Sign Language Translator Plag Report.docx.pdf

Shri Vile Parle Kelavani Mandal

Document Details

Submission ID

trn:oid:::9832:70314686

Submission Date

Nov 6, 2024, 3:00 PM GMT+5:30

Download Date

Nov 6, 2024, 3:01 PM GMT+5:30

File Name

Sign Language Translator Plag Report.docx.pdf

File Size

149.8 KB

12 Pages

1,527 Words

8,447 Characters

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

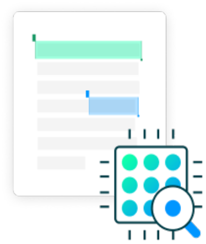
1 AI-generated only 0%
Likely AI-generated text from a large-language model.

2 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.



False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

SIGN LANGUAGE TRANSLATOR

Roll No.	Name	SAP ID
B116	Adit Khandelwal	70022200119
B119	Suhani Pugalia	70022200075
B134	Vishwa Bhanushali	70022200409

ABSTRACT

This project will focus on hand gesture recognition in real time using computer vision and machine learning. It captures images of hands by webcam, processes them, isolates the hand region, resizes it, and a CNN model will classify the gestures into predefined categories; thus, the system will interpret hand gestures in real time. These hand gestures are read precisely and continuously; so the corresponding characters that come up on the screen make them helpful for communication aids or control interfaces.

INTRODUCTION

There is a huge communication gap within the sign language community existing between sign language users and nonusers. This project aims to develop a hand sign language translator that can identify individual letters of the alphabet, which will be the first step in further translating sign language into readable text.

Using the OpenCV computer vision library, fully open source and flexible to capture and process hand gestures in real time, our system utilises machine learning models and image-processing techniques to identify particular shapes of hands that represent letters in the American Sign Language alphabet, thus being able to recognize letters letter by letter, then words and phrases.

The technical approach, design of the system, and implementation details based on the image segmentation, feature extraction, and classification technique for hand gesture analysis are provided in this report. It also deals with such challenges as background noise and lighting variation and diversified hand shapes and the measures used to increase the precision of the model and the confidence of the results. Although the system is only able to work with letters, it will eventually build up to a fully functional sign language translation system.

PROBLEM STATEMENT

This can be applied to human-computer interaction, assistive technology and sign language. While the system is also designed to be quick, precise and reliable, it is very challenging to achieve these metrics in the presence of varying light conditions, hand-orientation position, and also small variances between gestures. The aim of this project is to provide a solution to the requirement net result of such a system to recognize and interpret hand gestures via machine learning model and a camera feed in real time to allow touchless communication and control.

PROPOSED SYSTEM

This is an individual letter-based sign language translator system for the American Sign Language (ASL) alphabet. In this section, we will describe and outline the main components as well as functions of the proposed system on how hand signs are read, classified, and eventually output in terms of text.

1. System Architecture

The basic architecture of the system mainly involves these four core modules responsible for carrying out some particular functionalities for effective recognition of the gesture.

- Input Module (Image Capture): The input module employs a camera that captures video frames in real-time of the hand of the user. This module can be seen as the first step that will feed the visual input for further processing.
- Preprocessing Module: The images captured in this stage are prepared for analysis. Some of the key preprocessing steps include resizing, converting the images to grayscale, and background subtraction that minimises noise and maximises the accuracy of the feature extraction process.
- Feature Extraction: After the data preprocessing, the system discovers some specific features that constitute a hand gesture contour, edges, and even features which are crucial for differentiation from another letter gesture. They form a base for correct classification.
- Classification Model: Now a machine learning model, labelled with images of the ASL letters is being trained, and upon being classified by the process, extracts the features by the process, and after having been matched with one among known ASL hand shapes, guarantees correct recognition.
- Output Module: The final module makes the recognized letter appears in text on the screen so that users can read real-time each letter that is spelled out to eventually form words or phrases

2. Modules and Techniques

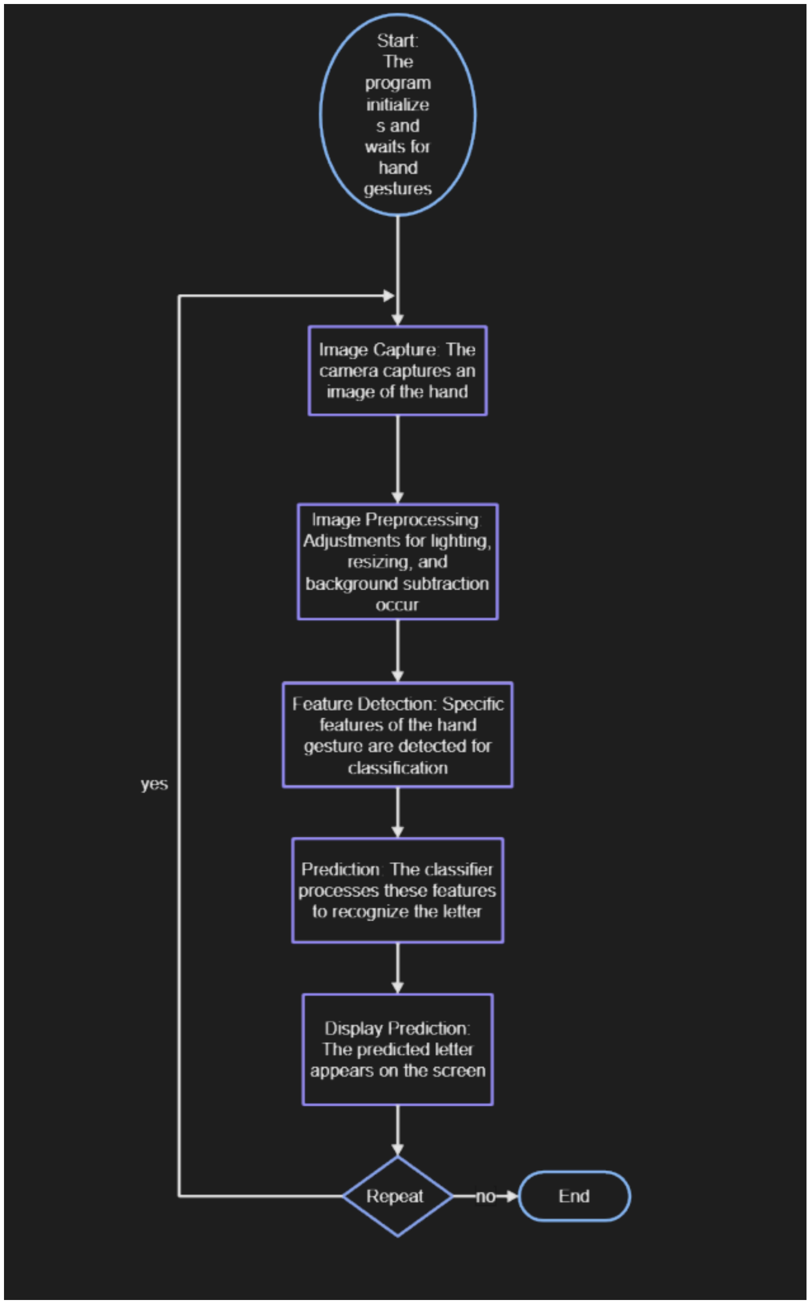
Each of the modules uses specific techniques and algorithms towards achieving accurate gesture recognition;

- Image Preprocessing: Techniques which include conversion to grayscale with the use of Gaussian blur enhance the reduction of the background noise through thresholding while ensuring good hand segmentation.
- Feature extraction: It uses contour or edge detection to find shape features of hands. Since all the letter gestures differ in some respect and characteristics, it supports effective classification.
- Classification Model: A CNN, or other machine learning classification model, is trained on diverse ASL letter data with the goal of letter-based recognition using the features seen. This model can help distinguish between very similar, slightly different gestures, giving a consistent letter recognition model.

3. System Requirements

The system is quite lightweight and requires minimal computational hardware and software resources in that:

- Hardware: webcam standard.
- Software: OpenCV library for processing images, TensorFlow is a machine learning library used for classification model; display module where text should be output.
- Performance Requirements:Real-time recognition at low latency allows for immediate feedback while in use.


Start: The program initialize s and waits for hand gestures
Image Capture: The camera captures an image of the hand
Image Preprocessing Adjustments for lighting, resizing, and background subtraction occur
Feature Detection: Specific features of the hand gesture are detected for classification
Prediction: The classifier processes these features to recognize the letter
Display Prediction: The predicted letter appears on the screen
Repeat
no
End
yes

WORKING

Data Collection:

- A webcam is used to capture hand images in real time. A cvzone library uses a HandDetector for single-hand detection in every frame.
- When a hand is detected, the coordinates of a bounding box around the hand are computed adding extra offset for optimal cropping.
- The cropped hand image was resized and centred onto a 300x300 white background image with uniform input for the classifier.

Gesture Classification:

- It sends the processed hand image to a pre-trained CNN model trained on labelled gesture images and saved as keras_model.h5.
- Every tracked movement is related to some symbol (for example "A," "B," "C", and further). The result of such a model is prediction according to the hand movement within a certain frame.

Consistency Check and Text Recognition:

- For accuracy, it checks if the same gesture is kept for some specified amount of time. It checks if the gesture has stayed the same for some time of 3 seconds; if so, then the gesture is recorded and added to the text display as a valid character.
- This accepted text shows it on the screen that makes it easier for users to see their translated gestures in real time.

Display Output:

- Camera feeds come with a gesture recognition overlay such that the user can see his hand gesture and its interpretation by the system.

Exit Mechanism:

- The system captures images until the user presses the ESC key, at which point resources are released, and the windows are closed.

RESULTS AND OUTCOME

The hand sign language translator effectively demonstrated its ability to recognize individual ASL letters, providing real-time feedback and translating gestures into text.

1. Recognition Accuracy

The translator successfully recognized a broad range of ASL letters. Clear, well-defined gestures were consistently identified with minimal errors, especially for letters with distinct hand shapes. Some letters with similar shapes posed slight challenges, but overall recognition was satisfactory for most users.

2. Speed and Response Time

The system gave instant feedback, and the users were able to report a fluid interaction as the translator showed the recognized letters almost right after every gesture. This prompt response led to an intuitive interaction that was natural and even suitable for spelling words letter by letter.

3. Performance in Different Environments

The system was tested in different environments to check its ability to adapt:

<u>Lighting:</u> The performance of the translator was maximum in moderate lighting where shapes of hands are clearly visible. In extremely bright and dim lighting, the performance is slightly variant, suggesting that consistent lighting is key to good results.

<u>Complexity of background:</u> In simple backgrounds where the hand can be easily isolated from the background, the system works well. In the case of busy or cluttered backgrounds, it became more difficult to achieve an accurate recognition; however, adjustments in preprocessing helped enhance the performance.

4. User Experience and Usability

The overall general good experience was testified to by user feedback with regard to the translator's use. The system of real-time text enabled fluid spelling of words; most found it intuitive. Some gestures have to be practised to use correctly, especially if the shape is complex enough like letters.

5. Observed Limitations

Some limitations came out during the testing:

- Light Sensitivity with complex background: Changed at times recognition conditions so need to have stable recognition.
- Limited recognition: Translator might recognize a letter and at times one would write an entire word letter by letter so it takes more time on making larger words or even phrases.

CONCLUSION & FUTURE SCOPE

The computer vision, deep learning and also realtime hand gesture recognition system proposed is really implemented. Using a CNN model and a consistency check, it is now accurately recognizing predefined hand gestures as the characters in the video stream. The system could be useful in assistive technology as a communication supplement for individuals that are speech or hearing impaired. The potential near futures of the project range from gesture vocabulary expansion, advancing the model to yield results at higher accuracy levels given variations in gesture conditions, to expanding its capability and modality in scope.