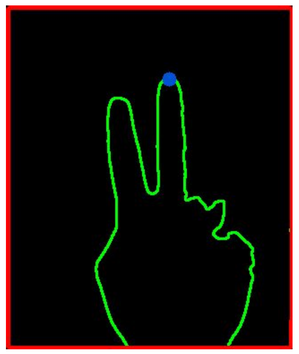
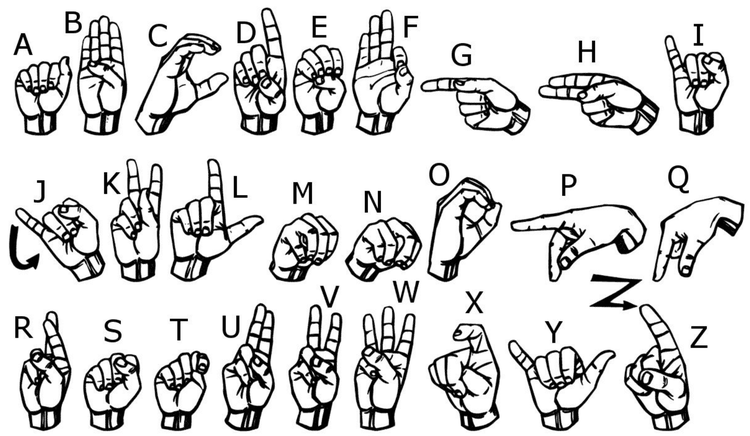
Signum: A real-time American Sign Language Translator

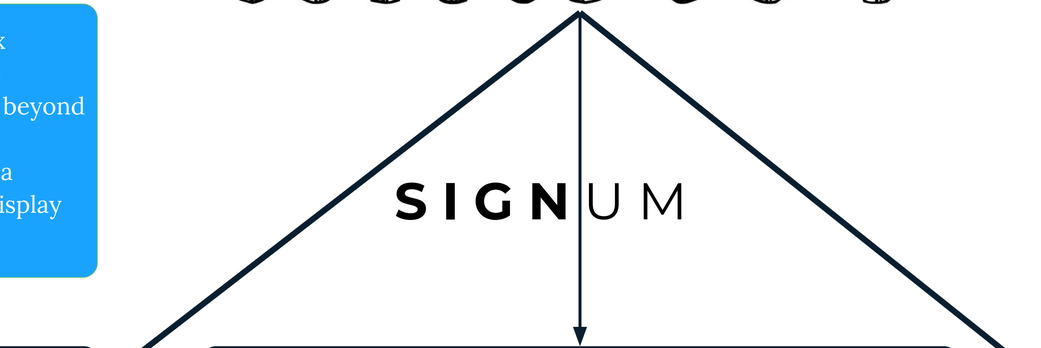
Diego Berny, Isaac Vandor, Utsav Gupta
Olin College of Engineering // Software Design SP18

About Signum

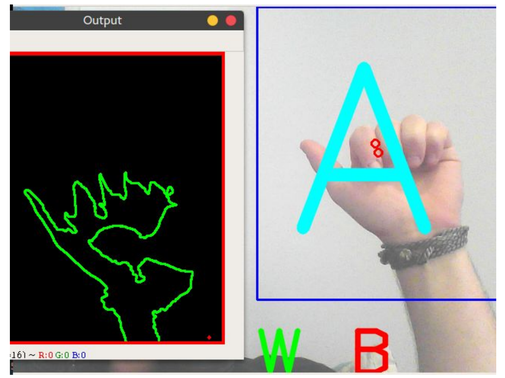


As the **sixth most used language in the U.S.** and a complex language with its own grammar and syntax, American Sign Language translation is an interesting project with **impact** beyond the scope of the Software Design class. We have created a program that can **detect** hand gestures, compare those to a **learned model** of American Sign Language gestures, and display the **predicted ASL gesture** onscreen.





Results



During **validation** of the model, our system resulted in an accuracy of **99.85%**. While this is impressive, the **real world performance** of the system is much less accurate as accurate hand gesture detection and **inconsistencies** between the dataset and the live frame caused the model to perform below the initial validation accuracy.

Computer Vision

We utilized **OpenCV,** a powerful computer vision library, to detect hand gestures. It allows us to use the computer's camera to see, and **manipulate** the images in order to account for differences in the background, lighting, proximIty, etc. Accounting for all of these factors is very important, because even small differences between hands could make it impossible to **recognize** the similarities between them. Properly **filtering** the input frame to the learned model had a huge effect on the accuracy of the predicted results.

Machine Learning

In order to determine what symbols are being shown in front of the camera, we implemented a **convolutional neural net** to **classify** the kinds of images it sees as a certain letter in ASL. For the algorithm to actually learn the meaning underlying gestures, we trained it on a dataset of **7,000 images** of people signing the ASL alphabet. From this set of training images, it learns **similarities and patterns** for the same letters so it can then recognize when they come up again in front of the camera.

Text-to-Speech

Once we have text output from the rest of the program, we leveraged Google's text-to-speech library gTTS to help us convert it to **spoken word.** This provides an additional interaction method for the end user, making Signum more **accessible** for all.

Resources Used: OpenCV - A computer vision library, Keras - Machine Learning Implementation, gTTS - Google's text-to-speech library, HTML5UP! - Used for the project website template