**Project Proposal**

**Title:**

Catch These Signs

**Team Members:**

Devon Tyson, Philip Cesani, and Randall Harrison

**Introduction:**

Sign languages are languages that use the visual-manual modality to convey meaning. Sign languages are expressed through manual articulations in combination with non-manual elements. Sign languages are full-fledged natural languages with their own grammar and lexicon. Sign language is one of the most commonly used minority languages in the United States. Therefore, there is a high demand for technology that can assist the deaf community and those who use sign language with a broader means of communication.

**Problem Description:**

The goal of the project is to better understand and improve communication for the deaf community by using machine learning and computer vision techniques. We will achieve this goal by using images of signed letters and translating them into text characters that are readable by a computer. We will use a Convolutional Neural Network to take images of signed letters and classify them as their respective letter in order to make predictions about them.

**Description of Data:**

The American Sign Language letter database of hand gestures represent a multi-class problem with 24 classes of letters (excluding J and Z which require motion). The dataset format is patterned to match closely with the classic MNIST. Each training and test case represent a label (0-25) as a one-to-one map for each alphabetic letter A-Z.

**What has been done:**

So far, we have reviewed many projects, chosen the sign language dataset and analyzed the data within the set. We then determined the attributes and classifications for the project. The attributes include the pixel value for each pixel in each image. The classifications include the corresponding letter of the alphabet. We have collected and prepared the data for use and set up a Github organization with a repository for the project. We have downloaded the proper software, reviewed the Python programming language and have begun working on the project in Jupyter Notebook.

**What remains to be done:**

For the next step, we will build the Convolutional Neural Network algorithm. We will set up the training parameters, find the placeholders, biases, and weights to use to train the data. We will set up the Convolutional and Pooling layers, create the loss function and begin training. Upon training, we will manipulate the data to find the optimal parameters and display the results. Once the data is trained, we will begin to test the data and make predictions.

**Future work:**

A visual recognition algorithm could provide not only new benchmarks that challenge modern machine learning methods such as Convolutional Neural Nets but also could pragmatically help the deaf and hard-of-hearing better communicate using computer vision applications. In the future, the results from this project can be used in collaboration with a text-to-speech application with the intention of making communication as easy as possible for the deaf community. It could potentially even make communication faster and easier between deaf and blind people.

**Preliminary Plan:**

1. Find a project
2. Understand the problem
3. Download the proper software
4. Review and learn the Python language
5. Set up a Github Organization
6. Define the project objective
7. Prepare the data
8. Collect the data
9. Select Algorithm
10. Train the model
11. Test the model
12. Make predictions
13. Conclude the results

**References:**

1.Huy V. Vo, Francis Bach, Minsu Cho, Kai Han, Yann LeCun, Patrick Perez, Jean Ponce.  Unsupervised Image Matching and Object Discovery as Optimization. In CVPR, 2019.

2. P. Isola, J-Y. Zhou, and A.A. Efros. Image-to-image translation with conditional adversarial networks. In CVPR, pages 1125 - 1134, 2017.

3. J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image to image translation using cycle-consistent adversarial networks. In CVPR, pages 2223–2232, 2017.

4. M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-toimage translation networks. In Advances in Neural Information Processing Systems, pages 700–708, 2017

5. Raia Hadsell, Sumit Chopra, Yann LeCun.  Dimensionality Reduction by Learning an Invariant Mapping.  In CVPR, 2006.