RESEARCH PAPER

SIGN LANGUAGE TRANSLATOR

1) Prathamesh Nanasaheb Raut

2) Yogesh Balaji Reddy

3) Aniket Sandipan Tandle

Abstract:

Sign language is the primary communication method for deaf and hard-of-hearing (DHH) individuals, yet it remains unfamiliar to most non-DHH people, creating a significant communication gap. To address this, we propose an AI-powered Sign Language Translation system that utilizes computer vision and deep learning to interpret hand gestures in real time and convert them into readable text. Our system is based on a Convolutional Neural Network (CNN) model trained on American Sign Language (ASL) datasets and uses webcam input for gesture recognition. This paper outlines the design, methodology, and implementation of the system, discusses the challenges in sign language translation (SLT), and highlights possible improvements using transformer-based approaches.

Introduction:

Communication is a fundamental aspect of human interaction. For deaf and hard-of-hearing (DHH) individuals, sign language serves as the primary means of communication. However, the lack of widespread understanding of sign language among non-DHH people creates a significant barrier, leading to social isolation and communication challenges for the DHH community.

Sign Language Translation (SLT) aims to bridge this gap by converting visual gestures into text or speech, enabling smoother interaction between DHH and non-DHH individuals. While human interpreters are effective, they are not always available, and services can be costly. Recent advancements in artificial intelligence (AI), computer vision, and deep learning have opened the door to automated SLT systems that are both accessible and scalable.

This project presents an AI-powered sign language translator that recognizes hand gestures in real-time using a webcam and translates them into corresponding textual output. By employing a Convolutional Neural Network (CNN) model trained on an American Sign Language (ASL) dataset, the system identifies individual letters and outputs their meaning. The project leverages technologies like Python, OpenCV, and TensorFlow to demonstrate a functional and low-cost translator system that can run on standard hardware.

The goal of this system is to promote inclusivity and make everyday communication more accessible for the deaf and hard-of-hearing population.

➢ System Overview:

The system workflow consists of the following major components:

1. **Image Input:**

   A webcam captures hand gesture images in real-time.

2. **Image Preprocessing:**
   The captured image is converted to grayscale, resized (e.g., 64×64 pixels), and normalized to reduce noise and enhance feature extraction.
3. **Gesture Classification:**
   A pre-trained Convolutional Neural Network (CNN) model is used to classify the input gesture into one of the sign language classes (e.g., A–Z).
4. **Output Display:**
   The predicted gesture is displayed as text on the screen and optionally stored for forming complete words or sentences.

➢ Technologies and Tools Used:

Component	Tool / Library
Programming	Python
Image Processing	OpenCV
Deep Learning	TensorFlow
Dataset	American Sign language (ASL) Alphabet Dataset

➢ Model Architecture (CNN):

The system uses a Convolutional Neural Network (CNN) trained on labeled images of hand signs. A typical architecture looks like:

- Input Layer: 64x64 grayscale image
- Convolutional Layers: Feature detection with ReLU activation
- Pooling Layers: Dimensionality reduction
- Fully Connected Layer: Gesture classification
- Output Layer: Softmax activation for 26 sign classes (A–Z)

➢ Data Preprocessing:

- Resize all input images to a consistent shape (e.g., 64x64)
- Convert color images to grayscale for simplicity
- Normalize pixel values to range [0, 1]
- Augment data (flip, rotate) to improve model generalization

Literature Review

Previous approaches to sign language translation ranged from expensive sensor gloves to vision-based systems using deep learning. Recent works have shown that Convolutional Neural Networks (CNNs) can effectively recognize static ASL alphabet gestures with high accuracy. Some studies also explore transformer models for sentence-level translation, but these are often computationally intensive. Our project focuses on a lightweight, real-time CNN-based system using webcam input to provide an affordable solution for daily communication.

Implementation

The AI-powered Sign Language Translator was developed using Python with key libraries including OpenCV for image capture and preprocessing, and TensorFlow/Keras for building and training the Convolutional Neural Network (CNN) model.

I. Dataset

We used the publicly available American Sign Language (ASL) Alphabet dataset, which contains thousands of labeled images of hand signs representing the 26 letters A through Z. The dataset was split into training (80%) and validation (20%) sets.

II. Model Training

The CNN model architecture consists of:

- Multiple convolutional layers with ReLU activation for feature extraction
- Max pooling layers to reduce spatial dimensions
- Fully connected layers for classification
- Softmax output layer to predict one of the 26 classes

The model was trained for 20 epochs using the Adam optimizer and categorical cross-entropy loss function. Data augmentation techniques such as rotation and flipping were applied to improve generalization.

III. Real-time Prediction

Using OpenCV, the system accesses the webcam feed and captures frames. Each frame is preprocessed by resizing to 64×64 pixels and normalizing pixel values before being passed to the trained CNN model. The predicted letter is then displayed on the screen in real time.

IV. User Interface

A simple GUI was implemented using OpenCV's overlay text functions to display predictions live. The system can be run on any standard laptop or PC with a webcam, making it accessible and low-cost.

Results and Discussion

The CNN model achieved an accuracy of approximately **92%** on the validation dataset, demonstrating effective recognition of ASL alphabet gestures. During real-time testing with a webcam, the system accurately predicted most static signs with minimal delay, confirming its usability for basic sign language translation.

However, some challenges remain, such as sensitivity to lighting conditions and variations in hand orientation, which occasionally caused misclassification. The model is limited to static alphabet signs and does not yet support full sentence translation or dynamic gestures.

Future improvements could include integrating transformer-based models for better temporal understanding, expanding the dataset to cover dynamic signs, and developing a more user-friendly interface.

Conclusion and Future Work

This project demonstrates a real-time AI-powered sign language translator that recognizes static ASL alphabet gestures using a CNN model. The system offers an accessible and cost-effective solution to help bridge communication gaps between deaf and hearing communities.

While effective for individual letters, the system’s scope is currently limited to static signs and requires improvements to handle dynamic gestures and full sentence translation. Future work will focus on incorporating advanced models such as transformers, expanding the dataset, and enhancing the user interface for practical daily use.