End-to-end Translation System for American Sign Language using Deep Learning

Christian Aguilar, Stephany Lopez, Allyssa Villanueva, Michael Cancino
Advisors: Dr. Dong-chul Kim, Dr. Weidong Kuang, Mr. Carlos Rodriguez-Betancourth

Department of Electrical and Computer Engineering
College of Engineering and Computer Science

Abstract

Language is the principal method of human communication. According to the National Center for Health Statistics, over 2 million Americans are classified as deaf, yet only 250,000-500,000 people use American Sign Language (ASL)[1]. While the use of smartphones and similar devices have aided the communication barrier between deaf and hearing individuals, we've observed that there is a lack of accommodations presented by establishments across the nation for individuals who are hearing-impaired. With the adoption of innovative technologies such as machine learning and computer vision, we present an end-to-end translation system to provide our developing solution for this barrier by utilizing deep learning to translate ASL. In this paper, we discuss the first iteration of this project and our forthcoming plans to expand this system.

Introduction

The goal of this project is to create a system that aids communication for conversations between hearing and deaf individuals. We have observed that between the lack of appropriate accommodations in several day-to-day establishments and obstacles surfaced by the COVID-19 pandemic for the deaf community, there is a large need for innovative solutions to help bridge this barrier of communication.

The first part of this product is to create a system that translates ASL to written English to communicate with hearing individuals, and the second part is to support a speech-to-text feature that allows hearing individuals to communicate with deaf and hard-of-hearing individuals, thus, creating an end-to-end system.

Product Model

Our project consists of a general 5-step process that can be seen in the figure below.

Figure 1. Product Flowchart

1. Our model takes an image of a sign
2. Image is fed into the deep learning model
3. The model returns the name of the sign (aka label)
4. The label is used in a separate program to send a series of digital signals to 4 different output pins
5. The digital signals are sent through a combinational logic circuit utilizing a 7-segment display allowing us to display the translation

Product Architecture

Given the complexity of our software, our project required a device with enough computing power to train and execute machine learning projects, as well as provide a method of controlling external peripherals. After surveying several options, we chose to use the NVIDIA Jetson Nano, as it was designed to develop AI/ML applications and provides access to 32 General Purpose IO pins to send/receive digital signals.

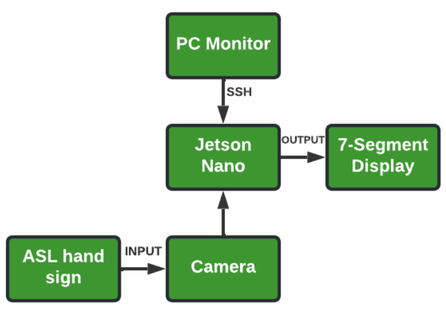

Figure 2. Product Hardware Architecture

For outputting the translation, we used 7-segment display for the first iteration of this project and will be moving to an LCD this semester for better versatility in translation display.

Methodology

Data Collection and Processing
We collected roughly 6,000 static images of 'A' and 'B' and pre-processed them by resizing, renaming, and reformatting all images to ensure they were consistent throughout the training process.

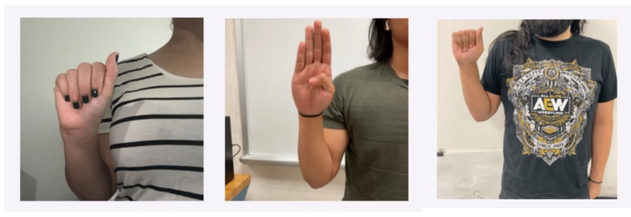

Figure 3. Data Collection Examples

Image Classification
After we collected and processed all our data, we moved into the first stage of the project where we used image classification and concluded this training phase and obtained a high accuracy of 0.96% for 100 epochs through our dataset. Upon testing the model with new data however, we were unable to gain a high prediction confidence despite the high accuracy we achieved as seen in the figure below where our model classified both signs as letter 'B'.

Figure 4. Incorrect prediction 'B' when sign is 'A'

Object Detection
In this second phase, we use object detection, a computer vision technique that locates and classifies an *object* in an image using a rectangular bounding box as opposed to classifying based on the entire image itself[2]. To implement this object detection, we needed to reprocess our data and manually annotate our dataset with bounding boxes as seen below.

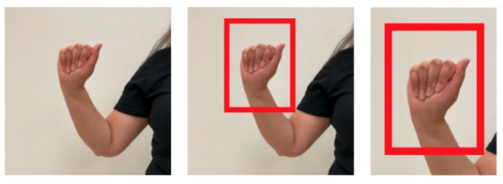

Figure 5. Example of bounding box on sample data

After training our data with the *Faster R-CNN ResNet-50 FPN model*[4] we achieved a total loss of 0.9 and a 98.5% confidence when predicting objects. While this model performed well, we encountered two major drawbacks: configuration errors when migrating the model to the Nano and a limited potential for real-time translation. We retrained a new model to address these issues- the *DetectNet*[3] model and were able to achieve a total loss of 0.07 with a ~100% confidence when translating letters and successfully sending the output to our hardware components in **real time**.

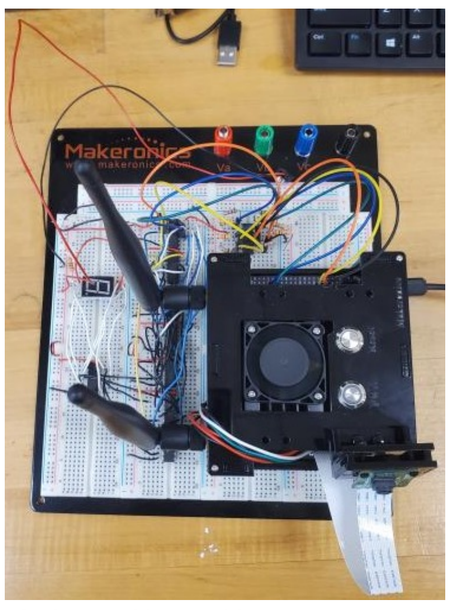

Figure 6. Jetson Nano with seven-segment display

Results & Discussions

Throughout the entire development of this project, we faced several obstacles that affected the timeline of our project and ultimately, we moved with urgency to adapt quickly and redesign new solutions.

We were able to successfully train our model and concluded with a total loss of .07 and 99% accuracy when translating A and B with **live** translation at 40-45 predictions occurring per second.

Additionally, we were able to create a combinational logic circuit for our seven-segment display and utilized an op-amp to change the power source so that the circuit would use the same source as the inputs.

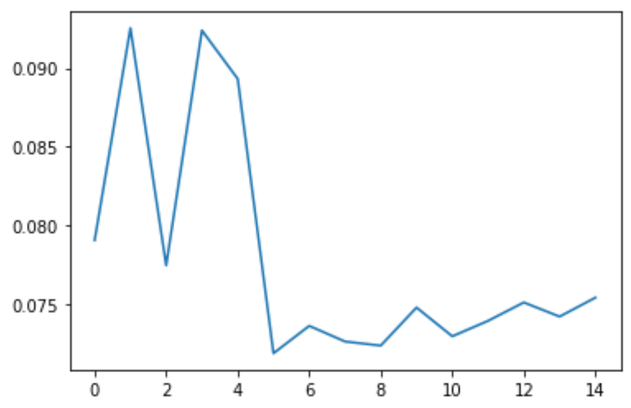

Figure 7. Object detection model loss summary

Conclusions & Future Work

While there recently has been several advancements in machine learning/computer vision, we faced several obstacles throughout the development of this project, requiring us to redesign our product due to hardware and environment constraints. We were able to implement various methods of computer vision in our project to reach our overall goal of aiding everyday communication between hearing and deaf individuals and believe that these breakthroughs will accelerate the trajectory of our work for the next semester.

We plan to train our model with the rest of the alphabet and essential complete phrases. We have collected and annotated our dataset with over 36,000 photos, thus far, and are analyzing different optimization methods to determine which provides a more efficient training process and a better experience for the users of this product.

Additionally, we will be switching our methods of display due to the restrictions presented by the 7-segment display, thus requiring further software and hardware development. With this new screen however, we will be able to display several translations for both letters and phrases.

Acknowledgments

The authors gratefully acknowledge the contributions of this project from our former team member, Catalina Diaz, who graduated December 2021, the support from Ana Lilia Hernández, president of the UTRGV ASL Club, and finally, the several individuals who were a part of our data collection process.

References

1. Mitchell, R., Young, T., Bachleda, B., & Karchmer, A. (2006). How Many People Use ASL in the United States?. *Gallaudet Research Institute*.
2. Zhong-Q, Z., Peng, Z. (2018). Object Detection with Deep Learning. *IEEE.*
3. Tao, A., Barker, J., Sarathy, S. (2016). DetectNet: Deep Neural Network for Object Detection in DIGITS. *NVIDIA Developer.*
4. Ren, S., He, K., Girshick, R., Sun, J. (2016) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *DBLP.*