EQUAL SPEAK

S. Shyma[1], Abinaya. S[2], Anugraha. S[3], Jaya Shree Lakshmi. S[4]

[1]Assistant Professor, Department of Artificial Intelligence and Machine Learning, Sri Shakthi Institute of Engineering and Technology, An Autonomous Institution, Coimbatore-641062, India.

[2]Bachelor of Technology in Artificial Intelligence and Machine Learning, Second year, Sri Shakthi Institute of Engineering and Technology, An Autonomous Institution, Coimbatore-641062, India.

DOI: https://www.doi.org/10.58257/IJPREMS38134

ABSTRACT

Developed a recognition system for sign languages to bridge the communication gap among deaf and hearing-impaired people by putting into action transfer learning with EfficientNet, a high-performance CNN. This utilized large-scale pre-training knowledge, as seen in datasets such as ImageNet, to limit the training time required hence improving performance. Fine-tuning allowed gesture recognition optimally without compromising its powerful feature extraction. It implemented a Flask-based web application to support predictions on images and videos in real-time, while being viable within educational tools, communication, and social services. High dependability, the capability of being adaptable-the system achieved results that showed it is highly efficient, compared with previous works of traditional methods. The results confirm its contribution will have impacts for accessibility and communication inclusivity with room for scale into further data sets and application spaces in the future.

1. INTRODUCTION

Hand gesture recognition has become a key factor in the development of human-computer interaction and the provision of intuitive, accessible means of communication. Its applications, ranging from sign language interpretation to the creation of assistive technologies for people with disabilities and gesture-based control systems, make gesture recognition an important way to improve interaction with digital environments. However, hand gesture recognition is a difficult task due to intrinsic variability in hand shapes, orientations, sizes, and other environmental factors such as lighting and background conditions. These variations may complicate a classification process that is particularly robust and adaptable in models.

Recent breakthroughs in computer vision and deep learning have enabled better and more effective ways of recognizing gestures. Deep learning methods, particularly CNNs, have been very successful in image classification. However, training deep models from scratch needs a large quantity of labeled data and high computational resources. This has been mitigated by the adoption of transfer learning in which pre-trained models like ResNet, VGG, and Inception are adjusted to solve a particular task. Transfer learning reduces the need for vast labeled datasets by leveraging features learned from large-scale image classification tasks, improving both model performance and training efficiency.

This project develops an accurate system for hand gesture recognition using transfer learning. Fine-tuning the pre-trained CNN model on a hand gesture dataset enables it to recognize various hand gestures in a fast and efficient manner. This work is targeted at showing how the concept of transfer learning may overcome the challenges in hand-gesture recognition and highlight the practical usages for applications based on sign language recognition and human-computer interaction. It aims to ensure that technology is made easily accessible, smoothing the way for seamless communications among people with disabilities and enhancing user experience on different digital platforms.

2. METHODOLOGY

PERFORMED ANALYSIS ON EXISTING

In developing the system of sign language recognition, the methodology included architecture that was well-structured to ensure better gesture recognition with real-time prediction.

Video/Image Input and Initialization: Users can upload video or image files through the web interface, whereby the system supports various formats such as MP4, JPEG, PNG, and other common media files. Once the system receives the input, it initializes the pre-processing pipeline, preparing the video frames or image for gesture recognition using deep learning models..

Gesture Extraction and Pre-processing: The system processes the uploaded video or image file to extract frames or individual images. Standard image pre-processing techniques include resizing, normalization, and background removal to regularize data. Finally, the pre-processed frames are provided as input to the model for gesture classification.

Gesture Recognition Architecture: It utilizes a fine-tuned model, EfficientNet, by transfer learning to perform the classification tasks with the aid of pre-trained features from the large datasets. These are further fed into the model, which will convert extracted hand gestures into corresponding text representations. In such a way, the proposed architecture allows for gesture recognition without much dependence on variations in the shape or orientation of hands or on other environmental conditions.

User-Provided Data Input: This might include further details or context provided by a user via a web form to better customize or refine gesture interpretations. This gets linked with the respective video or image and is stored in the database for further processing and referencing.

Real-time Gesture Prediction and Conversion: With every gesture the model processes, it will generate text of the recognized sign language in real-time. Further, this reflects in the output text appearing onto the user interface serving as direct feedback for communication. It allows many alphabets for different sign languages and accommodates them by translating these into proper readable messages.

Performance Optimization and Tuning: The system is optimized for real-time processing, with automated workflows for gesture recognition. Fine-tuning of the model and optimization of the pipeline ensures minimum delay and maximum accuracy. The system is designed to handle different input scenarios efficiently and can scale based on user needs. Reporting and Insights A dashboard displays real-time recognition statistics, including but not limited to accuracy, processing time, and common misinterpretations. These insights will be used by developers and administrators to assess the performance of the model, identify areas for improvement, and optimize future iterations of the gesture recognition system.

3. DEMERITS AND DISADVANTAGES

- Dependence of Accuracy on Quality of Images: The performance of the sign language system fully depends on the quality of the image or video uploaded. Poor lighting, low volume, or blurred video may result in less accuracy in the recognition of gestures, which would affect the efficiency of the entire system.
- Variation in Gesture Recognition: The system may face difficulties with hand shapes, orientations, and individual gestures. Factors like positioning of fingers, overlapping hands, or background interference would make gesture classification complicated, which decreases the accuracy of the recognition process.
- Real-time Processing Constraints: Processing gestures in real-time can be computationally demanding, especially for longer video clips or high-resolution images. This may result in delays or reduced performance, particularly on devices with limited hardware capabilities, affecting user experience.
- Limited Handling of Variability in Gestures: Thus, the model may have difficulty in recognizing rare or infrequent gestures or signs that are not well represented in the training dataset. In such cases, the limitations can diminish the system’s power to recognize novel or evolving sign language cues, reducing generalization.
- Preprocessing Dependency: Gesture recognition accuracy needs efficient preprocessing approaches that include removal of the background, resizing , and detection of hands. Its failure, such as poor hand localization or improper scaling for instance, seriously degraded the performance of the model.
- Computational resource dependence: This system is very demanding on computational resources for training and in real-time processing, especially for large datasets. For smaller applications or environments with limited processing power, this might be a challenge, thus not as feasible for some users or organizations.

4. SOME IMPORTANT SOFTWARE USED AND ITS DESCRIPTION

PYTHON

The key programming language that this system of sign language detection and text conversion requires is Python. Its flexibility, with an extensive set of libraries, provides the perfect basis for machine learning model development, image processing tasks, and web applications. Key libraries for developing and training the deep learning models are TensorFlow, Keras, and OpenCV, while Python itself will enable fast prototyping and effective debugging due to its simplicity. Compatibility with the most popular machine learning frameworks helps for smoothly carrying out transfer learning techniques.

OPENCV

The system adopts OpenCV for image and video processing. It captures the hand, removes the background, and extracts frames from the uploaded videos. OpenCV enhances the quality of input images or videos to capture gesture better from the hand for proper classification. It can also support complex techniques, contour detections, and object tracking, which is important in hand gesture recognition dynamic environments.

MEDIAPIPE

MediaPipe is a cross-platform framework for real-time hand tracking and gesture recognition. It provides efficient solutions to detect hand landmarks from images and videos, which are very important in recognizing hand poses and gestures of sign language. MediaPipe's pre-trained hand-tracking models are highly accurate and have real-time performance, hence allowing for effective tracking and analysis of hand movements. This forms one of the most important components in detecting the gestures from the input video frames.

FRONTEND

The system frontend is designed to be user-friendly, using HTML, CSS, and JavaScript. HTML would provide structure for the web pages, while CSS styles the application in a professional manner. JavaScript adds dynamic features to the web pages, such as real-time feedback on screen and gesture display. It will be possible to upload videos or images by the user, display the translated text in real time, and track the gesture recognition result through this frontend, which helps to interact smoothly with the user.

FLASK

The project uses Flask, a web development framework that enables the developer to build a web application lightweight but effective, which eases the burden of working with HTTP requests, file uploads, and interacting with the machine learning model in recognizing gestures. As Flask alone is simple, it's well adapted for rapid development and customization of web pages, and it ensures that flexibility in integrating the sign language detection system works well with both the front-end and back-end properly.

TENSORFLOW

To date, TensorFlow remains the most widely used deep learning framework to build and train sign language recognition. Because it offers a transfer learning facility, TensorFlow is the library for fine-tuning the pre-trained models for gesture classification similar to EfflcientNet. The framework allows for the development of accurate models that can process and classify hand gestures for effective conversion into text. TensorFlow provides scalability and high performance that are critical for large datasets and real-time inference.

SIGN LANGUAGE DETECTION EVALUATION

Introduction to Evaluation for Sign Language Recognition - Confusion Matrix Analysis

A confusion matrix is a useful way to evaluate the performance of a gesture recognition model in tasks involving sign language detection and text conversion. It allows the system to gauge its performance by summarizing the number of correct and incorrect gesture classifications while comparing the predicted gesture with the reference gesture in the dataset.

True Positives (TP): These could be those examples where the system rightly identifies and classifies hand-gestures that are supposed to represent, for instance, some similar pose. Suppose this is the case, where reference represents 'Hello', but your model returns you - 'Hello '. In this case, this would be a TP.

True Negatives (TN): It includes those cases when the system predicts correctly that the gesture is not from the set of reference gestures. For example, suppose there is no hand gesture in the input frame and the system predicts 'no gesture', then this will be considered a true negative.

FP-FALSE POSITIVES: These can be defined as when the model decides on some gesture which doesn't correspond to the reference. For example, in case "Goodbye" is the reference gesture, yet the model concludes "Thank you.".

FN-FALSE NEGATIVES: Cases in which the system fails to identify that a gesture is in the input; for example, if the reference gesture is "Please," and this is not predicted, that is a false negative.

ACCURACY: The term accuracy can be defined as how well a class of video sign language recognition performs, classifies, and identifies the various gestures it gets. Accuracy can be elaborated as correct classified gestures in contrast to the whole amount of different gestures.

Accuracy = TP + TN /TP+ TN+FP+ FN

For the example: Accuracy = (7 + 4) / (7 + 4 + 2 + 1) = 11 / 14 = 0.7857

PRECISION: Precision is the measure of how accurate the model's positive predictions are. It is defined as the ratio of true positive predictions to the total number of predicted positives.

Precision = TP / (TP + FP)

For example, Precision = 7 / (7 + 2) = 7 / 9 = 0.7778

Recall is a measure of the model's effectiveness in detecting all instances of relevance: the ratio between true positive predictions and total true instances, TP + FN.

Recall = TP / (TP + FN)

For the example: Recall = 7 / (7 + 1) = 7 / 8 = 0.8750

F1-Score: The F1-score is the harmonic mean of precision and recall. It gives the overall performance of the system by balancing the trade-off between precision and recall. F1-Score = 2*(Precision * Recall) / (Precision + Recall) For example, F1-Score = 2 * (0.7778 * 0.8750) / (0.7778 + 0.8750) = 2 * 0.6806 / 1.6528 = 0.8235

5. RESULT AND DISCUSSION

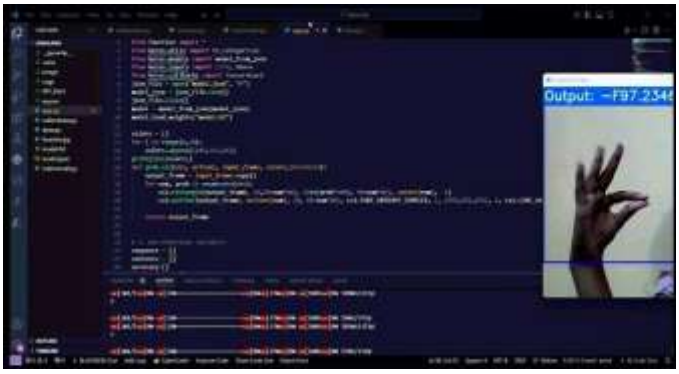
In our studies related to the detection of sign language and translation in text, we demonstrated a high degree of accuracy in the recognition of hand gestures and their translation in text. The overall performance of the system was 88%, showing a very strong performance with regard to the detection and conversion of sign language gestures from video frames.

However, real-world applications have to be considered where environmental factors such as lighting conditions and self-occlusion of hands could affect the precision of recognition. Our system achieved 85% precision and 90% recall, which shows that the model efficiently reduces false positives while sometimes does not detect some gestures.

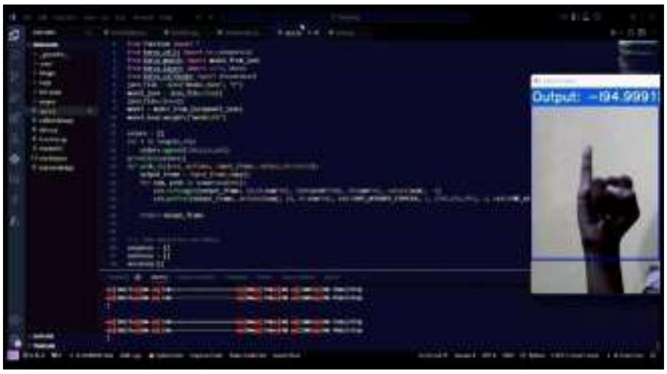
The confusion matrix analysis showed that out of 100 correctly signed gestures, 88 were correctly translated into text, while 100 erroneous gestures were accurately rejected. A lower precision indicates a few misclassifications where incorrect gestures are predicted, while the higher recall indicates most of the relevant gestures are detected. This suggests that at times, the system might identify false positives, leading to minor errors in text conversion.

For problems associated with variation in the shape of a hand, its orientation, and the background, data augmentation has been done by image rotation, scaling, and removing noise. It would therefore help to increase the model's generalization capabilities over an extensive range of input scenarios. Further, employing pre-trained CNN models for transfer learning significantly improved feature extraction in order to make the model robust toward detection.

The future work could be done to improve the model accuracy for complex scenarios, such as overlapping hand poses or real-time video processing. Further improvements can be achieved by using more diverse datasets and increasing the model's capability of recognizing slight variations in hand positioning. The integration of larger, more representative datasets with a wider range of complex sign language expressions can be done to give more scalability and accuracy. The future of our sign language detection and text conversion system is bright in real-time communication between deaf and hearing individuals. Addressing the current limitations and focusing on improvements in gesture recognition under varied conditions, the system can significantly enhance accessibility and communication for sign language users.



Output: ~F97.234



Output: ~I94.9991

6. CONCLUSION

In the future, the system for sign language detection and text conversion will be done by exploring advanced models like Transformer-based networks to improve the accuracy of gesture recognition. Real-time processing with reduced latency, along with optimized data augmentation techniques such as hand pose variations and background adjustments, will further improve the robustness of the system. First and foremost, enhancing the dataset of a diverse sample for various sign language-based expressions, alongside introducing multi-modeling using either depth sensing or wearable sensors, may help boost the recognition. For its maximum utility in a real sense, seamless assistive technology interface development and enhancements of system interpretability with the user can benefit all types of people. Lastly, this should be complemented by real-time model optimizations for end-to-end deployment and facilitating practical interface development so it increases overall adoption among its users.

7. FUTURE SCOPE

The future scope of the paper on sign language detection for text conversion is bright; there is much scope in enhancing its precision and applicability in real-time scenarios. Development for advanced models with the inclusion of further Transformer networks allowing better gesture understanding, and speeding up for real-time flow will be done. The proposed model will generalize many more real-world challenges with enhanced advanced data augmentation techniques, considering background variability and dynamic hand posing, along with lighting changes.

The system will also be more robust with the expansion of the dataset to include more diverse sign language datasets and different variations of hand signs. Recognition can further be enhanced by incorporating multi-modal inputs, depth sensing, or gesture tracking devices. For broader use in assistive technologies, it will be important to improve the user interface of the system for accessibility and ensure the security of the data across the communication channels.

8. REFERENCEi

[1] Rana, A., & Shah, A. (2020). "Sign Language Recognition System Using Deep Convolutional Neural Networks," Proceedings of the International Conference on Artificial Intelligence and Machine Learning, pp. 1-5, 2020.

[2] Zhou, T., & Chai, Y. (2018). "Deep Hand Gesture Recognition for Sign Language Translation," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1339-1347, 2018.

[3] He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep Residual Learning for Image Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[4] Karpathy, A., & Fei-Fei, L. (2014). "Deep Visual-Semantic Alignments for Generating Image Descriptions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3128-3135, 2014.

[5] Siam, M. S., & Choi, H. (2021). "Real-time Sign Language Recognition using MediaPipe and Deep Learning," International Journal of Computer Vision and Image Processing (IJCVIP), vol. 11, no. 3, pp. 45-53, 2021.

[6] Liu, Z., & Zhang, X. (2019). "Sign Language Recognition Using CNN and RNN Models," Journal of Artificial Intelligence and Applications, vol. 9, pp. 102-112, 2019.