Sign Language Image Classification

Ishita Jain, Jyothi Ganji, Mihir Gadgil

Problem:

Classify hand images by the sign language symbol they represent. If we can recognise these images with high accuracy, it can help in many applications. One example is the transcription of a lecture, where a translator is using sign language to convey the lecture to hearing impared audience. Since there are only a small and finitely many sign symbols possible, we believe this might be an easier problem to solve than audio transcription.

Dataset:

We are using a dataset from Kaggle, named Sign Language MNIST. This is a proposed drop-in replacement for the original handwritten digits MNIST dataset, just like the Fashion MNIST dataset. This is American Sign Language.
https://www.kaggle.com/datamunge/sign-language-mnist
The dataset has about 27,000 images and hence, is large enough to train a deep neural network.

Network Architecture:

As of now, the plan is to use Convolutional Neural Network, since that is the only image-data specific network we have learned about. If we come across any other architecture that might be more suited to the task, we will consider using it too.

Framework:

We aren't sure about this right now. Keras, Tensorflow and Torch are the only frameworks we are somewhat familiar with. We don't know enough about them to prefer one over the others.

Reference material:

We found some research papers that have attempted this problem.
https://pdfs.semanticscholar.org/397f/2b166136ff2f5606d4295da4fc3a797f1c4e.pdf
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8622141
Along with general material on neural networks and sign language, we will be referring to these papers.

Metrics:

The dataset doesn't have exactly equal number of examples for each class, but they are close. Also, accuracy is important for the kind of applications this system might be used in. So we are choosing accuracy as the metric of performance.

Schedule:

Explore different frameworks and decide which one to use for our project	By Nov 3
Data Cleaning and Preprocessing	By Nov 10
Model building, hyperparameter tuning, performance measurement	By Nov 20
Final report and presentation preparation	By Nov 30