American Sign Language Detection using Machine learning

Shanmukhanand Gangalla, Spring2020

***Abstract*—The project is about recognizing/detecting ASL us- ing machine learning algorithms. I would like to help half a million ASL users, mainly children with communicating with non ASL users.**

1. INTRODUCTION

The objective of the project is to detect a gesture made by hands and provide an accurate translation of the gesture. Recognize American sign language using machine learning models like CNN, GAN and other appropriate models to help deaf and mute communicate without much hardship.

1. DATA SOURCE

My dataset is from MNIST database (Modified National Institute of Standards and Technology database. The MNIST database (Modified National Institute of Standards and Tech- neology database) is a broad handwritten digit database, which is widely used to train different image processing systems. In the field of machine learning the database is also commonly used for training and research.

1. **SIMILAR PROJECTS**

I have researched on projects related to ASL detection and found two projects which are utmost similar to my capstone project

The name of the first project is American Sign language recognition using leap motion sensor.

In this project the team used a hardware device called leap motion sensor. The sensor is used to detect hand movements and hand gestures. The team used the device’s api to detect gestures and store the data. They then used K nearest neighbor and SVM to detect alphabets

The classification rate of this model is 72.78% for KNN and 79.83% for SVM.

The second project is ASL recognition and conversion to speech is a good project which detects ASL and converts it into speech. They used convolution neural network algorithm to build their model. The team used 4-layer CNN model and used background subtraction technique to recognize gestures. For predicting alphabets, they used Keras predict. For speech, they used python speech libraries. The classification rate of this model is around 80%.

So how is my project different from theirs?

Most of the ASL detection models need some kind of hardware for detecting these gestures and the accuracy rate is at best 80%. These detections take lot of time even after training a lot. My project would be hardware independent, and I am hoping my model would fetch me an accuracy of 90% or above. I would be using 6 Layer CNN and group it with GAN to train the models.

1. IMAGE DATA

Pixels are the image’s bare building blocks. A 1,000 \* 1,200 resolution means the image is 1,000 pixels wide and 1,200 pixels tall. The matrix is composed of 1,000 columns and 1,200 rows. Most pixels are colored or grayscale. Every pixel in a greyscale image is scalar between 0 and 255, with the darkest being zero. In RGB color space the color pixels are represented: red, green, and blue. The pixels are no longer scalars, but rather lists of three values: (R, G, B), all within the [0, 255] range. Of the 256 shades 0 shows no representation and 255 shows complete representation.

1. EDA ANALYSIS

My dataset is from MNIST database (Modified National Institute of Standards and Technology database. Which is com- manly used database for image training purpose. Thankfully, I had no missing data or null values. The data is distributed evenly and hence has no bias. I have around

27455 rows with 785 columns with 28x28 pixel values. I further need to use label binarizer to change my target variable from nominal.

*A. More Analysis*

Did more analysis on the data to understand how the data is distributed. Used the reference of the famous model for image detection and object detection: “handwritten digit identifier”. Some of the images of sign language were very similar to others and some were not recognizable.

1. MODEL FITTING

For the model fitting, I have used label binarizer to encode the categorical values of label parameter. I had to normalize the data values. My data values ranged from 0 to 255 so I have divided my image values by 255

I did some preprocessing of data by using keras prepro- cessing library. The preprocessing helped me achieve more accuracy and hence was a great input to the model.

1. KERAS IMAGE DATA GENERATOR

Keras preprocessing image data generator is used to con- figure image data based on the dimension wise, sample wise, rotation of the image, height and width of the image. This helped me achieve greater accuracy.

1. ACCURACY

I was able to achieve an overall accuracy of 96% with a little bit of noise. I had random fluctuations of accuracy at epochs 26-32. After 40 epochs the test data showed a constant accuracy of 1.0



Fig. 1. Distribution of Images of Sign ’D’

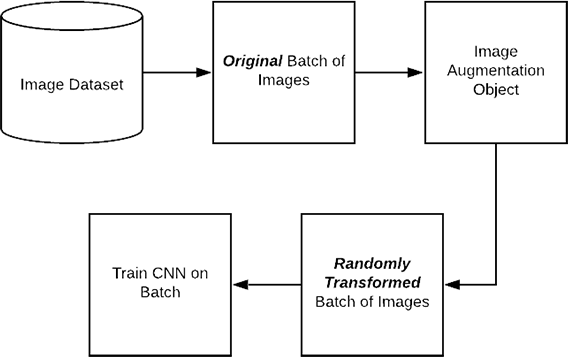


Fig. 2. Keras ImageDataGenerator and Data Augmentation

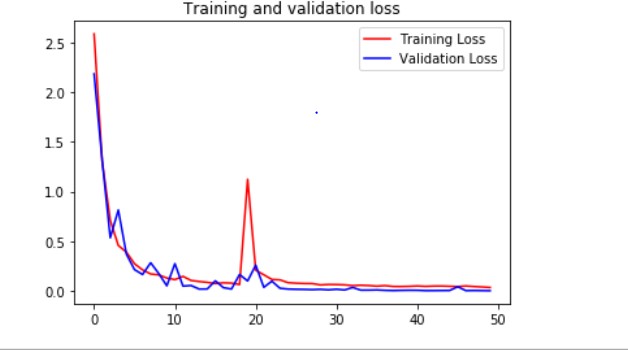


Fig. 3. Training and validation loss

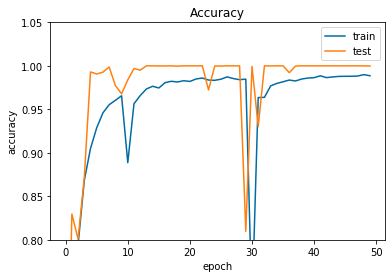


Fig. 4. Accuracy chart of test and train datasets

*A. Things to do*

I still have to find some new model to increase the accuracy and also build a real time ASL detection model. Try to build a model to detect letter ‘Z’ as it has motion involved. Also, build a model to detect sign in videos.

1. CONCLUSION

I was successfully able to build a model which recognizes American sign language using CNN with and accuracy of 99.97%.

ACKNOWLEDGMENT

1. C. Chuan, E. Regina and C. Guardino, "American Sign Language Recognition Using Leap Motion Sensor," 2014 13th International Conference on Machine Learning and Applications, Detroit, MI, 2014, pp. 541-544. doi: 10.1109/ICMLA.2014.110 URL: [http://ie](http://ie/) eexplore.ieee.org.proxy-bc.researchport.umd.edu/stamp/ stamp.jsp?tp=&arnumber=7033173&isnumber=703307 4
2. Kharpude, Simran & Hardikar, Vaishnavi & Munot, Gautam & Lonkar, Omkar & Agarwal, Vanita. (2020). ASL Recognition and Conversion to Speech. 10.1007/978-3-030-37051-0\_53.
3. Tzanakou, E.M., 2000. Classifiers: An Overview,from Supervised and Unsupervised Pattern Recognition: Fea- ture Extraction and Computational Intelligence. CRC Press LLC.,ISBN: 0-8493-2278-2, pp: 371.
4. Sebastian Raschka. 2015. Python Machine Learning. Packt Publishing. ISBN: 978-1-78355-513-0
5. Dataset: https://github.com/blackbird71SR/Neural-Netw ork-Projects/tree/master/Sign%20Language%20and%20 Static-Gesture%20Recognition/dataset