**SIGN HAWK**

**Mentor:** Assistant Professor Shilpa

(Department of Computer Science and Engineering)

**Team Members:**

Rishi Dhawan (18103013)

Aryan Singla (18103062)

Manav Singla (18103065)

Suhel Naryal (18103092)

**Project Description**

**Introduction:**

Communication with the hearing impaired (deaf/mute) people is a great challenge in our society today, this can be attributed to the fact that their means of communication requires an interpreter at every instance. According to WHO, around 5% of the world’s population – or 466 million people – has disabling hearing loss out of which 432 million are adults and 34 million are children. Around 3 million people in the world know sign language. We observe that there is a huge gap between the hearing population and non-hearing population. So, to bridge this gap, we aim to develop an AI sign language translator capable of translating sign language captured in a video to text. The video provided to the translator can be an already recorded video or live feed from the camera.

**Proposed Solution/Features:**

● Translate sign language to text.

● Translate from existing signs video.

● Translate live signs using a web-cam.

**Previous Works:**

· Kinect Sign Language Translator: It’s a sign language translator developed by Microsoft. It uses Kinect technology for 3D human pose estimation. This application is under development.

· Google is also developing a Sign language translation solution which uses 2D images from camera for pose estimation.

**Technical Details:**

* FrameWork Used - Django
* For front - end we have used HTML-CSS and Bootstrap with JavaScript for dynamic pages
* For detecting Signs we have used Deep Learning (CNNs & Bidirectional LSTMs)

**ROLE OF WEB TECHNOLOGIES IN PROJECT**

The use of **Web Technologies** has been made in the front end of the portal, multimedia streaming and back end of the portal.

**Front End**:

The front end of the portal consists of **HTML** pages which has the basic layout of all the content of the page, and for the designing and beautification **CSS** has been used. **JavaScript** has been used to add dynamic buttons while turning on the webcam for live stream and for showing up alerts. We are using **pagination HTML** class with **AJAX**  to show different signs in our database with their respective clip.

For the purpose of convenience all of the pages have been designed using **Bootstrap4** classes.

**Multimedia Streaming:**

To capture live webcam streams, we are using an API called **getUserMedia**.

With the help of this API we are both streaming and recording the webcam stream.

To transfer this recorded stream to the client side we are using **AJAX**.

To start or stop recording we are using **JavaScript**.

**Back-end:**

We are using Django Framework for back - end development. We are using the django rest framework to provide restful services in a project.