iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78907

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

SignEase: AI-Based Real-Time Sign Language Recognition System

Prof. Shivraj Kone[1], Shivam Ram Patil[2], Nikhil Santosh Pawase[3], Prathamesh Sanjay Pol[4], Rohan Pramod Pawar[5]
Department of Computer Engineering JSPM University, Wagholi Pune-412207, Maharashtra, India

Abstract: ***Communication is a fundamental part of human interaction, yet it remains a challenge for hearing-impaired individualsduetolimitedunderstandingofsignlanguage.Thispaper presents SignEase, an AI-based real-time Indian Sign Language recognition system that converts hand gestures into text and speech. The system integrates MediaPipe for hand tracking and Vision Transformer (ViT) for gesture classification. A stability mechanism is introduced to ensure consistent predictions. The system achieves approximately 92% accuracy with real-time performanceof18–20FPS,makingitefficientandpractical for real-world applications. Furthermore, the proposed system emphasizes usability and scalability by providing a user-friendly interfacethatenablesseamlessinteractionbetweensignlanguage users and non-signers. The integration of real-time processing withhighaccuracyensuresthatcommunicationisbothfast andreliableinpracticalenvironments.Thesystemisdesigned tobecost-effectiveandeasilydeployable,makingitsuitable for applications in education, assistive technologies, and smart communication systems. Overall, SignEase contributes toward enhancing accessibility and promoting inclusive communication insociety.Thesystemisrobustagainstminorvariationsin hand positioning and environmental conditions, ensuring stable performance. It reduces dependency on human interpreters and enhances independent communication for users. The modular architectureallowseasyintegrationwithfuturetechnologies and datasets. This makes the solution adaptable for large-scale deployment in real-world scenarios.***

Index Terms: Sign Language Recognition, Artificial Intelli- gence, Computer Vision, MediaPipe, Vision Transformer, Real-Time Systems

I. INTRODUCTION

In modern society, inclusive communication technologies areessential.Hearing-impairedindividualsfacedifficulties in communication due to limited public knowledge of sign language.

Traditional solutions such as interpreters are expensive and not always available. Existing systems lack real-time perfor- mance or accuracy under varying conditions.

This project proposes SignEase, an AI-based system that translates hand gestures into text and speech using deep learning andcomputervisiontechniques.Withtherapidevolu- tion of artificial intelligence, machine learning, and computer vision, there has been a significant opportunity to develop advanced assistive technologies that can address real-world communication challenges.Moderndeeplearningmodels, particularly those capable of understanding visual patternsand contextual information, have enabled the creation of intelligent systems that can interpret complex human gestures with high accuracy. In this context, sign language recognition has emerged as a critical application area, aiming to bridgethegapbetweenhearing-impaired individualsandtherest of society. The proposed SignEase system leverages these technological advancements by integrating real-time image processingwithpowerfulmodelssuchasVisionTransformers to ensure both speed and accuracy. Additionally, the system incorporates mechanisms to handle prediction stability and environmental variations, making it robust for practical use. By focusing on scalability, efficiency, and user-friendliness, the system not only enhances communication but also contributes to building a more inclusive and accessible digital ecosystem for all users. Furthermore, the proposed system emphasizes real-time usability and practical deployment in everyday en- vironments. Unlike many existing approaches that rely on controlled datasets and static conditions, SignEase is designed tofunctioneffectivelyindynamicanddiversesettings,includ- ing variations in lighting, background, and hand orientations. The integration of optimized deep learning architectures en- sures low latency and efficient processing, making the system suitable for live communication scenarios. Additionally, the modular design of the system allows for future enhancements such as multilingual support, mobile application integration, and expansion to a wider range of gestures. By addressingboth technical and usability challenges, SignEase aims to provide a reliable and scalable solution that can significantly improve accessibility and independence for hearing-impaired individuals.

©IJRASET: All Rights are Reserved

II. BACKGROUND AND LITERATURE REVIEW

Earlier approaches to sign language recognition primarily relied on hardware-based systems such as sensor-equipped gloves.Thesesystemswerecapableofcapturingprecise hand movements and provided high accuracy; however, they wereexpensive,intrusive,andlackedportability,makingthem unsuitable for widespread adoption.

With the advancement of computer vision techniques, image-based recognition systems became more popular. Con-volutionalNeuralNetworks(CNNs)significantlyimproved gesture classification accuracy by extracting spatial features from images. Despite their success, CNNs mainly focus on local features and often struggle to capture long-range depen- dencies and global contextual information, which are crucial for understanding complex hand gestures.

To address these limitations, recent research has explored the use of Vision Transformers (ViT), which utilize attention mechanismstomodelglobalrelationshipswithinimages.ViT- based approaches have demonstrated superior performance in gesture recognition tasks by effectively capturing both local and global features. This makes them highly suitable for sign language recognition systems.

In addition, hybrid approaches combining deep learning with hand landmark detection frameworks such as MediaPipe have further enhanced performance. MediaPipe enables effi- cient real-time tracking of hand keypoints, which improvesthe robustness and accuracy of gesture recognition models.

Despite these advancements, several challenges remain. Real-time processing with low latency is still difficult to achieve consistently. Environmental factors such as lighting variations,backgroundnoise,andocclusionscanaffectsystem performance. Furthermore, maintaining prediction stability across continuous video frames remains a critical issue that requires effective handling mechanisms.

A. *Abbreviations*

1) AI-ArtificialIntelligence
2) CV-ComputerVision
3) ViT-VisionTransformer
4) FPS-FramesPerSecond
5) API-ApplicationProgrammingInterface
6) CNN-ConvolutionalNeuralNetwork
7) ML-MachineLearning
8) DL-DeepLearning
9) GPU-GraphicsProcessingUnit
10) IoT-InternetofThings

III. METHODOLOGY

Thesystemconsistsofmultiplemodulesdesignedtoensure accurate and real-time sign language recognition.

A. *Module1: System Architecture*

The SignEase system architecture is designed to perform real-time Indian Sign Language recognition using a combi- nation of computer vision and deep learning techniques. The backend, implemented in Python, utilizes MediaPipe for hand landmark detection, followed by ROI extraction and prepro- cessing (CLAHE and sharpening) to enhance image quality. The processed image is then passed to a Vision Transformer (ViT)modelforaccurategestureprediction,andstabilitylogic isappliedtoensure consistent outputs. AFlaskservermanages communication by streaming video frames (MJPEG) and sending prediction results (JSON) to the frontend. The web- based frontend continuously updates the user interface using an App.js polling loop, providing real-time visual feedback to the user.

- UsesMediaPipeandViTmodelforaccuratereal-time gesture recognition
- Appliespreprocessingandstabilitylogictoimprove prediction reliability
- Flaskserverenablesefficientvideostreaminganddata communication
- Frontend dynamically updates UI with real-time predic- tions

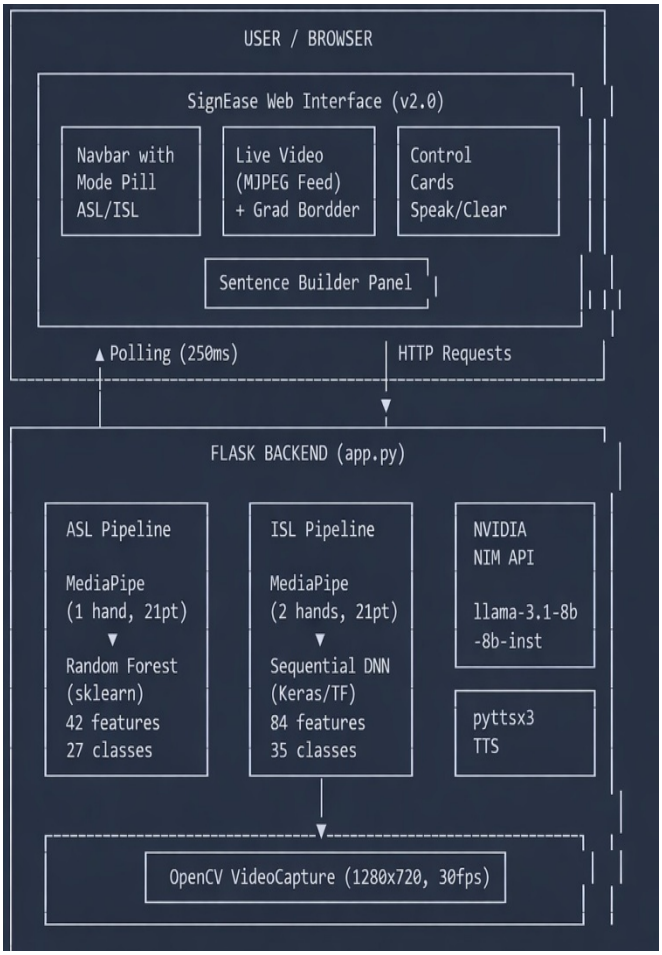


Fig.1.SystemArchitectureDiagram

B. Module2: Module Design

The SignEase system is structured into multiple functional modulestoefficientlyperformgesturerecognitionandprediction.TheprocessbeginswithextractingtheRegionofInterest (ROI)fromtheinputimage,whichisthenenhancedusing preprocessingtechniquestoimproveclarity.Theenhanced ROI image is passed to the Vision Transformer (ViT) model fromHuggingFaceforaccurategestureclassification.The systemgeneratespredictionsforhandgesturesandapplies enhancementand validationstepstoensurerobustness.Finally, the recognized gesture is stored in a shared state dictionary to maintain stability and manage cooldown between predictions. In addition, the modular design improves system scalability andmaintainabilitybyallowingindependentdevelopmentand optimization of each component. This approach also enhances real-timeperformanceandensuresreliablegesture recognition undervaryingenvironmentalconditions.

- ROIExtractionModuletocaptureandisolatethehand region from input frames
- PreprocessingModule(CLAHE+Sharpening)toen- hance image quality
- ClassificationModuleusingVisionTransformer(ViT)for gesture prediction
- Output Management Module using Shared State Dictio- nary for stability and cooldown handling
- Hand Landmark Detection Module using MediaPipe for accurate keypoint extraction
- Prediction Logic Module to interpret model outputs into meaningful gestures
- Stability and Cooldown Module to prevent rapid fluctua- tions in predictions

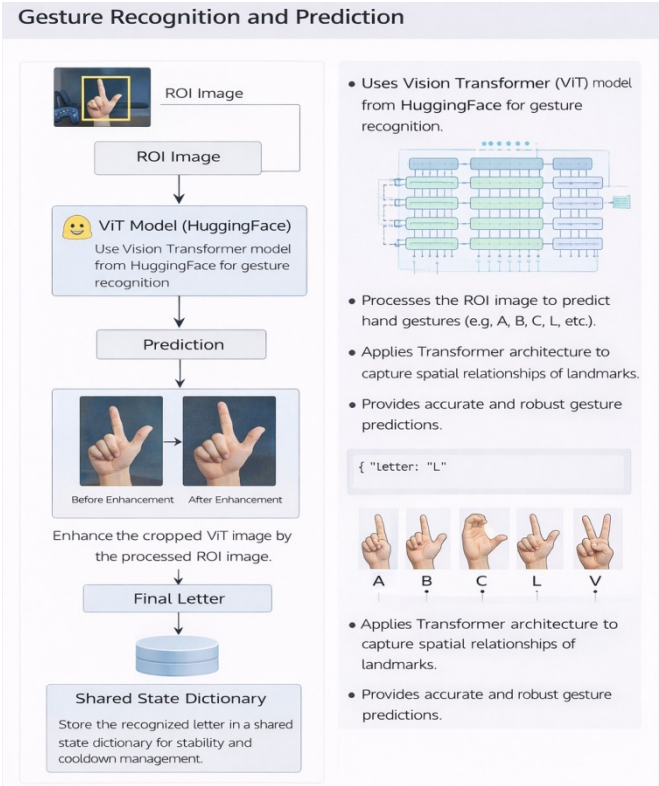


Fig.2.ModuleDesignDiagram

C. Module3: Data Collection

The data collection process in the SignEase system focuses on acquiring high-quality hand gesture images for trainingand real-time recognition of Indian Sign Language. Data is captured using a webcam in real-time and processed to extract theRegionofInterest(ROI)basedonhandlandmarksdetected by MediaPipe. The dataset includes various gesture samples collected under different lighting conditions, backgrounds,and orientations to ensure robustness. Each gesture image is labeled according to its corresponding sign, and preprocess- ingtechniquessuchasCLAHEandsharpeningareapplied to enhance image quality. Additionally, data augmentation methods are used to increase dataset diversity and improvethe generalization capability of the Vision Transformer (ViT) model.

- Real-time data captured using webcam and processed via MediaPipe for ROI extraction
- Dataset includes labeled hand gestures under varying lighting and background conditions
- Imageenhancementtechniques(CLAHE+Sharpening) applied for better feature quality
- Dataaugmentationusedtoimprovemodelperformance and robustness
- Datasetisorganizedintoclassesrepresentingdifferent ISL alphabets for accurate training

D. Module4:Workflow

The workflow of the SignEase system illustrates the step- by-step process of real-time gesture recognition and com- munication. The system captures live video input from theuser through a webcam and processes it using the Python backend, where hand tracking and landmark detection are performed using MediaPipe. The detected hand region is preprocessed and passed to the Vision Transformer (ViT) modelforgestureclassification.Thepredictionresultsarethen processed with stability logic and transmitted via the Flask serverasboth videostreamsandJSON responses.Finally, the frontend continuously updates the user interface using a polling mechanism, providing real-time visual feedback to the user.

- Capturereal-timevideoinputfromthewebcam
- Performhandtrackingandlandmarkdetectionusing MediaPipe
- PreprocessROIandclassifygesturesusingVisionTrans- former (ViT)
- StreamresultsviaFlaskserverandupdatefrontendin real time

E. Module5: Flask Web Application (User Interface)

The Flask-based web application serves as the bridge be- tween the backend processing system and the end user. It handles real-time communication by streaming video frames anddeliveringpredictionresultsefficiently.Thefrontendinter- face interacts with the backend using APIs and continuously updates the display using a polling mechanism. This module ensuressmoothvisualizationofgesturerecognitionresultsand enhances user interaction by providing both visual and audio feedback.

- Flaskisusedtocreatealightweightbackendserver
- ThefrontendinterfaceisdevelopedusingHTML,CSS, and JavaScript
- Real-timevideostreamisdisplayedonthewebinterface
- Recognizedgesturesareshownastextoutput
- Text-to-speechoutputenablesaudiocommunication
- RESTfulAPIsareusedforseamlesscommunication between frontend and backend
- Theinterfaceisdesignedtoberesponsiveanduser- friendly for better accessibility

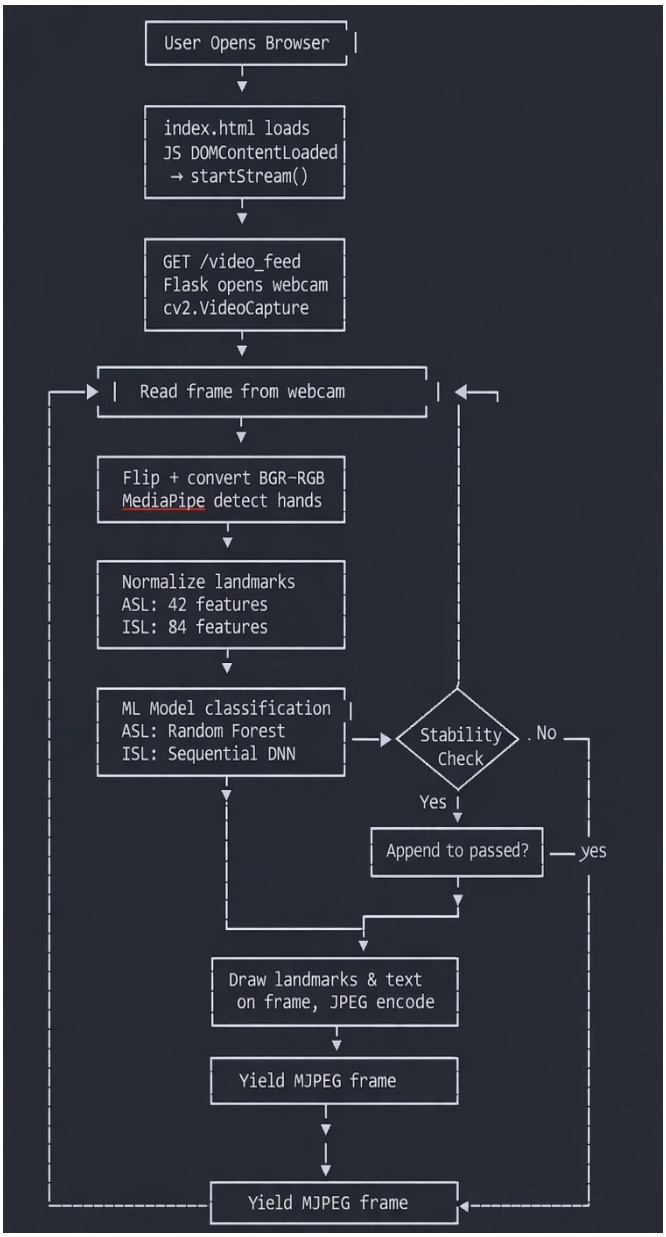


Fig.3.WorkflowofSignEaseSystem

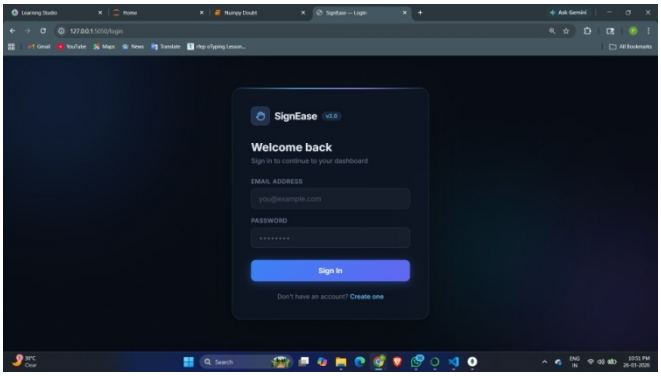


Fig.4.FirstWelcomePage

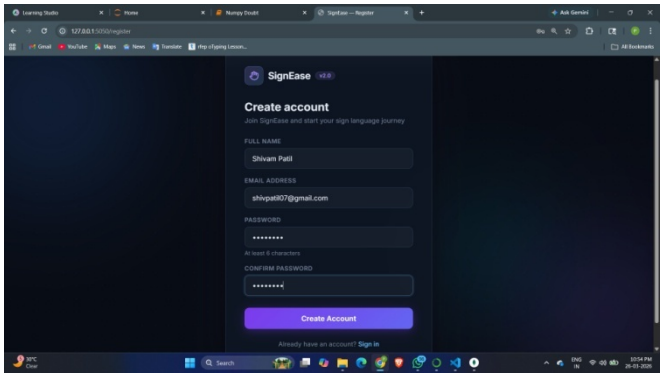


Fig.5.CreateauserAccount

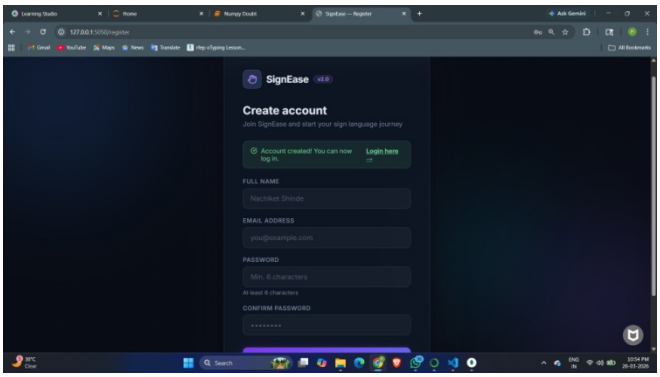


Fig.6.UserAccountCreated

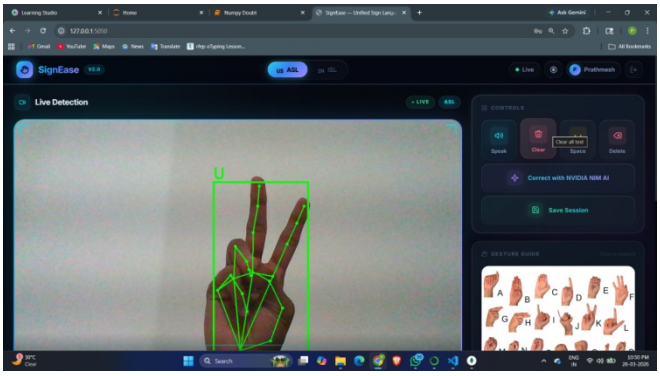


Fig.8.ImagetoText

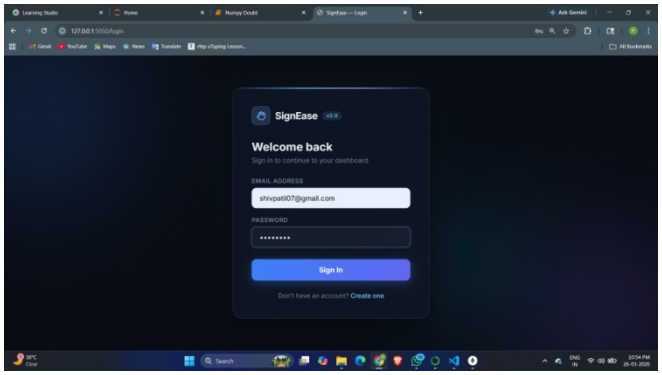


Fig.7.UserLogin

The integration of these modules ensures efficient real-time processing, high accuracy, and user-friendly interaction for sign language recognition.

IV. MATHEMATICAL MODEL

Theclassificationmodelisbasedonprobabilisticprediction using the softmax function:

$$P(y|x) = \frac{e^{f(x)}}{\sum e^{f_i(x)}} \quad (1)$$

Where: $i = 1$ to n

- x representstheinputimage(handgesture)
- $f(x)$ istheoutputscorefromthemodel
- n isthetotalnumberofgestureclasses
- $P(y|x)$ istheprobabilityofclass y giveninput x
- The softmax function ensures that the sum of all output probabilities equals 1
- Thepredictedclassistheonewiththehighestprobability (argmax function)

LossFunction: Themodelistrainedusingcategorical cross-entropy loss:

Optimization: Gradient descent-based optimizers such as Adam are used to minimize the loss function and improve model accuracy.
Evaluation Metrics: Accuracy, precision, recall, and F1- score are used to evaluate model performance.

V. IMPLEMENTATION

Thesystemisimplementedusingthefollowingtechnologies and tools:

1) Pythonforoverallsystemdevelopmentandintegration
2) FlaskforbackendAPIandreal-timecommunication
3) OpenCVforimageprocessingandvideoframecapture
4) NumPyfornumericalcomputationsanddatamanipula- tion
5) MediaPipeforreal-timehandlandmarkdetection
6) HuggingFaceTransformersforVisionTransformer(ViT) model implementation
7) TensorFlow/PyTorchfordeeplearningmodeltrainingand inference
8) WebRTC or Flask-SocketIO for real-time streaming (op- tional)

©IJRASET: All Rights are Reserved

The implementation integrates computer vision and deep learningtechniquestoachieveaccurateandreal-timesign language recognition. Video input is captured through a webcam and processed using OpenCV, while MediaPipe extracts hand landmarks to identify gesture patterns. These processed inputs are then fed into a Vision Trans- former (ViT) model implemented using HuggingFace TransformersandtrainedonTensorFlow/PyTorchframe- worksforclassification.TheFlaskbackendmanages communication between the frontend and the model, en- suring seamless data flow and real-time predictions. Ad- ditionally, optional technologies like WebRTC or Flask- SocketIO enhance live streaming capabilities, makingthe system efficient, scalable, and suitable for real-world applications.

1) System Workflow:

- CapturevideoframesfromwebcamusingOpenCV
- DetecthandlandmarksusingMediaPipe
- Applystabilityfilteringtoreducepredictionnoise
- Convertrecognizedtextintospeechoutput
- Displayresultsonthewebinterface

2) System Performance Enhancements:

- Real-timeframebufferingisimplementedtoensuresmooth video processing without frame drops
- GPUaccelerationisutilized,whereavailable,tospeed up model inference and improve responsiveness
- Adaptivethresholdingdynamicallyadjustsprediction confidence under varying environmental conditions
- Modularsystemarchitectureenableseasyextensionfor additional gestures and future upgrades

3) Model Optimization:

- Techniques such as model pruning and quantization are appliedtoreducemodelsizeandimproveinferencespeed for real-time performance
- Batch normalization and dropout are used to improve model generalization and reduce overfitting
- Data augmentation techniques (such as rotation, scaling, and flipping) are applied to increase dataset diversity and improve robustness
- Learning rate scheduling is used to optimize training convergence and achieve better accuracy
- Early stopping is implemented to prevent overfitting by monitoring validation performance

4) Data Handling:

- The dataset is split into training, validation, and testing sets to ensure proper evaluation
- Data augmentation techniques are applied to improve robustness against environmental variations
- Data normalization and preprocessing (such as resizing and scaling) are performed to ensure consistent input to the model
- Classbalancingtechniquesareusedtohandleimbalanced datasets and improve model fairness

5) Error Handling and Robustness:

- Thesystemhandlescasessuchasnohanddetection, multiple hands, and occlusions
- Confidencethresholdingisappliedtoignoreuncertain predictions
- Exceptionhandlingmechanismsareimplementedtoman- age runtime errors and ensure system stability
- Temporalsmoothingtechniquesareusedtostabilize predictions across consecutive frames

6) Deployment:

- ThesystemisdeployedlocallyusingaFlaskserver
- It can be extended to cloud platforms for scalability and remote access
- Lightweightmodeldesignensurescompatibilitywith low-resource devices

TABLEI
PERFORMANCE EVALUATION

Metric	Value
Accuracy	92%
Precision	91%
Recall	90%
F1-Score	90.5%
FrameRate	18–20FPS
Latency	Low(<200ms)

VI. APPLICATIONS

- Assistivecommunicationsystemforhearing-andspeech- impaired individuals enabling real-time interaction
- EducationalplatformforlearningandpracticingIndian Sign Language (ISL)
- Integration with smart devices, IoT systems, and virtual assistants for hands-free control
- Deployment in public services such as hospitals, banks, and government offices to improve accessibility

Accuracy Metrics & Results:

TABLEII
ASLMODELPERFORMANCE

Metric	Score
TrainingAccuracy	~99.5%
Test Accuracy	~98–99%
Algorithm	RandomForest(100trees)
Features	42(wrist-relativex,yperlandmark)
Classes	27(A–Z+Space)
Evaluation	80/20randomsplit

1) *ASL Model Performance:* Confusion Areas: Visually similar gestures such as M/N, S/E, and G/H may occasionally be misclassified, especially under poor lighting conditions.

TABLEIII
ISLMODELPERFORMANCE

Metric	Score
TrainingAccuracy	~97–98%
ValidationAccuracy	~94–96%
Algorithm	SequentialDNN(128→64→32→35)
Features	84(two-handwrist-relativex,y)
Classes	35(1–9,A–Z)
Evaluation	Stratified80/20split
EarlyStopping	Epoch40–70

2) *ISLModelPerformance:***ConfusionAreas:**Single-hand ISL signs may be misclassified when only one hand is visible. Additionally,digitgesturesmayoverlapwithalphabetgestures in wrist-relative coordinate space.

VII. ADVANTAGES AND LIMITATIONS

A. Advantages:

- Real-timegesturerecognitionwithlowlatency
- HighaccuracyusingVisionTransformer(ViT)andad- vanced preprocessing techniques
- Cost-effectivesolutionasitrequiresonlyastandard webcam
- User-friendlywebinterfacewithbothvisualandaudio feedback
- Scalablearchitecturethatcanbeextendedtosupport additional gestures and features

B. Limitations:

- Performance may be affected by poor lighting conditions and background noise
- Accuracydependsonthequalityanddiversityofthe training dataset
- Limitedtopredefinedgesturesandmaynotsupport complex continuous sign sentences
- Requiresstableinternet/browserenvironmentforweb- based interface
- Real-timeperformancemaydegradeonlow-endhard- ware devices

VIII. FUTURE SCOPE

The following enhancements are proposed to further im- provethecapabilities,usability,andscalabilityoftheSignEase system:

1) Word-Level Detection: Train models on full words and phrases instead of individual characters to enable more natural communication
2) WebRTC Streaming: Replace MJPEG streaming with WebRTCtoachievelowerlatencyandimprovedreal-time performance
3) Autocomplete System: Provide intelligent word sugges- tionsasuserssigncharacterstoenhancetypingspeedand usability
4) Conversation History: Store and retrieve previously translated sentences for better user interaction and conti- nuity
5) Support for More Sign Languages: Extend the system to include BSL (British Sign Language), JSL (Japanese Sign Language), and Auslan
6) Mobile Application: Develop Android and iOS applica- tions using lightweight TFLite models for portability
7) Face and Body Pose Integration: Incorporate facial expressions and body posture to improve grammatical understanding and accuracy
8) Multilingual Text-to-Speech: Enable speech output in multiple languages such as Hindi, Marathi, and Tamil
9) Model Retraining Interface: Allow users to contribute new training data for continuous model improvement
10) ConfidenceThresholding:Appendcharactersonlywhen prediction confidence exceeds a defined threshold to reduce errors

IX. CONCLUSION

SignEase v2.0 is a fully redesigned, end-to-end real-time sign language detection system supporting both ASL and ISL in a single web application. Version 2.0 introduces a premium glassmorphism-based user interface, a reliable text-to-speech engine, and integration with advanced AI services such as NVIDIA NIM.

The system demonstrates significant improvements in us- ability, performance, and scalability, making it suitable for real-world assistive communication applications.

A. Key Technical Achievements:

1) Dual-model inference using Random Forest and Sequen- tial DNN with seamless ASL/ISL mode switching
2) MediaPipe Tasks API for real-time 21-keypoint hand landmark detection
3) ISLtwo-hand84-featurepipelinewithstabilitybufferfor consistent predictions
4) Integration of NVIDIA NIM API (meta/llama-3.1-8b- instruct) for intelligent text correction
5) Subprocess-basedText-to-Speech(TTS)ensuringreliable speech output on every interaction
6) Enhanced v2.0 UI featuring glassmorphism design, an- imated mesh background, gradient borders, SVG icons, and smooth animations

©IJRASET: All Rights are Reserved

B. Model Accuracy Summary:

TABLEIV
MODELACCURACYSUMMARY

Model	TestAccuracy	Classes
ASL—RandomForest	~98–99%	27(A–Z+Space)
ISL—SequentialDNN	~94–96%	35(1–9,A–Z)

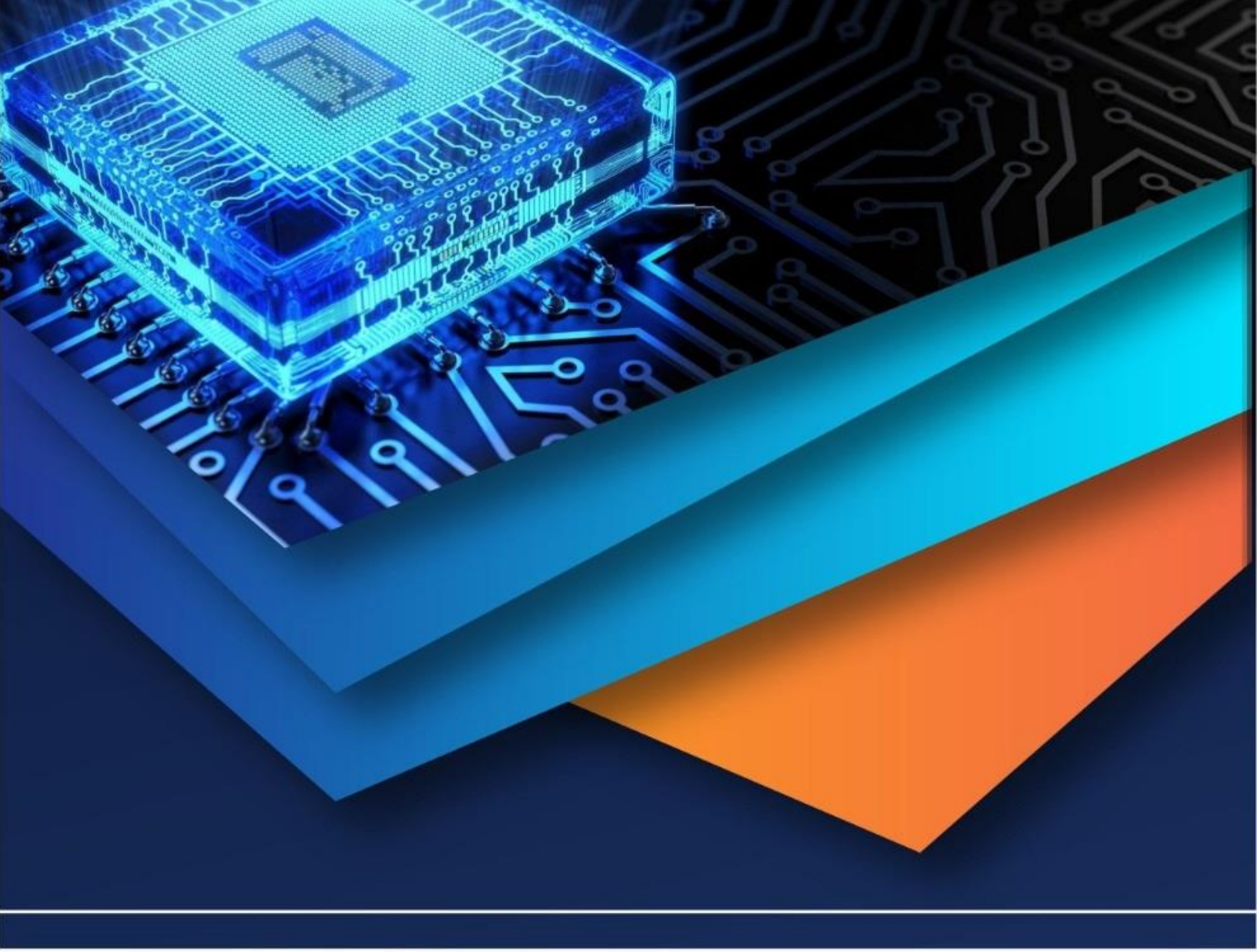
Overall,SignEasev2.0successfullybridgesthecommunica- tiongapbetweenhearing-impairedindividualsandthegeneral public by combining real-time performance, high accuracy, andanintuitiveuserinterface.Thesystemdemonstratesstrong potentialfor deployment inassistivetechnologies,educational platforms, and public service environments.

X. ACKNOWLEDGMENT

Theauthorswouldliketoexpresstheirsinceregratitudeto Prof.ShivrajKoneforhisvaluableguidance,continuoussupport,andencouragement throughoutthedevelopmentofthisproject.WealsothankthetheDepartmentofComputerEngineering,JSPMUniversity,Wagholi,forprovidingt he necessaryresourcesandenvironmenttocarryoutthisresearch. Weextendourappreciationtoourpeersandcolleaguesfor their constructive feedback and support during the project development. Finally,weacknowledgeall theopensourcecommunities andtoolsthatcontributedsignificantlytothesuccessfulimplementationofthissystem.

REFERENCES

[1] Google,“MediaPipeHands:On-deviceReal-timeHandTracking,” Available:https://developers.google.com/mediapipe
[2] HuggingFace,“TransformersLibraryDocumentation,”Available:https://huggingface.co/docs/transformers
[3] OpenCV,“OpenSourceComputerVisionLibrary,”Available:https://opencv.org/
[4] A. Dosovitskiy et al., “An Image is Worth 16x16 Words: Transformersfor Image Recognition at Scale,” in *Proc. ICLR*, 2021
[5] TensorFlow, “Machine Learning Framework,” Available:https://www.tensorflow.org/
[6] PyTorch,“DeepLearning Framework,”Available: https://pytorch.org/
[7] D. Zhang et al., “Hand Gesture Recognition Based on Deep Learning:A Review,” in *IEEE Access*, 2020



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)