iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IV Month of publication: April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69696

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Sign Language Detection with Learning

Rachana More[1], Harsh Yadav[2], Deepali Shrikhande[3]
Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, India

Abstract: "Sign Language Detection with Learning" is a project designed to create the possibility of real-time conversation for both hearing-impaired signers and the general audience by detecting ASL hand gestures with the ability to convert them into text and spoken words. The web app's deep-learning models (EfficientNetV2, MobileNetV2) are trained in a Kaggle-provided ASL dataset that has been optimized from TensorFlow to TensorFlow Lite so it can be deployed as quickly as and be as lightweight as possible. The web app is designed to use live webcam video as input and perform frame-by-frame processing by OpenCV to detect what the user did and to display both the same alphabet or word and the confidence value that OpenCV comes up with. In addition to real-time detection, the project features an interactive learning module which allows an English-to-ASL converter with backend performance visualization, scoring by character, and suggestions for better clarity in making the gestures. We have also included a quiz module to allow students to re-enforce what they are learning, which makes the application a communications and education platform. The application itself is programed in Python using Flask as the backend, HTML, CSS, and JavaScript for the frontend, and the Web Speech API to turn the recognized text into speech with different voices and languages to choose from. Users can also modify detection threshold, consecutive acceptance rate, and speech speed settings as per their requirement to personalize their experience. Overall, the project integrates machine learning, accessibility, and education on one platform, providing a user-friendly solution to enable inclusive learning and communication.
***Keywords**: Sign Language Detection, American Sign Language (ASL), Deep Learning, MobileNetV2, Real-time Gesture Recognition, TensorFlow Lite, Accessibility Technology, Interactive Learning Module, English-to-ASL Converter, Assistive Communication.*

I. INTRODUCTION

Communication is a large aspect of our everyday lives-it enables us to connect with others, learn, develop careers, and keep relationships. For individuals who are hearing or speech impaired (deaf), sign language becomes their major method of communication. Of the numerous varieties of sign languages used worldwide, American Sign Language (ASL) is one of the most recognized and used forms. But when there is an attempt at communication from sign language speakers to non-sign language speakers, it becomes a daunting barrier. The resulting communication barrier then creates an experience of alienation, misperception, and diminished access to much of everyday life.

With the technological advancements of machine learning, artificial intelligence, and computer vision, it is a wide possibility to develop systems capable of recognizing sign language gestures and translating them into readable or audio forms in real time. The project "Sign Language Detection with Learning" strives to provide such a capability as a web-based system. The system, apart from identifying static ASL hand signs using a webcam and deep learning models, translates the signs into speech and text and bridges sign language users and non-signers into communication.

The project leverages light and efficient models (EfficientNetV2 and MobileNetV2) that are trained on a large dataset of ASL data downloaded from Kaggle. The models are wrapped in an easy-to-use web interface using TensorFlow, TensorFlow Lite, Flask, and OpenCV. JavaScript and Web Speech API enhance the functionality of the platform by providing customizable voice output in multiple languages. Detection sensitivity, speech rate, and other features can be customized by users through a settings dashboard, thus making the platform versatile enough to accommodate diverse applications. Besides real-time recognition, the system also includes a learning aspect in encouraging ASL awareness and learning. It features an English-to-ASL translator, performance tracking per alphabet by user, sign clarity prompts, and a quiz mode presented as an interactive game. Using this, the site can be useful, besides its communication value, as well as an education site for novices and even hobbyists.

The current implementation offers single-hand static gestures to ASL alphabets. Future versions can include dynamic gesture detection, word and sentence recognition, multilingual sign language support, and integration with mobile apps. The project's ultimate purpose is to enable inclusive communication, bridge accessibility gaps, and assist in advancing the overall mission of applying technology for social good.

II. PROBLEM STATEMENT

Individuals with hearing or speech disorders typically want to use sign language to communicate, but very few people have a knowledge of it. This creates a massive gap between both sides. Current solutions are focused on gesture detection but do not focus on interactive learning. As of now, there is no tool that can detect sign language in real-time and engage the user in teaching them how to learn it through a simple and fun, interactive manner. This project does just that, with the use of an internet-based platform that incorporates ASL detection and interactive learning features.

III. LITERATURE SURVEY

Early sign language recognition methods were based mainly on image processing methods and handcrafted features like edge detection, contour analysis, and skin color segmentation. These techniques were good for controlled environments but lacked scalability, varying lighting, background noise, and real-time capabilities. Traditional machine learning-based hand gesture recognition systems with algorithms such as SVM, KNN, and decision trees needed a lot of preprocessing and manual feature extraction, making them less adaptable [1].

With the advent of deep learning, CNNs gained favor due to their capacity to learn features directly from gesture images automatically. The authors in [2] suggested employing CNN architectures to discriminate static ASL alphabet signs with remarkable improvements in accuracy.

Later, MobileNet and EfficientNet architectures were introduced for their light-weight nature and efficiency, which made them appropriately chosen for real-time web and mobile applications. These models scored promising outcomes in terms of both speed and accuracy, particularly when deployed with TFLite. Research like [3] unleashed real-time sign language detection systems with an embedded text conversion, prioritizing accessibility for the deaf-mute community [4]. Others ventured into English-to-ASL translation interfaces that produce ASL signs from typed or spoken input [5]. An interactive web tool was suggested that integrates gesture detection with feedback tips and quizzes to facilitate learning, making sign language more accessible to non-signers. Utilization of Web Speech API for text-to-speech functionality across languages has also improved cross-language communication [6] [7].

Current advancements include transfer learning, data augmentation, and quantization of models for optimizing sign recognition systems for browsers and low-power devices [8] [9] [10]. In addition, the inclusion of backend visual analytics, e.g., detection accuracy scores, and gesture suggestions has been found beneficial for user engagement and education. These innovations reflect an increasing trend toward inclusive, intelligent platforms that enable both real-time detection and active learning within sign language systems.

IV. REQUIREMENTS

Operating System: Windows 10/11, Linux (Ubuntu is recommended), or macOS
Programming Language: Python 3.7 or later
Backend Framework: Flask - a light web application framework to handle backend routes and interactions

A. Key Python Libraries

1) TensorFlow 2.x - for deep learning model development and inference
2) OpenCV - for grabbing and processing webcam video feed in real-time
3) NumPy - for numerical computations and data handling
4) Pillow (PIL) - for image preprocessing and manipulation
5) Flask-SocketIO - to enable real-time communication between frontend and backend
6) Flask-CORS - to manage cross-origin resource sharing
7) Web Speech API - for text-to-speech conversion, voice output

B. Models Used

1) EfficientNet V2 - a high-performance deep learning model for image classification and sign detection
2) MobileNet V2 - an efficient and lightweight deep learning model for real-time inference on mobile and web applications
3) TFLite Model Format - TensorFlow Lite models for high-performance optimization and acceleration of inference on low-spec devices

C. Frontend Technologies

1) HTML5, CSS3, JavaScript - to develop the web application's user interface
2) opencv.js - for accessing webcam and image processing in-browser
3) connection.js - personal JavaScript to create WebSocket connections between server and client

D. Others

1) Web Browser - Google Chrome, Firefox, or Microsoft Edge
2) IDE - VS Code, PyCharm, or any other Python IDE
3) Package Manager: pip - used to install Python dependencies and libraries

V. PROPOSED SYSTEM

The suggested system is an interactive web-based program that enables the detection of American Sign Language (ASL) signs by using a webcam in real-time and offers a learning environment with quizzes for starters. The system is intended to fill the communication gap between hearing-impaired individuals and non-signers by translating hand gestures into readable and audible text using the predictive nature of machine learning models.

The detection component of the system employs pre-trained lightweight deep learning models (MobileNetV2 and EfficientNetV2) optimized for real-time execution using TensorFlow Lite. The webcam records the hand gestures of the user, processes the frames of the image, and predicts the signed alphabet with a corresponding confidence score. These predictions are rendered on the screen and can further be translated into speech using the Web Speech API, supporting multiple global languages.

The app also features other useful features:

1) English to ASL converter: Helps users learn how to sign any word.
2) Live detection console: Shows real-time predictions and accuracy levels.
3) Settings module: Allows users to personalize speech rate, voice choice, and detection thresholds.
4) Backend preview and feedback: Displays individual alphabet prediction performance with improvement advice.
5) Learning and Quiz section: Intended to assist users to learn ASL gradually using visual flashcards and interactive questions.

Developed with Python, Flask, HTML/CSS, JavaScript, OpenCV, and TensorFlow, the system is lightweight and functions well on basic hardware with an internet connection. It does not retain user data, maintain privacy and ethical use. In all, this system not only facilitates communication via real-time detection of signs but also fosters learning and knowledge of sign language, which makes it a significant tool for inclusiveness and education.

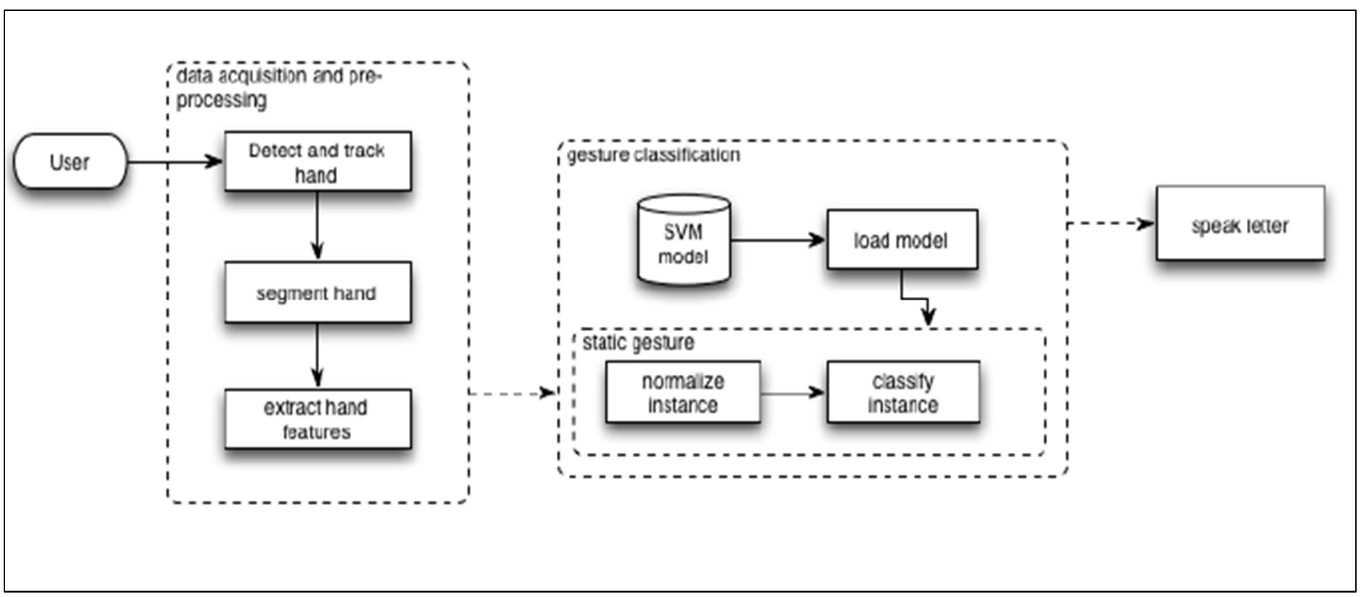


Fig 1: Prototype Diagram

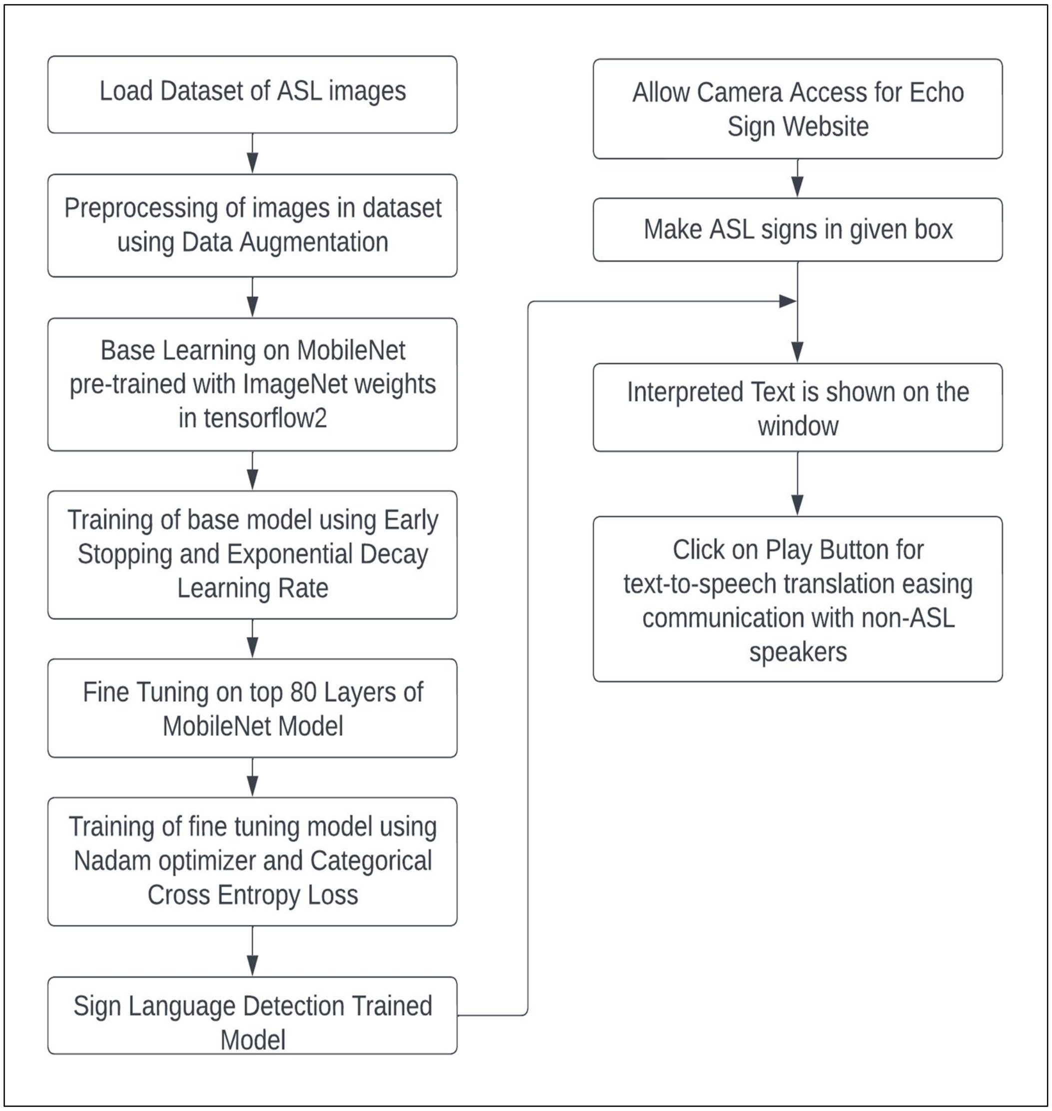


Fig 2: Block Diagram

VI. IMPLEMENTATION AND RESULTS

We built the system with Python featuring a Flask-based backend and a simple, yet intuitive frontend constructed using HTML, CSS, and JavaScript. For real-time American Sign Language detection, we utilized deep learning models mainly (MobileNetV2 and EfficientNetV2) pretrained on ImageNet and fine-tuned against the ASL alphabet dataset on Kaggle. To promote efficient operation on light devices, the trained models were adapted to TFLite model for easy deployment in web apps. OpenCV was used to capture and process live webcam feed frame by frame for enabling real-time recognition of hand gestures. Identified signs are translated to the respective characters or words and output along with confidence value. The platform has adjustable parameters like confidence thresholds, speech synthesis settings, and consecutive recognition criteria to extend customization.

Besides real-time detection, the project also involves a learning module, English-to-ASL converter, and quiz interface to support knowledge retention. These modules not only help users recognize and practice gestures but also as educational interfaces for inclusive learning. Upon testing, the MobileNetV2 model recorded a validation accuracy of 91.7% and EfficientNetV2 a slightly better accuracy at 95.0%, which shows the competence and stability of the models. Class-wise performance was studied through confusion matrices and accuracy distribution plots, which depicted regular recognition of most ASL alphabets. Overall, the system has great promise for real-world use in both assistive communication and sign language training.

VII. CONCLUSION

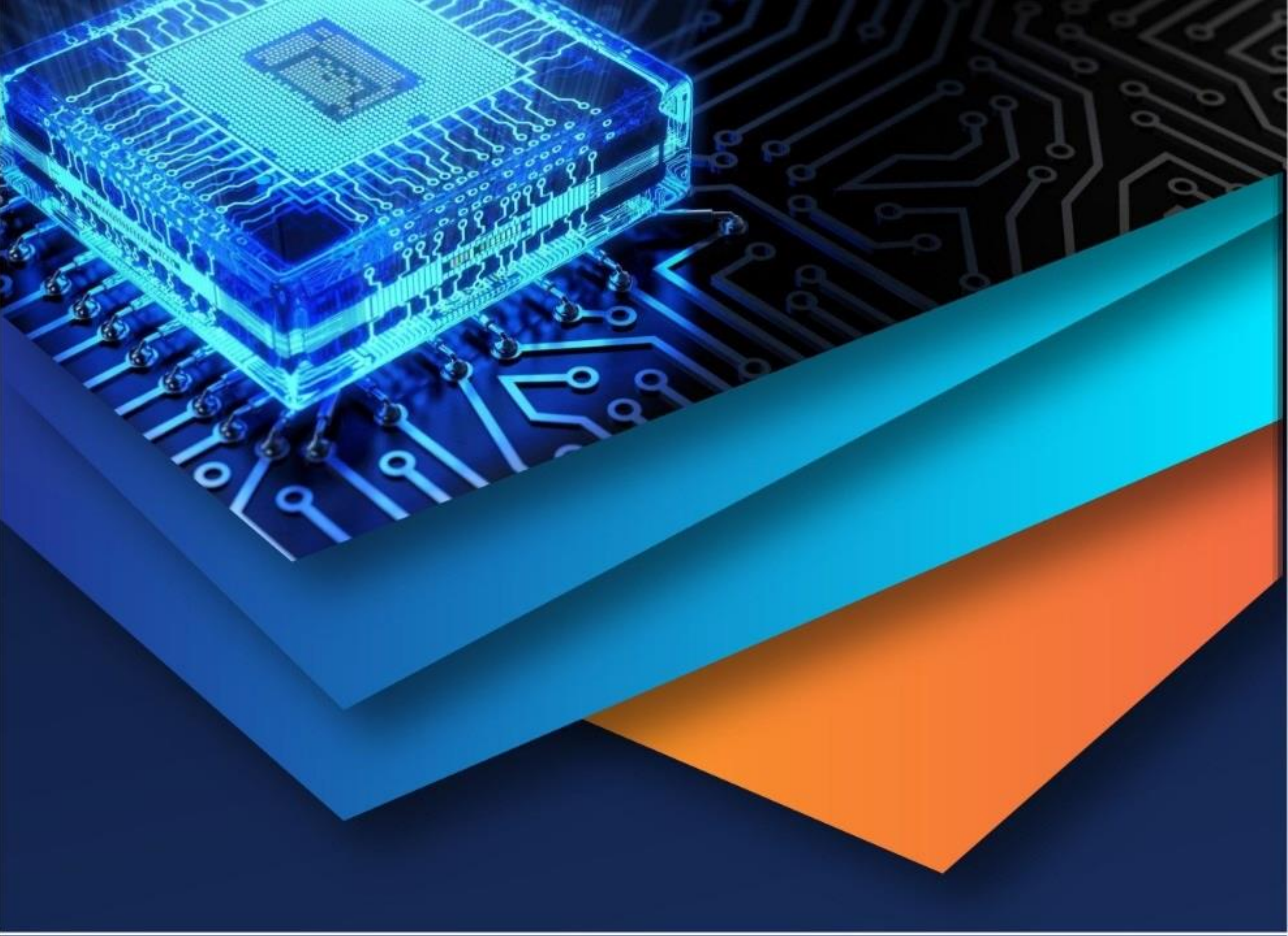
This project was fueled by a clear mission to connect the hearing-impaired with the rest of society through communication. Based on the MobileNetV2 deep learning model, we created a real-time ASL alphabet detection system that is both effective and user-friendly. Beyond detection, we added an interactive learning module featuring tutorials and quizzes to facilitate ongoing learning.
From data gathering to deployment, each process was crafted with accuracy and accessibility in mind. Validating at more than 91%, the system is both reliable and pragmatic for applications in everyday life. Future applications include extension to dynamic gesture recognition, entire sentence translation, and multilingual sign systems, opening doors to more accessible communication technologies.

VIII. ACKNOWLEDGMENT

This work is supported by the Department of Information Technology, Vidyalankar Institute of Technology, Mumbai.

REFERENCES

[1] E. J. H. Praiselin, "Sign Language Detection and Recognition Using Media Pipe and," International Journal of Scientific Research in Science and Technology, p. 9, 2024.
[2] M. J, V. Brahmadesam, S. N. S and S. K, "Sign Language Recognition using Machine Learning," IEEE, p. 5, 2022.
[3] A. A. I, B. O. O and A. A. M, "Machine learning methods for sign language recognition: A critical review and analysis," Elsevier, vol. 21, no. 2667-3053, p. 200056, 2021.
[4] C.-H. C. and C. A. G. , "Designing SmartSignPlay: An Interactive and Intelligent American Sign Language App for Children who are Deaf or Hard of Hearing and their Families," in Association for Computing Machinery, New York, 2016.
[5] H. Orovwode, I. D. Oduntan and J. Abubakar, "Development of a Sign Language Recognition System Using Machine Learning," 2023.
[6] H. Brashear, V. Henderson and K.-H. Park, "American sign language recognition in game development for deaf children," Ney York, 2006.
[7] P. Mekala, Y. Gao, J. Fan and A. Davari, "Real-time sign language recognition based on neural network architecture," 2011.
[8] A. Sahoo, G. Mishra and K. Ravulakollu, "Sign language recognition: State of the art," ARPN Journal of Engineering and Applied Sciences, vol. 9, pp. 116-134, 2014.
[9] A. Hekmat, H. Abbas and H. Shahadi, "Sign Language Recognition and Hand Gestures Review," Kerbala Journal for Engineering Science, 2022.
[10] Y. Z. and X. J. , "Recent Advances on Deep Learning for Sign Language Recognition," Computer Modeling in Engineering & Sciences, vol. 139, p. 2399–2450, 2024.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

TOGETHER WE REACH THE GOAL

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY