

# ASL Recognition with Machine Learning

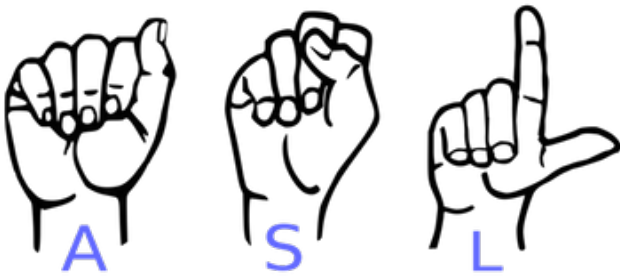
Jinwook Chang  
University of Colorado Boulder  
jich2129@colorado.edu

## ABSTRACT

In this project, I will train Neural Network model to classify images of American Sign Language (ASL) letters. After loading, examining, and preprocessing the data, I will train the network and test its performance.

## 1. INTRODUCTION

American Sign Language (ASL) is the primary language used by many deaf individuals in North America, and it is also used by hard-of-hearing and hearing individuals. The language is as rich as spoken languages and employs signs made with the hand, along with facial gestures and bodily postures. In this project I will build ASL prediction model.



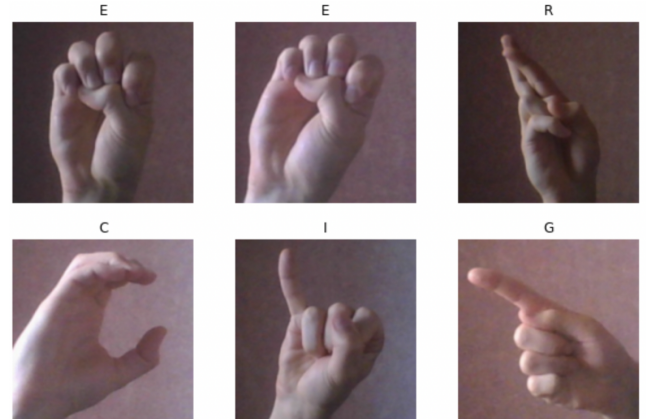
## 2. RELATED WORKS

A lot of recent progress has been made towards developing computer vision systems that translate sign language to spoken language. This technology often relies on complex neural network architectures that can detect subtle patterns in streaming video like Real-time American Sign Language Recognition with Convolutional Neural Networks[1], American Sign Language Alphabet Recognition using Deep Learning[2].

## 3. PROPOSED WORK

Using ASL images from Kaggle[3], build Neural Network model with Tensorflow framework.

Below are examples of images from the Kaggle dataset:



Each image is numbered from 0 to 25 which is matched for alphabet A to Z such as (0, A), (1, B), (2, C)

Since Label is a categorical variable, Labels should be encoded as one-hot encoded like below :

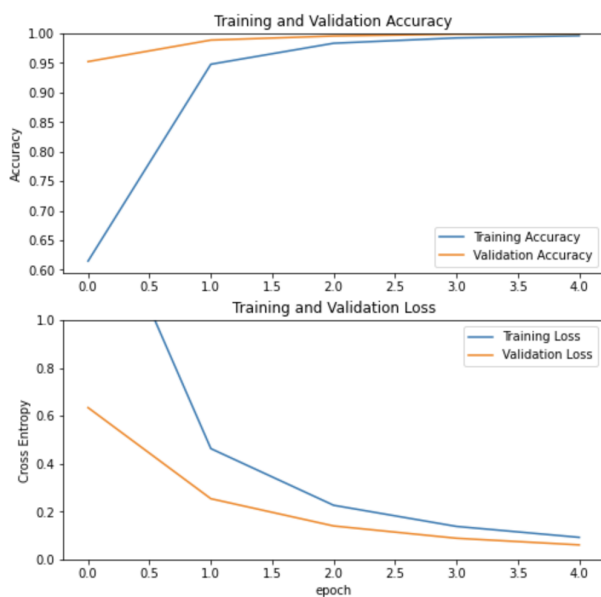
	Labels	One Hot Encoded
1	A 0	[1, 0, 0]
2	B 1	[0, 1, 0]
3	C 2	[0, 0, 1]
4	A 0	[1, 0, 0]

To Define model I will use Transfer Learning method with MobileNetV2 and add Global Average Pooling, Dropout Layer to avoid overfit. Train was being processed with 70% of datasets (16,800 sets), 15% is for validation set (3,600 sets) and I will use other 15% to test (3,600 set) the model.

Below is a structure of model. Model is compiled with Adam Optimizer, 0.0001 Learning Rate, SparseCategorical-Crossentropy Loss function and Accuracy metrics.

Model: "model"		
Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 224, 224, 3)]	0
tf.math.truediv (TFOpLambda)	(None, 224, 224, 3)	0
tf.math.subtract (TFOpLambda)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Func	(None, 7, 7, 1280)	2257984
global_average_pooling2d (Gl	(None, 1280)	0
dropout (Dropout)	(None, 1280)	0
dense (Dense)	(None, 24)	30744
Total params: 2,288,728		
Trainable params: 30,744		
Non-trainable params: 2,257,984		

Below is loss/accuracy for train and validation sets. With only 5 epochs, accuracy for each data set is 99.6%, 99.8% loss for each data set is 0.09, 0.06



## 4. EVALUTAION

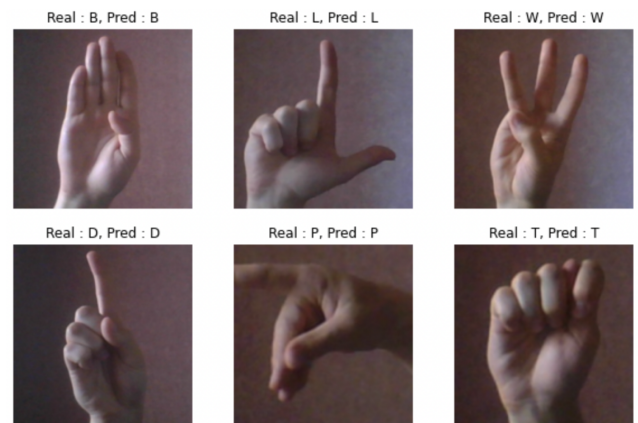
To evaluate model, I will use 15% of datasets that has never been used to be trained for model.

The Accuracy of test is about 99.99% for model.

```
loss, accuracy = model.evaluate(test_dataset)
print('Test accuracy :', accuracy)

113/113 [=====] - 8s 69ms/step - loss: 0.0613 - accuracy: 0.9997
Test accuracy : 0.9997222423553467
```

Belows are expamples of predictions



## 5. DISCUSSION & CONCLUSION

I build model which predict only one alphabet, and training epoch only 5 times. Prediction results is very impressive considering its epoch.

However this is just first step to predict ASL. Predicting one letter at a time will not be very helpful in real life. I think it is necessary to develop a model that can translate one word or one sentence in real time.

Deviating from the image classification problem model, Using Object Detection model such as YOLO or CoCoSSD.

- [1] Garcia, B. Real-time American Sign Language Recognition with Convolutional Neural Networks.
- [2] Kasukurthi, N. American Sign Language Alphabet Recognition using Deep Learning.
- [3] Ruslan, B. Sign Language (ENG Alphabet).