Conference on Information Science and Technology, Vietnam, 2013.

[17] Ashwin S pol,Dr S L Nalbalwar and Prof. N.S. Jadhav; Sign Language Recognition Using Scale Invariant Feature Transform and SVM.International Journal of Scientific & Engineering Research , Volume 4 ,Issue 6, June-2013.