CSE 473/573 - COMPUTER VISION AND IMAGE PROCESSING
SIGN LANGUAGE TO PHRASE CONVERTOR
PROJECT PROPOSAL

Vignesh Prakash
UBID : 50478782
prakash8@buffalo.edu
Github Link

ABSTRACT

This project aims to provide real-time communication for individuals with Vocal Impairment by converting sign language into written phrases which is displayed as subtitles on a video feed. My aim is to make communication possible for people who don't know sign language to understand it, and to enable more efficient communication between differently-abled individuals and their non-disabled peers. This system will help bridge the communication gap and promote inclusivity, allowing individuals with vocal impairments to fully participate in conversations and social interactions.

1 Introduction

1.1 State of the Art

Currently, there are numerous Machine Learning models that use Convolutional Neural Networks to identify characters in sign images. However, these models often struggle with identifying characters in real-time video streams, as they are typically designed for static image analysis. To overcome this limitation, I plan to implement a YOLOv5 model in Pytorch. This model allows for real-time detection of characters in a live camera feed, and can even predict best possible words from a list of commonly used chat phrases.

1.2 Input and Outputs

1.2.1 Input

To complete this project, two inputs are required. The first is an image dataset for sign language detection. I found two suitable datasets on Kaggle for this purpose: the ASL Alphabet and the Sign Language Gesture Images Dataset. For predicting words, I plan to use open source ebooks and create a Markov Model. This approach can help generate a statistical model that predicts the likelihood of each word based on the words that come before it in the text.

1.2.2 Output

The goal of this project is to display generated text as a subtitle to the live camera feed in real-time. To achieve this, the YOLOv5 model will first predict characters from the sign language gestures in the live camera feed. The predicted characters will then be fed into the Markov Model to predict the most likely words, which will be displayed progressively to the viewer.

Therefore, the intermediate output of this project would be the predicted characters generated by the YOLOv5 model, while the final output would be the best possible result displayed as a subtitle in real-time

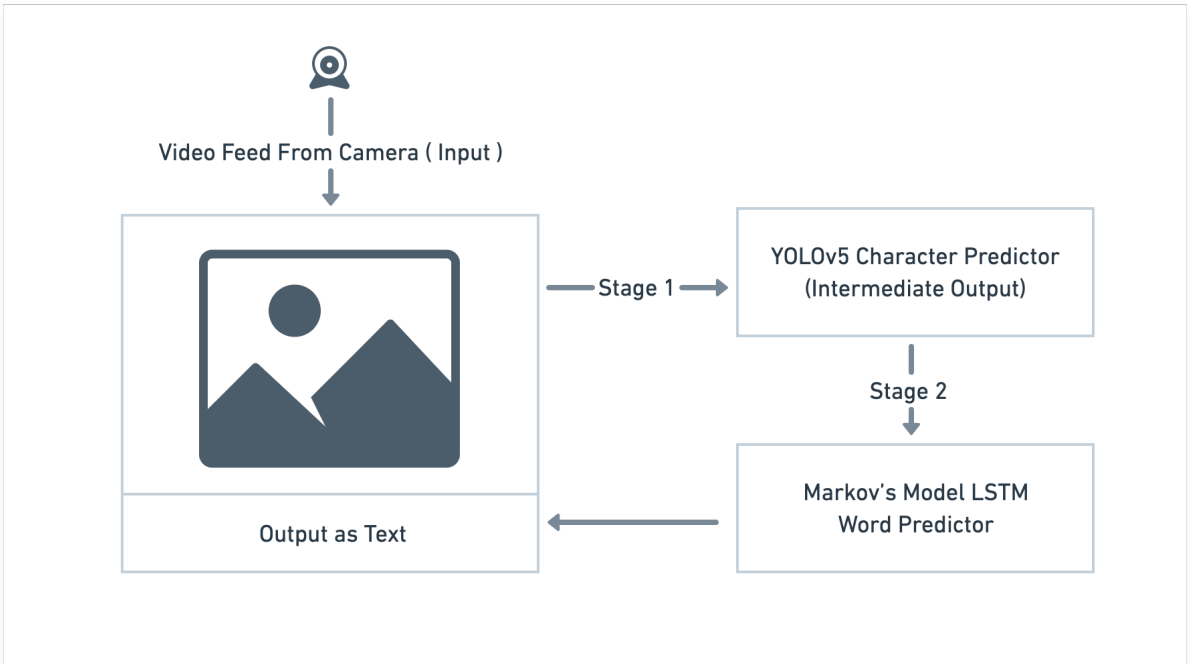

Figure 1: Flowchart

2 Data

To complete this project successfully, I need two datasets: an image dataset and a word collection dataset. To ensure the image dataset is representative and avoids any bias in the background, I have sourced it from two different sources on Kaggle. Together, the combined dataset contains about 198,000 images with a resolution of 50px $\times$ 50px.

For the word collection dataset, I have opted to use an open source e-book collection, which includes over 10,000 books from various genres and authors. This dataset offers a vast and diverse vocabulary that is well-suited for building a Markov's Model.

3 Coding Resource Requirements

For the scope of this project, I plan to use three libraries: opencv-python, PyTorch, and YOLOv5.

opencv-python, is a library commonly used for image and video processing. In this project, I plan to use it to preprocess the images by converting them to grayscale, thresholding them, and producing a binary image suitable for YOLOv5.

PyTorch, is a popular deep learning framework that provides efficient GPU support for training neural networks. I plan to use PyTorch to develop a convolutional neural network (CNN) that will enable image classification and the word prediction model.

YOLOv5, is an object detection algorithm that uses deep learning to detect and classify objects in real-time. I plan to use YOLOv5 in conjunction with PyTorch to create a model for identifying objects in the images.

As the project progresses, additional libraries may be incorporated as needed. While I don't plan to use complete code from any GitHub repository, I may reference some available code as inspiration for my own implementation.

The majority of the project will be implemented independently coded by myself.

4 Computational Resource and Effort Requirements

I plan to train the model entirely on CPU using my secondary machine. Currently, I don't have a precise estimate for the number of days required to complete the project. However, since I'm the only person working on it, I anticipate investing about an hour per day on average, resulting in an estimated total of 50 man hours.

Please note that this estimate may vary depending on the project's progress and any unforeseen complications that may arise

5 Evaluation

The success of the project is defined by the effectiveness in facilitating the real-time communication between individuals who do not know sign language and those who are vocally impaired. To achieve this, the system must accurately translate sign language into text with the intended context. The success of the project can be measured by evaluating the prediction and contextual errors made during the translation process.

One way to quantify the accuracy of the system is by using a metric such as the F1 score, which takes into account both precision and recall. This metric can be calculated by comparing the generated text with the ground truth text annotations and assessing how well the system captures the intended meaning of the sign language.

The accuracy metric can be reported as a percentage or score, which would indicate how well the system is able to accurately convert sign language into written text. This metric can be used to track the system's performance over time and identify areas for improvement.

6 Project Expectations

The most exciting aspects of this project is the opportunity to apply my knowledge in computer vision technology to improve communication accessibility for individuals with vocal impairments. By providing real-time sign language translation, I can help to break down communication barriers and help the vocally impaired people to voice out their problems.

From this project, I hope to gain practical experience in computer vision models by working with the YOLOv5 library. I aim to improve my understanding of computer vision technology and apply it on my future projects.

Also, I am excited about the opportunity to learn American Sign Language (ASL) as part of this project. ASL is a valuable skill set and this project provides a chance to combine my passion for computer vision with my desire to learn the sign language.