DATS 6203 Project Final Report
Individual Report: Teryn Zmuda

[Group 9: Toufik Bouras, Noah Olsen, Teryn Zmuda]

Topic: Classifying American Sign Language (ASL) Images

Network: Convolutional Neural Network (CNN) with Pytorch

Overview:

The problem we would like to solve is classifying American Sign Language (ASL) images. We wanted to do this project in order to gain experience applying CNN to images. The dataset is published on Kaggle platform and has 87,029 images and is over 1 GB, which is sufficient to train a deep network. We adapted VGG16 and Resnet50 models and created a custom model and compared their performances. We used Pytorch for this project, and Google Cloud Platform (GCP) and (Colab) to allow for appropriate computational power on the images.

Dataset Description

We downloaded the data from kaggle platform. This data contains a total of 87,029 images, each 64 x 64 pixels. The training data includes 87,000 images, with 3,000 files for each class. The validation set is 29 images, which allows for one classification for each image within the 29 classes. Classes include 26 letters from A-Z and three other classes: space, delete, and nothing. The data is located at https://www.kaggle.com/grassknoted/asl-alphabet.

Work Outline

The division of labor for our group was divided by technical and coding components and theory and organization of the project. Toufik and Noah worked closely on coding with Pytorch, and working and troubleshooting the execution of the model. I worked on the theory and explanation, report production, and slide compilation.

Results

With our model, we started with very low accuracy and the model learned significantly over its lifetime. The loss also improved overtime, but overall, there was very slow performance.

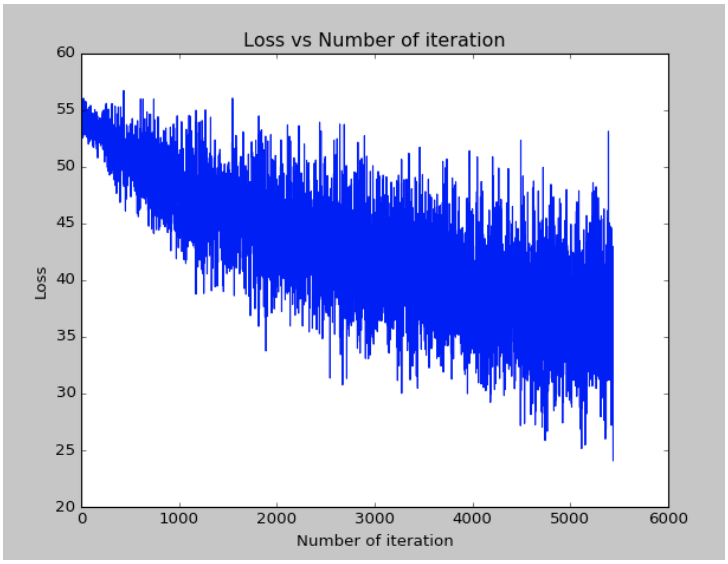

Fig. 4: Our model loss over time

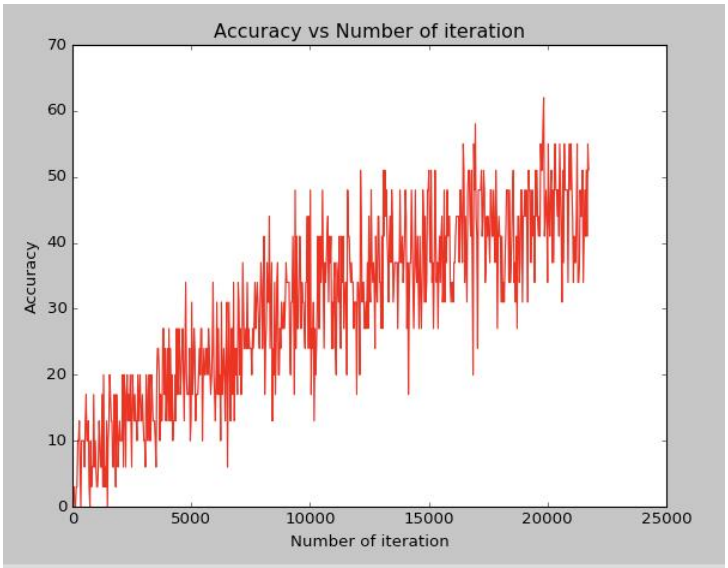

Fig. 5: Our model accuracy over time

The adapted Resnet50 model started off with very low accuracy but then scaled rapidly and had impressively quick performance. As demonstrated in Fig. 7, we ended with 90% accuracy.

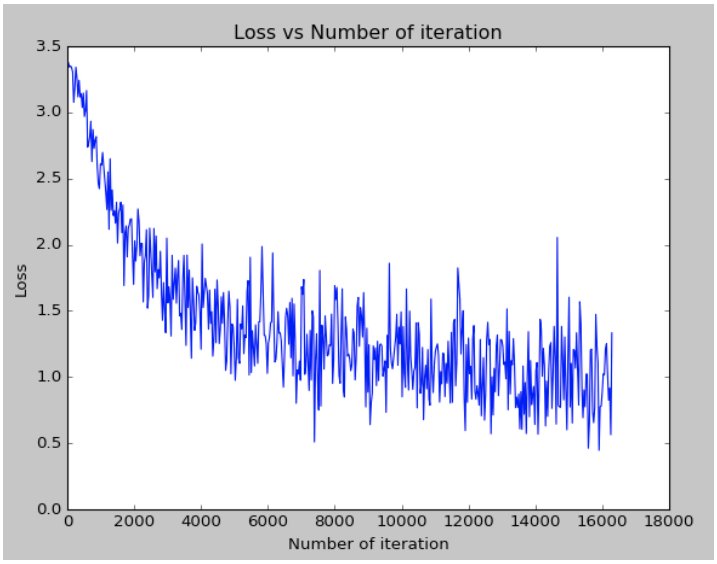

Fig. 6: Resnet50 model loss over time

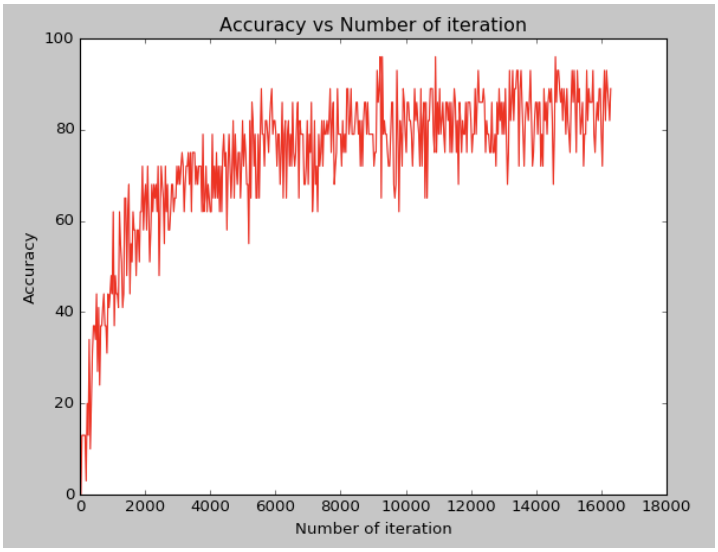

Fig. 7: Resnet50 model accuracy over time

Summary

Overall, learning about the CNN architectures was a big contribution to this report. The basis of the gradient descent model allowed us to experiment with VGG-16 and Resnet, largely keeping our model similar to the original but working towards optimization. Ultimately, we were able to get the Resnet accuracy to 90% and produce an algorithm that ran effectively to identify the ASL images.

References

https://www.kaggle.com/grassknoted/asl-alphabet

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556 . Retrieved from http://arxiv.org/abs/ 1409.1556

Stanford School of Engineering, CNN Architecture Lecture. Aug. 11, 2017. Retrieved from https://www.youtube.com/watch?v=DAOcjicFr1Y