CS 230: Deep Learning

ASL Optical Flow

Diego Celis
dcelis [at] stanford [dot] edu

Ella Hofmann-Coyle
ellahofm [at] stanford [dot] edu

Introduction

Today there are countless language translators for nearly all spoken languages. However, no such translator exists for American Sign Language (ASL), leaving those who rely on ASL at a disadvantage. We hope to bridge the language barrier between those who speak ASL and those who only speak English by creating a translator. Our system will take in video input of a person signing and classify the sign.

Data

We got our data from Purdue University's RVL-SLLL American Sign Language Database. The database is comprised of video snippets of collections and combinations of various subjects signing different letters, words, and phrases in ASL. Data is mostly labeled and is presented colorized.

Features

In one approach we apply layered optical flow images: 3 dimensional input with angle and magnitude encoded into two channels. In another we applied a CNN to several raw RGB frames per sign from the segmented data set. We saw this as appropriate as our data was good at localizing the movement of the signage, meaning that a CNN and optical flow would both derive good results from such a setup.

Model

$$Z_1 = W_1^T X + b_1$$
$$A_1 = relu(Z_1)$$
$$Z_2 = W_2^T A_1 + b_2$$
$$A_2 = relu(Z_2)$$
$$Z_3 = W_3^T A_2 + b_3$$

In our first attempt at this problem we fed in raw RGB images into a 3-layer CNN using 2D convolutions. For this approach, we attempted to minimize the cost via softmax cross-entropy.

In another attempt we stacked 5 optical flow images generated from 5 frames of a signing clip as input to a 3D NN architecture. In this approach we attempted to minimize categorical cross-entropy.

Discussion

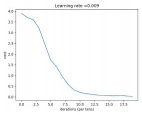
Our project took a very different form than what we had initially had hoped. When scoping this problem, we planned on using OpenPose, a software from CMU, that vectorizes hand positions and then feeding this into a network. Unfortunately we were not able to install the software, and thus took on the challenge of action recognition, a much more challenging problem, but a great learning opportunity.

We based our inspiration for looking at optical flow images from the two-stream CNN model. We attempted the optical flow CNN. The poor results from running a 3D CNN were likely due to difficulties from optical flow in deciphering hand movement. This is caused by finger positioning (covering other fingers) and granularity issues.

Future

While our model predicts still hand signs well, it does a poor job on dynamic signs. Our results would greatly be improved, by using either a 2-stream CNN with a temporal and spatial stream, or the OpenPose vectorization.

Results



Despite not using OpenPose, we received favorable results with the CNN RGB. For the 3D NN, our efforts were largely hindered by lack of computational power alongside architectural challenges.

Model	TrainAcc	Samples	TestAcc	Samples
CNN RGB	.991	455	.872	195
3D NN	.181	11	.143	7

References

SIMONYAN, K. AND ZISSERMAN, A.
Two-Stream Convolutional Networks for Action Recognition in Videos
In-text: (Simonyan and Zisserman, 2014)
Your Bibliography: Simonyan, K. and Zisserman, A. (2014). *Two-Stream Convolutional Networks for Action Recognition in Videos*. [online] Arxiv.org. Available at: https://arxiv.org/abs/1406.2199

FEICHTENHOFER, C., PINZ, A. AND ZISSERMAN, A.
Convolutional Two-Stream Network Fusion for Video Action Recognition
In-text: (Feichtenhofer, Pinz and Zisserman, 2016)
Your Bibliography: Feichtenhofer, C., Pinz, A. and Zisserman, A. (2016). *Convolutional Two-Stream Network Fusion for Video Action Recognition*. [online] Arxiv.org. Available at: https://arxiv.org/abs/1604.06573 [Accessed 9 Jun. 2018].