A Wireframe-Based Approach for Classifying and Acquiring Proficiency in the American Sign Language (Student Abstract)

Dylan Pallickara[1], Sarath Sreedharan[2]
[1]Poudre High School International Baccalaureate Program, Fort Collins, CO, USA.
[2]Computer Science Department, Colorado State University

Abstract

We describe our methodology for classifying American Sign Language (ASL) gestures. Rather than operate directly on raw images of hand gestures, we extract coordinates and render wireframes from individual images to construct a curated training dataset. We also explore distilling wireframe representations as joint angles. Because we construct wireframes that contain information about several angles in the joints that comprise hands, our methodology is amenable to training those interested in learning ASL by identifying targeted errors in their hand gestures.

Introduction

Sign language has, for centuries, been the primary mechanism for the deaf and hard-of-hearing communities to communicate. However, bridging the gap between verbal and non-verbal communications has been a challenge (Pangestu et. al. 2022). Improving sign language translation models can have a transformative impact on communication and accessibility for the non-verbal. Crucially, these advancements will allow individuals who are reliant on sign language to express themselves and access essential services without communication barriers. Lowering such barriers can improve communications in myriad settings including, but not limited to, healthcare, legal proceedings, customer service interactions, and emergency situations (Ravikiran et. al. 2009). In this paper, we introduce a novel technique for ASL recognition that leverages a wireframe-based representation. We also use our approach as a foundation to develop an ASL tutoring system, that aims to provide feedback on signs generated by its users.

Methodology

Several efforts have targeted ASL recognition from raw images directly. However, depending on the data used to train the models, such methods may be subject to biases that stem from variations in the skin tone, texture, and finger thickness that may not have been sufficiently captured in the datasets. We get take a two-phased approach (Sherif 2022). In the first phase we extract wireframes of hands while the second phase classifies the ASL sign.

Our primary dataset comprising images of ASL hand gestures is from Kaggle[1]. We transformed these images to extract wireframes associated with each gesture. The data are first passed through TensorFlow's hand landmarking functionality that resulted in a set of 20 coordinates per image. We generate a wireframe image (rendered over a black background) from these coordinates as depicted in Figure 1. The generated wireframe image effectively attenuates background interference and noise by reducing variability across participants in the dataset. Distilling each image into a simple wireframe also removed differences in skin tone, finger thickness, and skin texture.

The wireframe extraction process was performed for 3,000 images per ASL sign to construct a curated dataset of wireframe images. This curated ASL dataset contained numbers 0 through 9 as well as every alphabet with the exception of J and Z that are not amenable for classifications using still images. The curated ASL wireframe dataset was then used to train our deep network. Our deep network was built using Tensor Flow and Keras as a sequential 8- layer model that was trained over 10 epochs.

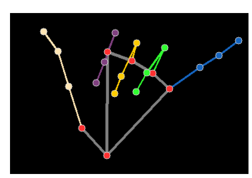

Figure 1: Processed Wireframe for the Hand sign "Y".

Next, we transformed the wireframe images into a set of joint angles. To accomplish this, joint angles were extracted from each wireframe image. Taking three data points (each normalized with a reference to the wrist or "point 0"), the

[1] https://www.kaggle.com/datasets/grassknoted/asl-alphabet

angle of individual fingers relative to each other was first calculated. The use of joint angles allowed us to simplify the model used for classification. We use RandomForests to fit a model to the data. The use of RandomForests, and its underpinning decision trees, provide a degree of interpretability for our classifications. The RandomForest model is also more memory efficient compared to our deep network for wireframe classifications. Finally, the use of joint angles allows the detection to reconcile variations in the size of the palm (metacarpal) or the length of the fingers (phalanges).

Our wireframes and the various joint angles that they encapsulate can be used as a teaching tool to support language acquisition. Initially, we defined standards for each hand sign and gathered a dataset of finger angles to determine the average finger angles for every letter. This dataset of average joint angles for each ASL sign serves as the reference for identifying deviations.

The teaching mode of our system involves two phases. In the first phase, we construct wireframes and compute finger angles from raw images of hand gestures representing ASL signs. Second, we use our classifier to identify the sign being attempted by the user. The predicted ASL sign is then cross-referenced with the established finger angles standard for that sign. This is done by identifying deviations for each individual finger and the overall similarity of the inputted hand sign. To find the similarity, each finger angle in the inputted image was subtracted from the average finger angle and divided by the average joint angle to find the extent of deviation. Then, the data regarding the accuracy of each finger joint and the overall average accuracy of the inputted hand sign are returned and displayed (figure 2). Since numerical feedback isn't always easily interpretable, we plan on tracing the average wireframe on the raw input image to guide users where their fingers need to move.

```
                         ACCURACY OF HAND SIGN
DETECTED SIGN : Y
FINGER ANGLE  : Thumb - Index | Index - Middle | Middle - Ring | Ring - Pinky
DEVIATION     : 0.1150        |0.0823          |0.1392         |0.0719
AVG DEVIATION : 0.1021
```

Figure 2: Example Output for Hand Sign Y

Discussion of Results

We partitioned our curated wireframes dataset into training, validation, and test datasets using an 80:15:5 split. We profiled the accuracy and loss metrics of the model built to classify wireframes at the end of training. We found that we had achieved a 94% accuracy over the test data set (validation and training accuracy depicted in Figure 2). We used joint angles and RandomForests to improve performance (giving us a 97% accuracy on the test set)

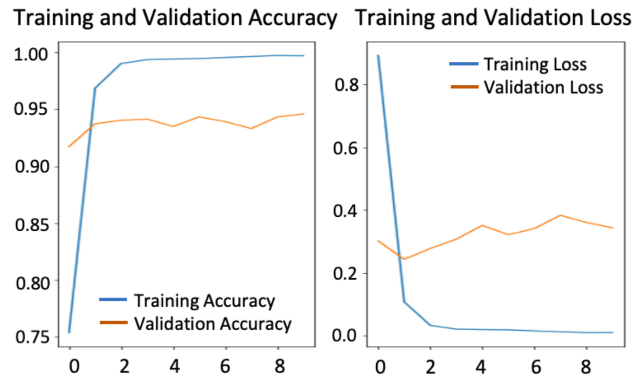

Figure 3: Classification Model Accuracy and Loss Graphs.

Conclusion

Our methodology presents a novel approach to ASL gesture classification by extracting wireframes from images to enhance the accuracy and efficiency of sign recognition systems. Wireframe representations eliminate factors such as skin tone, texture, and finger thickness that could introduce biases into the dataset.

The curated dataset of wireframe images facilitated the training of a deep neural network, resulting in a robust classifier achieving a 94% accuracy. Moreover, our approach doesn't just focus on classification but extends to teaching support in language acquisition by pinpointing specific errors in their hand gestures.

Acknowledgements

Sreedharan's research is supported in part by NSF 2303019. We would also like to thank Dr. Nathaniel Blanchard for his help and feedback, in regard to this project.

References

John, J., Sherif, B. 2022. Hand Landmark-Based Sign Language Recognition Using Deep Learning. In Machine Learning and Autonomous Systems: Proceedings of ICMLAS 2021 (pp. 147-157). Singapore: Springer Nature Singapore.

Pangestu, Y.; Heryadi, Y.; Suparta, W.; Arifin, Y. 2022. The Deep Learning Approach For American Sign Language Detection. 2022 IEEE Creative Communication and Innovative Technology (ICCIT), Tangerang, Indonesia, 2022, pp. 1-5 doi: 10.1109/ICCIT55355.2022.10118626.

Ravikiran, J.; Kavi Mahesh.; Suhas Mahishi, R.; Dheeraj, S. Sudheender.; and Nitin V. Pujari. 2009. Finger detection for sign language recognition. Proceeding of The International MultiConference of Engineers and Computer Scientists 2009 Vol I IMECS 2009,Hong Kong, March 18 - 20.