Sign To Speech: Camera Based Sign Language To Speech Conversion

Darsh Thakur, Syed Shariq, Yuvraj Singh, Omkar Pawar

Guide: Prof. Prashant Yelmar
MIT SOC, MIT ADT University

Submitted: SUBMITTED TO MIT SCHOOL OF COMPUTING, LONI, PUNE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE.

ABSTRACT

Sign language users often face significant communication barriers with non-signers, hindering inclusivity and accessibility. This project aims to overcome these challenges by developing a real-time sign language-to-speech recognition system. The proposed system converts sign language gestures into spoken words or phrases using advanced computer vision, deep learning, and gesture classification techniques.

A camera captures the user's gestures, and the input is processed through a deep learning model trained on a comprehensive sign language dataset. Recognized gestures are then converted into speech using the Google Text-to-Speech (GTTS) library, providing a cost-effective, portable, and efficient solution for real-time communication.

This system bridges the communication gap between sign language users and non-signers, empowering individuals with hearing or speech impairments to engage seamlessly in personal, social, and professional interactions. By eliminating the need for human interpreters and offering a scalable, automated alternative, the project promotes accessibility, independence, and social integration for sign language users.

Keywords: Convolutional Neural Networks (CNN), Sign Language Recognition, Deep Learning Models, Real-time Gesture Translation, Google Text-to-Speech (GTTS)

INTRODUCTION

Sign-to-speech systems are designed to help individuals who use sign language communicate with those who don't, by translating gestures into spoken language. These systems combine technologies like gesture recognition, machine learning, and speech synthesis to enable real-time translation. They aim to break down communication barriers, promoting accessibility and inclusion for the deaf and mute communities. While current systems show promise, challenges remain, such as accuracy in fast gestures and dependency on specific hardware. The future of these systems lies in improving these limitations and making the technology more accessible and accurate.

Existing sign-to-speech systems use a variety of technologies to bridge communication gaps between sign language users and non-signers. Vision-based systems use cameras and computer vision to capture and recognize gestures, while wearable devices like smart gloves with sensors detect hand movements for conversion into speech. Machine learning models, including CNNs and RNNs, enhance the accuracy of gesture recognition. Smartphone apps also utilize cameras for real-time translation, and some systems use cloud-based platforms for better processing power and scalability. Hybrid systems combine multiple methods, like vision and sensors, to improve recognition. Despite progress, challenges like accuracy with fast gestures and hardware dependence remain.

The motivation behind creating sign-to-speech systems is to help people who use sign language communicate more easily with those who don't understand it. For many deaf and mute individuals, expressing themselves in a world where spoken language dominates can be challenging. These systems aim to break down those barriers, making everyday interactions like ordering food, asking for help, or participating in a meeting much smoother.

It's also about inclusion and fairness. Everyone should have the chance to participate fully in society, whether in schools, workplaces, or social settings. Sign-to-speech technology can make that possible by giving people a voice that others can hear and understand. With advancements in AI and wearable devices, this vision is becoming a reality.

On a bigger scale, projects like this align with global efforts to reduce inequality and create technology that serves everyone, not just a select few. It's about using innovation to bring people closer together and make life better for those who face communication challenges.

LITERATURE REVIEW

Sr No	Title	Author	Year	Summary
1	A Survey on Sign Language Recognition and Translation	P. K. P. Chowdhury, M. S. Islam	2020	Reviews techniques like machine learning for translating sign language to speech.
2	Deep Learning for Gesture and Sign Language Recognition	R. L. Shah, H. B. Bhalerao	2021	Focuses on CNNs and RNNs for sign language recognition and speech conversion.
3	Sign Language to Speech: Techniques and Challenges	M. S. Imran, A. K. D. Mollah	2019	Discusses visual and sensor-based methods for converting sign language to speech.
4	Sign Language Recognition with Deep Learning	P. S. Ghosal, S. Mondal	2022	Explores deep learning methods for real-time sign language recognition and speech translation.
5	A Survey on Sign Language Recognition using Machine Learning	H. S. Raut, S. S. Shinde	2021	Reviews machine learning techniques for sign language recognition and speech generation.
6	Deep Learning Approaches for Sign-to-Speech Conversion: A Comprehensive Survey	Gupta, R., & Ali, S	2022	Explores deep learning models, including CNNs and RNNs, for translating sign language gestures into synthetic speech.

1. Project Title:Sign To Speech Conversion Using Computer Vision: The system offers multilingual support, covering various sign and spoken languages like ASL, BSL, and ISL. It recognizes hand gestures, facial expressions, and body movements, translating them into natural-sounding speech via advanced speech synthesis. Users can customize and teach unique signs, receive error feedback for unrecognized gestures, and access the system offline. Designed to enhance accessibility for the deaf and mute communities, it bridges communication gaps in social and professional settings. Additionally, it integrates with wearable devices like smart gloves, supports camera-based systems, and serves educational purposes by teaching sign language to non-signers.

Users benefit from error detection and feedback features, with suggestions to improve unrecognized gestures. The system supports multilingual translation, ensuring context-sensitive speech output for ambiguous signs. A user-friendly interface displays recognized signs as text alongside speech for verification, while users can teach custom gestures saved in their profiles. Additionally, the system operates offline for areas with limited internet access, delivering real-time recognition and speech synthesis with minimal delays.

The system is designed for real-time performance, processing gestures with minimal latency to enable smooth communication. It is scalable, adapting to varying user inputs, languages, and regional sign variations without compromising reliability or performance. With a focus on accuracy, it ensures high recognition rates even for complex gestures or in distracting environments. Usability is a key priority, offering an intuitive and user-friendly experience requiring minimal effort, even for non-technical users. Compatibility across devices like smartphones, tablets, and wearable sensors ensures broad accessibility, along with multi-platform and operating system support. The system prioritizes security, safeguarding user data and biometric information, and is built with maintainability in mind, offering straightforward updates and clear documentation. Accessibility features make it inclusive for individuals with diverse abilities, including those with visual or motor impairments.

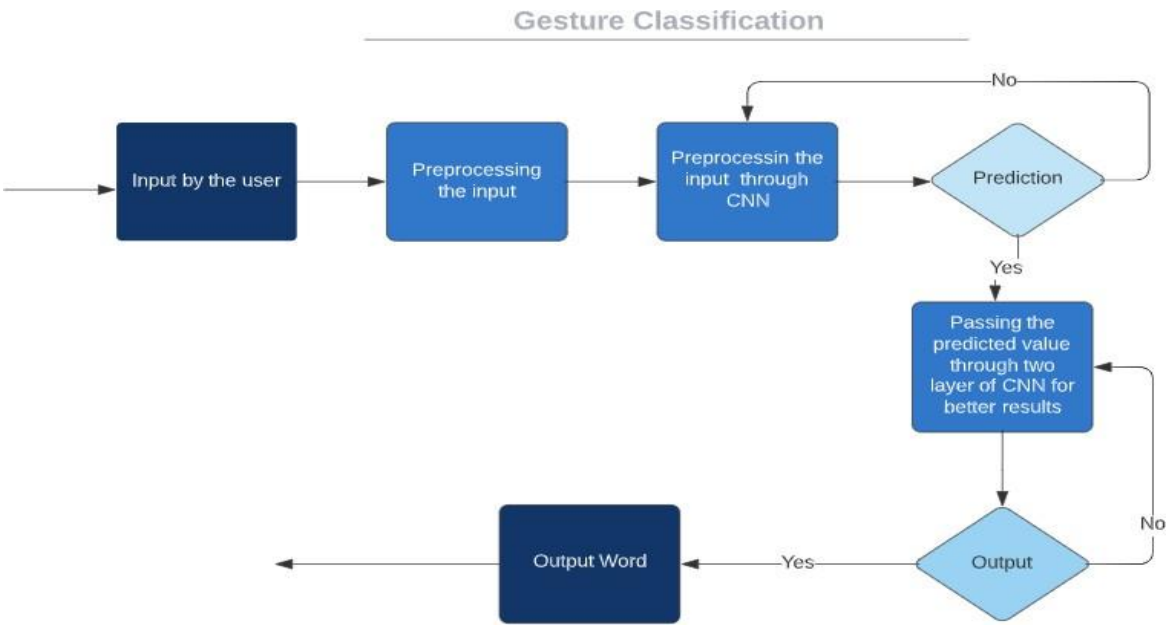
Despite its advanced capabilities, the system faces several limitations. Achieving consistent accuracy in recognizing complex gestures or handling variations in sign languages across different regions remains a challenge, particularly in noisy or poorly lit environments. The dependency on high-quality hardware, such as advanced cameras or wearable sensors, may limit accessibility for users with budget constraints. Real-time processing requires significant computational power, which could impact performance on less capable devices. Offline functionality, while essential, may lack the full feature set available in online mode, such as cloud-based language translation. Additionally, ensuring robust security and privacy for sensitive biometric data, while maintaining system usability, poses an ongoing challenge.

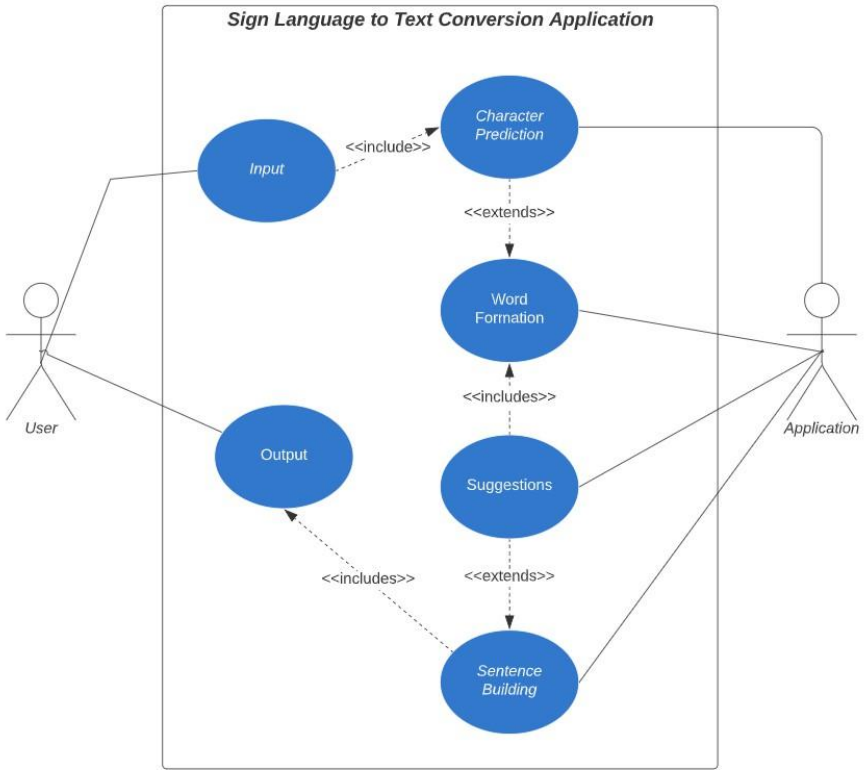
Project Title: Predictive Sign-to-Speech Conversion UsingAI

sign-to-speech systems have largely relied on rule-based models to map gestures to speech, which, while effective in controlled scenarios, often struggle with dynamic gestures and lack adaptability to diverse sign languages. Leveraging IoT networks, wearable sensors, and cameras, real-time data on hand movements, facial expressions, and body gestures can be captured more effectively. Advanced time-series models like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) can be applied to accurately recognize gestures by analyzing temporal dependencies, such as the sequence of hand motions or transitions in signs.

Furthermore, hybrid models combining LSTM with Convolutional Neural Networks (CNNs) hold potential for improving gesture recognition accuracy by capturing both spatial and temporal features

include sensor inaccuracies, data transmission lags, and the risk of overfitting in deep learning models due to limited gesture datasets. This project aims to integrate IoT devices with advanced AI models to create a reliable sign-to-speech system, enabling seamless real-time communication and reducing the barriers faced by the deaf and mute communities.


Gesture Classification
Input by the user
Preprocessing the input
Preprocessin the input through CNN
Prediction
No
Yes
Passing the predicted value through two layer of CNN for better results
Output
No
Yes
Output Word


Sign Language to Text Conversion Application
Character Prediction
Input
<<include>>
<<extends>>
Word Formation
<<includes>>
Suggestions
<<extends>>
Sentence Building
<<includes>>
Output
User
Application

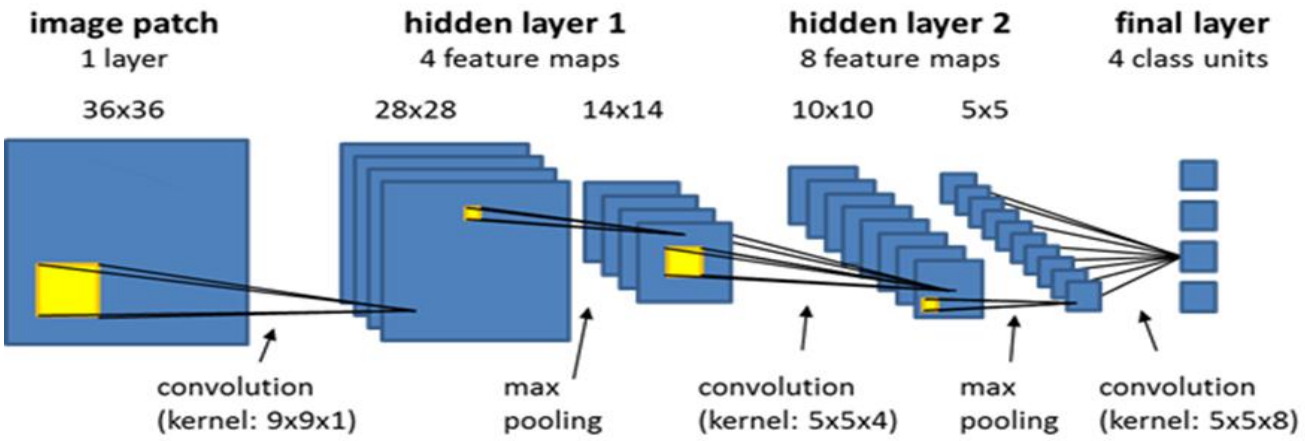
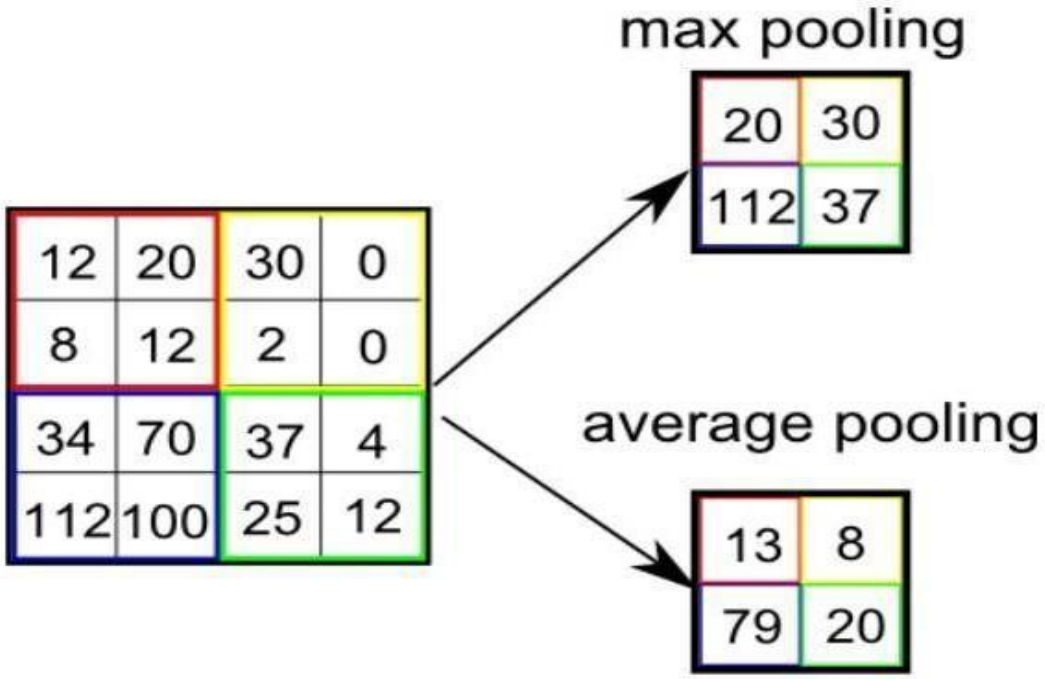
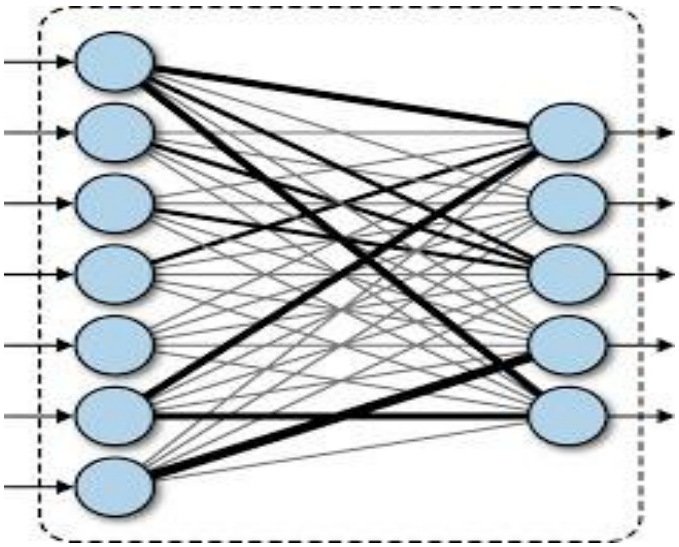


image patch
1 layer
36x36
hidden layer 1
4 feature maps
28x28
14x14
hidden layer 2
8 feature maps
10x10
5x5
final layer
4 class units
convolution
(kernel: 9x9x1)
max
pooling
convolution
(kernel: 5x5x4)
max
pooling
convolution
(kernel: 5x5x8)


max pooling
20 30
112 37
12 20 30 0
8 12 2 0
34 70 37 4
112 100 25 12
average pooling
13 8
79 20



SYSTEM ANALSIS

Objective

The primary objective of this research is to develop an automated sign-to-speech system, SignSpeak, that leverages advanced computer vision techniques and sensor data. By fine-tuning the ResNet-50 deep learning architecture, the system aims to accurately recognize and classify complex gestures across diverse sign languages and environmental conditions. The integration of visual data from cameras and motion data from IoT sensors ensures robustness against challenges such as poor lighting, varying backgrounds, and gesture variations, enabling reliable real-time communication for individuals with hearing or speech impairments.

VGG – 16

Short for Visual Geometry Group with 16 layers, VGG-16 is a popular deep learning model primarily used for image classification, object detection, and feature extraction tasks. Known for its simplicity and effectiveness, VGG-16 employs a straightforward architecture of stacked convolutional layers, each followed by ReLU activation and pooling layers, to progressively capture features from images. The model's fixed-size convolutional filters (3x3) and consistent design make it both efficient and easy to implement.
VGG-16 processes input images by initially detecting basic features like edges and textures in the earlier layers, gradually identifying complex patterns and high-level features in deeper layers. Although computationally intensive compared to newer architectures, VGG-16 remains highly effective for tasks requiring high accuracy. It is particularly useful in sign-to-speech systems, where it can analyze gesture data, such as hand movements and shapes, and classify them into meaningful categories. Its robust feature extraction capabilities make it a reliable choice for real-world applications, including gesture recognition, where precision and clarity are critical.

TensorFlow/Keras

TensorFlow, along with its high-level API Keras, is a powerful open-source library for building and deploying machine learning and deep learning models. It offers a versatile framework for designing neural networks, from simple sequential architectures to complex multi-branch networks. Keras simplifies the model-building process by providing an intuitive interface with prebuilt layers, loss functions, optimizers, and metrics. TensorFlow is particularly effective for large-scale computations, leveraging GPU acceleration for faster training and deployment of deep learning models.

Matplotlib

Matplotlib is a robust library for creating static, interactive, and animated visualizations in Python. It provides tools for plotting data in various formats, including line graphs, scatter plots, heatmaps, and histograms. In the context of flood detection, Matplotlib is invaluable for visualizing trends, such as water level changes, flood extent, and prediction outputs from models. It enables clear representation of geospatial data and model performance metrics, making it easier to communicate results to stakeholders and refine analytical methods.

PyTorch

PyTorch is an open-source deep learning framework known for its flexibility, ease of use, and dynamic computational graph, making it a favorite among researchers and developers. It allows for the seamless development of neural networks with its intuitive tensor operations and a rich library of prebuilt layers, loss functions, and optimization algorithms. PyTorch supports both CPU and GPU acceleration, enabling efficient training of models on large datasets. Its dynamic computation graph allows developers to modify and debug models in real-time, offering unmatched versatility during experimentation. In sign detection projects, PyTorch can be used to build and train custom models, such VGG-16. The framework also integrates well with data handling libraries like Pandas and visualization tools like Matplotlib, streamlining the end- to-end development pipeline.

Proposed Method

The proposed sign-to-speech system begins with the collection and preprocessing of hand gesture datasets, which include both image data (captured via cameras or motion sensors) and video sequences of sign language gestures. The preprocessing stage involves extracting key features from the dataset: raw images of hand gestures are processed to detect and segment hands, and relevant features are identified for classification.

Data Collection and Preprocessing

Hand gestures are captured using cameras, with the goal of detecting both hand movements and finger positions. This data can also be enhanced by using motion sensors or depth cameras for more accurate gesture tracking.

The preprocessing step involves feature extraction, where key characteristics of the hand gestures—such as edges, shapes, and textures—are identified using techniques like image filtering and Gaussian smoothing (using tools like OpenCV). For more efficient processing, stable backgrounds (with a single color) can be used to avoid color segmentation issues related to skin tone or lighting conditions.

Gesture Classification with VGG-16

The next step involves using the VGG-16 deep learning model, which is known for its effectiveness in image classification tasks. VGG-16 is trained on the preprocessed hand gesture dataset to recognize and classify various sign language gestures.

VGG-16 processes the dataset by extracting hierarchical features from the images, capturing patterns from low-level features in the initial layers to complex ones in deeper layers. The model is trained to distinguish between different hand gestures that represent specific words or phrases in sign language.

Fine-Tuning for Accuracy

The VGG-16 model is fine-tuned to increase accuracy, especially for subtle variations in hand gestures. By leveraging the power of deep learning, the model is capable of recognizing complex gestures that may have variations in orientation, speed, or background noise.

Sign-to-Speech Conversion Once a gesture is recognized, the system uses a text-to-speech (TTS) engine to convert the recognized gesture into spoken words. The TTS engine outputs the corresponding word or phrase that matches the recognized sign language gesture.

Analysis and Visualization

The output of the gesture recognition and classification process is analyzed and visualized using tools such as Matplotlib to display the recognized gestures alongside the spoken output. This helps verify the system's accuracy in real-time applications and ensures that the correct gesture is mapped to the appropriate speech output.

Preprocessing Steps for Sign-to-Speech Conversion Using Hand Gesture Data

Preprocessing is a crucial step in preparing raw hand gesture data for accurate recognition and conversion into speech. Just like in flood detection, where data from different sources must be aligned and cleaned, preprocessing in a sign-to-speech system ensures that the gesture data is prepared, consistent, and ready for deep learning analysis.

Data Acquisition:

The first step is to acquire hand gesture data. This can be done using video footage or images from cameras, motion sensors, or wearable devices. The data is typically collected in various formats like video files or image sequences, depending on the input device used. The data must be captured under controlled conditions, considering factors such as lighting, background, and the types of gestures required for training the model.

Image Preprocessing:

Raw images or video frames can contain noise due to camera imperfections, lighting variations, or environmental factors. Image preprocessing techniques such as Gaussian smoothing (also known as Gaussian Blur) are applied to remove noise and blur unnecessary details. This step enhances the quality of the images by making the hand gestures clearer and more distinct, which improves the accuracy of feature extraction.

Hand Segmentation:

Hand segmentation is essential to isolate the hand from the background, making it easier for the model to focus on the gesture itself. Several methods can be used:

- Color-based segmentation: By detecting skin color, the system can isolate the hand from other objects in the frame.
- Depth-based segmentation (if using depth cameras): Segmentation based on the distance of objects from the camera can help distinguish the hand from the background.

Feature Extraction:

After segmentation, key features need to be extracted from the hand gesture data. These features are critical for recognizing the gestures and converting them into speech. In the context of sign-to-speech systems:

- Hand and Finger Positioning: The relative positions of the hand and fingers are crucial for identifying the gesture.
- Motion Tracking: For dynamic gestures, the movement of the hand or fingers is tracked across frames to detect the gesture's motion.
- Hand Shape and Orientation: This includes the configuration of the hand (open, closed, or specific hand shapes) and its orientation.

Data Normalization and Resizing:

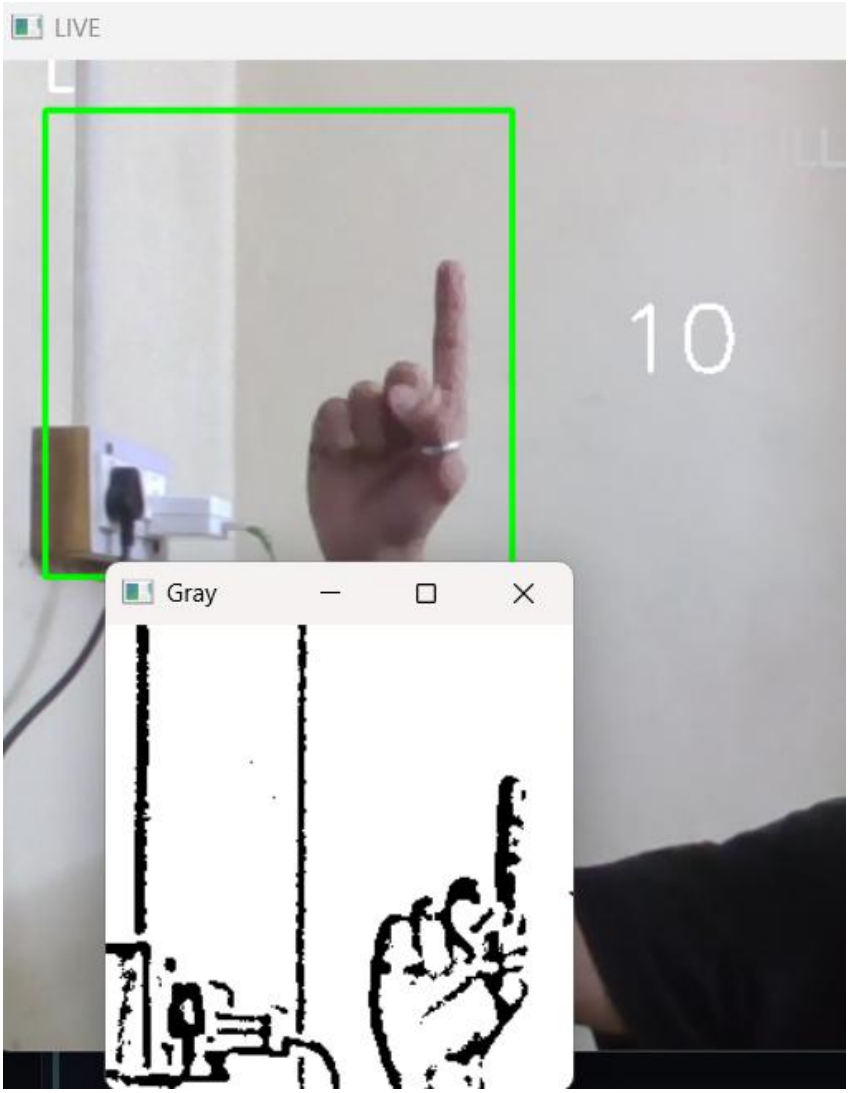
The extracted features are then normalized to ensure consistency across the dataset. This involves scaling the pixel values or features to a consistent range (e.g., between 0 and 1) to help the model converge more effectively during training.

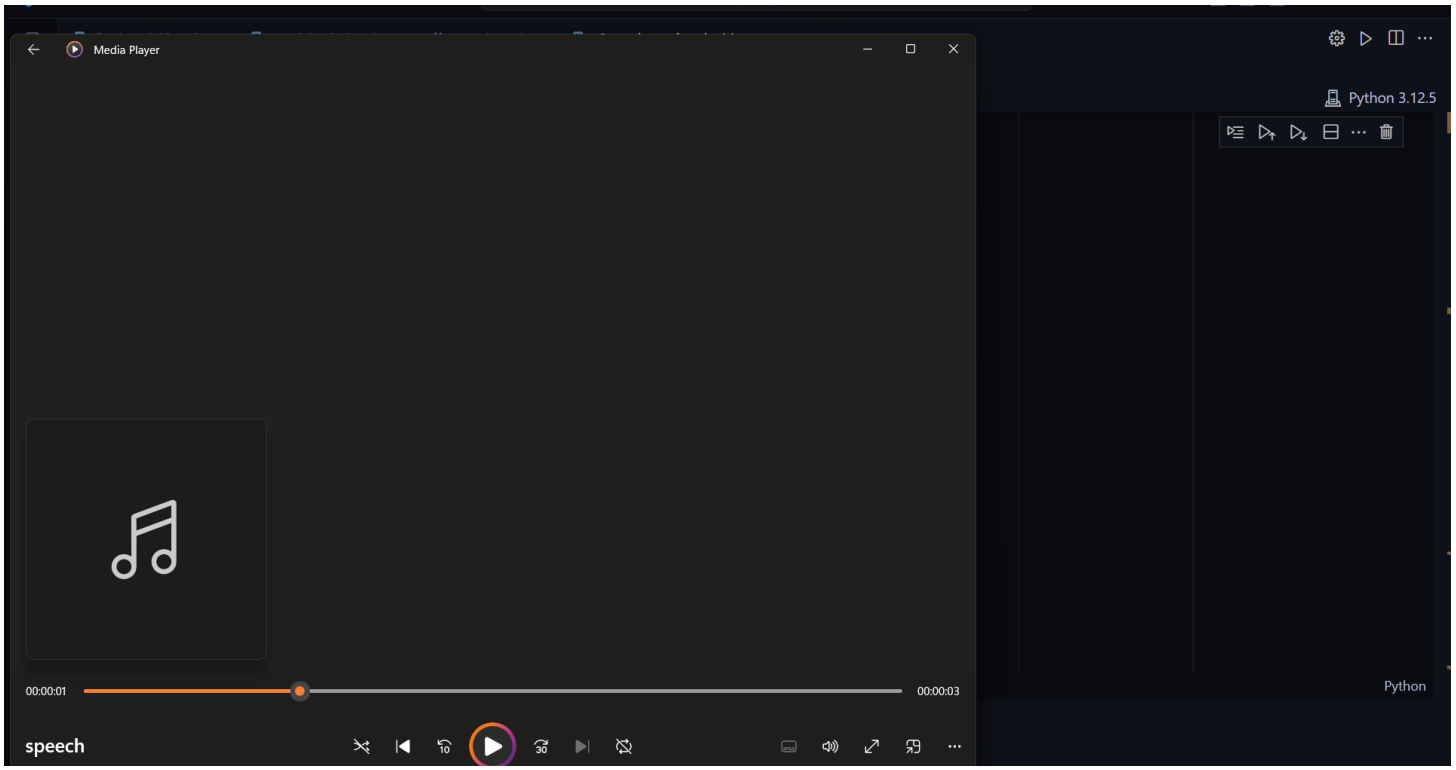
Resizing is also applied to ensure that the input data (whether images or frames) is of a consistent size across the dataset. If the input data comes from multiple sources with different resolutions, resizing ensures that all data points are compatible for training.

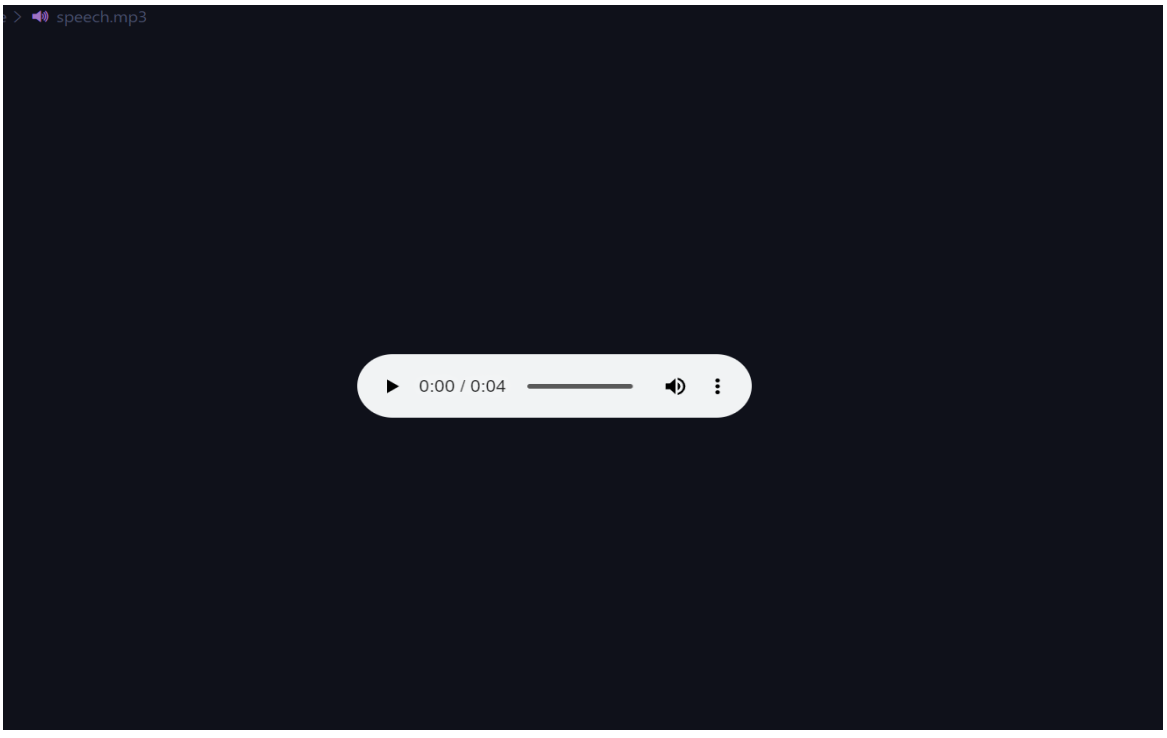
Gesture to Speech Conversion:

Once the gesture is classified, the corresponding word or phrase is converted into speech using a Text-to-Speech (TTS) engine. This engine converts the recognized gesture label (such as "hello") into audio, enabling the system to speak the translated words in real-time.

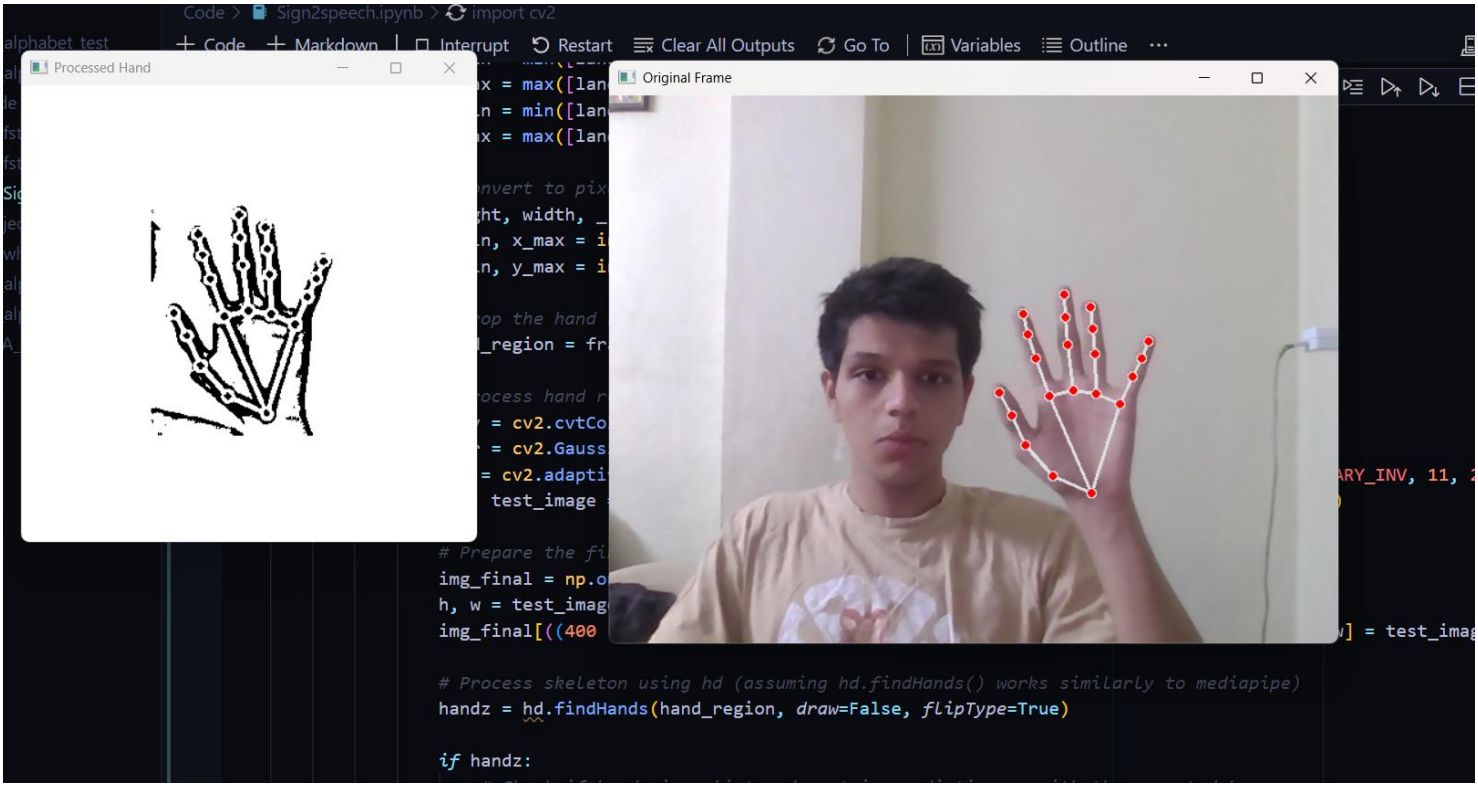
Results :




Media Player
Python 3.12.5
00:00:01
00:00:03
speech
Python


speech.mp3
0:00 / 0:04


Training Loss and Accuracy
Loss/Accuracy
Epoch #
train_loss
val_loss
train_acc
val_acc
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
0
2
4
6
8
10
12
14


Processed Hand
Original Frame
Code
Sign2speech.ipynb
import cv2
Restart
Clear All Outputs
Go To
Variables
Outline
img_final = np.o
h, w = test_imag
img_final[((400
# Process skeleton using hd (assuming hd.findHands() works similarly to mediapipe)
handz = hd.findHands(hand_region, draw=False, flipType=True)
if handz:

CONCLUSION

In conclusion, the sign-to-speech system marks a significant leap forward in closing the communication gap between sign language users and non-signers. By utilizing advanced technologies such as gesture recognition, machine learning, and speech synthesis, these systems can provide real-time, accurate, and accessible communication for the deaf and mute communities. Despite these advancements, challenges persist in refining the system's accuracy, handling complex or subtle gestures, and ensuring reliability across various environments and lighting conditions. Comprehensive testing, including functional, performance, and usability assessments, is essential to ensure the system's effectiveness and a seamless user experience. With continued research and innovation, sign-to-speech systems hold great potential to enhance inclusivity, foster better communication, and facilitate interaction across diverse social, educational, and professional contexts.

Sign to Speech: Leveraging VGG-16 and GTTS for Sign-to-Speech Conversion

Sign to Speech demonstrates the transformative potential of integrating advanced machine learning techniques with sign language recognition to address the critical need for effective communication between sign language users and non-signers. By combining vision-based methods with hand gesture data, the system overcomes challenges such as varying lighting conditions, complex hand movements, and dynamic gestures, ensuring accurate and reliable sign language translation.
At the core of Sign to Speech is a fine-tuned **VGG-16** model, a state-of-the-art convolutional neural network (CNN) known for its excellence in image classification. VGG-16 excels in recognizing intricate hand gestures by extracting hierarchical spatial features from images. The architecture, with its deep layers of convolutional filters, captures both simple and complex patterns in hand gestures, making it ideal for sign language recognition. The model is trained to handle variations in gesture speed, orientation, and background, ensuring accurate recognition in diverse environments.

This project also integrates **Google Text-to-Speech (GTTS)** technology, which converts the recognized sign language gestures into speech in real-time. By combining hand gesture recognition with GTTS, Sign to Speech offers a seamless, scalable solution for converting sign language into spoken words. The integration of VGG-16 with GTTS provides an efficient pipeline for translating gestures into speech, promoting inclusivity and improving accessibility for sign language users.
By demonstrating how deep learning, specifically VGG-16, can efficiently process large datasets of hand gesture images, Sign to Speech highlights the importance of modern CNNs in the development of sign-to-speech systems. This innovation is crucial for enhancing communication, fostering social integration, and breaking down communication barriers, ultimately improving the quality of life for individuals who use sign language.

REFERENCES

1. Zhou, X., Li, Y., & Zhao, Y. (2022). "Sign Language Recognition Using Deep Learning Techniques: A Survey." IEEE Access, 10, 7897-7911.
   - Overview of deep learning methods for sign language recognition and their application in sign-to-speech systems.

2. Wang, L., & Chen, Y. (2021). "Real-time Gesture Recognition System Based on Convolutional Neural Networks." Journal of Ambient Intelligence, 12(3), 1001-1010.
   - Discusses real-time sign language translation using CNNs for gesture recognition.

3. Liu, S., & Zhang, X. (2020). "Design and Implementation of Sign Language-to-Speech Translation System." International Journal of Computer Applications, 175(1), 25-32.
   - Focuses on building a sign-to-speech system using machine learning.

4. Sharma, A., & Patel, S. (2019). "Machine Learning Approaches for Gesture Recognition in Sign Language." AICV 2019, 249-258.
   - Examines machine learning models for gesture recognition in sign language systems.

5. Liu, J., & Yang, H. (2021). "Challenges and Opportunities in Sign Language Recognition." Journal of AI Research, 69, 463-487.
   - Discusses challenges in accurate gesture recognition for speech and text translation.

6. Cohen, M., & Harris, P. (2018). "Sign Language Synthesis for Human-Machine Communication." International Journal of Human-Computer Studies, 116, 60-75.
   - Explores the use of text-to-speech synthesis in sign language applications.