Problem Statement

Communication is a fundamental necessity in every social scale

There is a need to solve language barriers

Most research and developments focus on verbal language translation

This puts the deaf community at a disadvantage

   29.5 million population in the U.S.
Our Solution?

Create a prototype ASL translator!
Background
Machine Translation

The translation of natural languages by machine

Traced back to the 1950s

Largely neglected ASL until the 2000s

Liwei Zhao and his team from University of Pennsylvania worked on one of the first attempts to create an MT system for translating English to ASL (2000)

 Provided insight on essential design principles for translating ASL
MT: Liwei Zhao et al.

Described a MT system that took into account the linguistic, visual, and special information associated with ASL.

Represented ASL as a fully articulated 3D human model.

Took into account the phonology (hand shape, location, movement) and morphology of ASL to capture its linguistic complexity.

Used a Lexicalized Tree Adjoining Grammar based system to translate between English and ASL glosses.
Gesture Recognition

X. Zabulis et al. from the Institute of Computer Science Foundation for Research and Technology - Hellas (FORTH) Heraklion, Crete, Greece 2009

- One of the most recent work on Gesture Recognition for Human-Computer Interaction

Jayashree Pansare et al. from M. E. Society’s College of Engineering, Pune India, & Devi Ahilya University, Indore, India 2012

- Real-Time Static Hand Gesture Recognition for ASL
Zabulis: Vision Based Hand Gesture Recognition

Provides insight on the current state of vision based Hand Gesture Recognition

**Detection Techniques** - Color, Shape, Pixel Values, 3D model, Motion

**Tracking Techniques** - Template Based, Optimal Estimation Techniques, Mean Shift Algorithm, Particle Filtering

**Recognition Techniques** - Template matching, Principal Component Analysis, Boosting, Contour and Silhouette matching, model-based recognition, HMMs
Pansare: Static Hand Gesture Recognition (ASL)

One of the first ASL Gesture Recognition systems

Uses a webcam and RGB color space

Able to recognize the 26 alphabet signs in ASL
Current Technology

1. By SignAloud team
   - using gloves extracting skeletal motion

2. By MotionSavvy team
   - using an equipment with two cameras: Leap Motions
SignAloud
MotionSavvy
Our Objective

Create a prototype real-time ASL to English (audio) translator using a camera input: that translates numbers, letters and a set of common words & phrases

Create a real-time English (speech) to ASL (video) translator

Why? A basic camera is very accessible

Extend single camera gesture recognition approaches to non-static gestures
Design Approach
What we want to implement?

1. Take input from camera
2. Image Processing using openCV
3. Extract meaningful gesture
4. Machine Learning using scikit-learn
5. Find out the corresponding gesture
7. Display text on screen
8. Translate into text
9. Spoken English on the speaker
10. Festival
11. Take input from a microphone
12. Sphinx
ASL to English
Extract hand gesture image - using OpenCV

1. Separate hand and arm image from the background
Extract hand gesture image - using OpenCV

2. For static hand recognition: Track the position of fingers
Extract hand gesture image - using OpenCV

3. For moving hand recognition: Track the movement of fingers
Get meaning from gesture
- scikit learn & openCV

1. Cluster similar shapes

A  E  S  T  N  M
D  R  G  I  ...

The Fingerspelled Alphabet
Lifeprint.com
Get meaning from gesture
- scikit learn & openCV

2. Classify different gestures in the group
Interpreting Gesture and Representing as Text

All simple gestures will be added to the program to simplify communication and given a word depending on the action. When the action is performed the program will recognize it and immediately come up with the textual response for the other person.
Text into speech

Our approach will be based on previous work to keep the focus on gesture recognition.

The Festival Speech Synthesis System (Festival for short) is a framework for Text-To-Speech Conversion.

Pyfestival is a Python wrapper around Festival.

So, we will use Pyfestival which is an existing text-to-speech library.
English to ASL
Speech into Text

Use Speech Recognition

- **API** - converts spoken language (in microphone) into written text (python strings)
- Basically a person speaks into the program
  - We will use existing library for speech recognition
- The program records the audio from the microscope
  - The audio is recorded using the speech recognition module
- Then sends it to the speech API
- As a result, it returns a Python string
  - Output could look like `r.recognize_(audio)`
Text into hand gesture (Sign Language)

When converting text into gestures, the program will first analyze whether it has the gesture stored

- If it is stored then an image such as a GIF or short video will appear on the screen
- If the gesture is NOT stored then it will break down the text letter by letter and illustrate the gesture in letters only which will spell out the name of the gesture
Hand gesture (Sign Language) into Graphic
- using Vpython (graphical tool)
- has simpler modules to work with and there are many ways of animating them
- 3D graphic in a sphere or cylindrical shape to that it can easily recognize hands
- hand model simulates ASL on screen
Deliverables

Primary: An executable program for computers with cameras

Secondary: An application for smartphones
ASL to English

Research and Analysis for optimal approach - 2 weeks
Development of Number Recognition - 4 weeks
Development of Alphabet Recognition - 3 weeks
Development of Word Recognition - 3 weeks
Text-to-Speech Implementation - 1 week
English to ASL

Learning Speech Recognition for Speech-to-Text function - 1 week

Developing translation algorithm for English text to ASL gloss - 3 weeks

Developing computer animation of ASL - 3 weeks
Total of 20 Weeks
Conclusion

Our team hopes to expand the current gesture recognition systems that use single camera inputs by developing an approach to interpret non-static gestures.

Ultimately, we want this technology to be accessible to everyone, and can easily be used for daily interactions.
Questions?