Sign Language Gesture Segmentation and Recognition

Amit Jaggi
2015013
amit15013@iiitd.ac.in

Mayank Mohindra
2015056
mayank15056@iiitd.ac.in

I. Data-set Used

A. MNIST Data-set

Mnisttraindata
Labeled training pixel data in CSV for 28x28 gray-scale (0-255) values of American Sign Language letters as hand shapes (27,455 labeled cases)
Mnsittestdata
Labeled testing pixel data in CSV for 28x28 gray-scale (0-255) values of American Sign Language letters as hand shapes (7172 labeled cases)

B. Finger-spelling Data-set

The data-set comprises 24 static signs (excluding letters j and z because they involve motion). This was captured in 5 different sessions, with similar lighting and background.It was used in the paper-Spelling It Out: RealTime ASL Finger-spelling Recognition.

C. Online Dataset for identifying Hand/NonHand

The dataset comprises of the data of 6 users and the the bound for each image containing the boundary for hand region (the region that may contain section of Hand only) in the form of a csv file.

II. Objective

As mentioned in the mid evaluation, we wanted to identify in any given image, the section of Hand/Non-Hand using multi-class classifiers rather than convolutional neural network. Then we wanted to know what the gesture meant as well. We would like to reiterate the fact that we want to achieve this for any image not limiting to the ones ised for training.

A. Image Segmentation for Mid-Evaluation

Transformation

B. image Segmentation using Hand/Non-Hand Classifier

Transformation

As we mentioned that the segmentation that we performed was using the kinect data for image depths in combination with the image. We performed naive segmentation based on depth closeness and extracted the hand from the orignal image (after suitable rescaling).

We also mentioned that we do not want to use this method since kinect is not always accessible and is not the right way to proceed with our problem.

C. Image Cropping

We realised that cropping an image for finding the hand is the first segmentation step. This was also our major goal for mid-eval.

D. Image Segmentation

We then perform Segmentation on the cropped image for the performance of gesture recognition and prediction.

E. Feature Extraction

For the extraction of the features, we use PCA and LDA to analyze on the important features and work optimally. Our main aim is to apply this training on videos and frames and so they must perform without taking too much computation time.

III. Image Cropping: First Step in Segmentation

Here we would like to explain in brief the steps we undertook for Image cropping

1) Understand and load the set of images with thei sections of Hand marked through the csv file.
2) Crop the hand according to the given bound and vectorise using feature extractors (HOG in our case).
3) Construct a dataset with images of hand/non-hand marked respectively based on the complete data and some overlap function.

4) Train the binary classifier using the same and use random sampling or sliding window approach.
5) We used random sampling but this comes with a lot of false-positives
6) To improve the classifier we use the concept of Hard negative Mining.
7) Now we are in position to test on random images, slide through the image and find the section that best contains the Hand.

IV. Skin color Segmentation

For skin colour segmentation we have used the training set of all different RGB values of Skin. We used Decision Tree to train the the classifier and then put the test image to segment the skin non-skin part.

V. Results Obtained

The Results observed were quite expected. We tried to perform this feature on our friends and the results are as follows:

As far as the results for classification are concerned:
Without Skin Segmentation:

1) PCA: 81.234
2) LDA: 56.342
3) Without Transformation: 71.342

With Skin Segmentation:

1) Without Transformation: 74.627
2) PCA: 85.0142
3) LDA: 38

Here the classifier used is the RandomForestClassifier.