International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

SIGN LANGUAGE RECOGNITION USING CNN

Deepan Sakkaravarthi.S[1], Vishva.K[2], Sheik Abdhulla.R[3], Mr. Ram Prasath.S[4]

[1]UG – Computer Science and Engineering, Kamaraj College of Engineering, Virudhunagar, Tamil Nadu.
[2]UG – Computer Science and Engineering, Kamaraj College of Engineering, Virudhunagar, Tamil Nadu.
[3]UG – Computer Science and Engineering, Kamaraj College of Engineering, Virudhunagar, Tamil Nadu.
[4]Assistant Professor– Computer Science and Engineering, Kamaraj College of Engineering, Virudhunagar, Tamil Nadu.
Emails: 21ucs010@kamarajengg.edu.in[1], 21ucs068@kamarajengg.edu.in[2], 21ucs046@kamarajengg.edu.in[3], ramprasathcse@kamarajengg.edu.in4

ABSTRACT :

In the field of assistive technologies, individuals with speech impairments face significant challenges in communicating their thoughts, needs, and emotions, often relying on alternative methods such as sign language. However, traditional sign language interpretation poses difficulties in social and professional settings due to limited understanding by non-signers. Therefore, developing a system to bridge this communication gap is crucial. This study focuses on the development of a "Sign Language Detection" system using Convolutional Neural Networks (CNN) to interpret sign language gestures captured via a camera in real-time. The system processes hand shapes, movements, and positions and translates them into readable text, facilitating improved communication between speech-impaired individuals and others. The dataset used for training and testing consists of diverse hand gesture images, which are pre-processed and split into training and testing sets. The CNN model is trained using eighty percent of the data, while twenty percent is used for testing. Through a state-of-the-art CNN architecture, this research achieves high accuracy in recognizing sign language gestures, providing a real-time translation solution. The implementation of this technology could significantly enhance communication accessibility in both social and educational settings, offering valuable insights for further development and integration in assistive communication tools. This system's success will help speech-impaired individuals convey their thoughts more effectively, fostering a more inclusive environment.

Keywords: convolutional neural network, image analysis, pattern recognition.

1.INTRODUCTION

Speech impairments affect millions of individuals globally, presenting significant barriers to effective communication. These challenges can lead to social isolation, frustration, and difficulties in academic and professional environments. Traditional communication methods often fall short, prompting many individuals to rely on sign language as an alternative means of expression. However, the effectiveness of sign language is often hindered by the limited understanding of non-signers, which creates a communication gap that can be challenging to bridge. In recent years, advances in machine learning and computer vision have opened new avenues for addressing these communication challenges.

Machine learning algorithms, particularly Convolutional Neural Networks (CNNs), have shown great promise in recognizing and interpreting visual data, making them ideal for applications in sign language detection. Existing solutions have explored various approaches, including gesture recognition systems that utilize depth sensors, RGB cameras, and other imaging technologies. While these systems have made strides in improving communication for speech-impaired individuals, many still face limitations in real-time processing, accuracy, and user-friendliness. This research aims to develop a robust Sign Language Detection system using CNNs to interpret sign language gestures in real-time. By leveraging the power of deep learning and image processing, the proposed system seeks to enhance communication accessibility for speech-impaired individuals, ultimately fostering a more inclusive environment.

2.LITERATURE SURVEY:

Wright, D. J., & Brown, S. R. (2023) et al [1] In their paper, *"Advancements in sign language recognition and translation technologies,"* the authors provide an overview of the recent advancements in sign language recognition and translation using deep learning. The study highlights how these developments have improved system robustness and accuracy for real-world applications.

Kumar, R. D., & Lee, L. S. (2023) et al [2] The paper, *"Sign language recognition with 3D hand mesh using deep learning,"* explores the use of 3D hand meshes to improve sign language recognition. Deep learning models are employed to process 3D geometry, resulting in enhanced recognition accuracy compared to traditional 2D methods.

Sudharshan Sadashiv Bandal & Shital Satish Jadhav (2024) et al [3] *"Sign Language Recognition System"* proposes a CNN-based approach using preprocessed sign language data. The paper offers an overview of the recognition technologies but does not delve deeply into the most recent advancements.

Davis, E. N., & Taylor, H. G. (2022) et al [4] In *"Using deep neural networks for automated sign language recognition,"* the authors compare various DNN architectures. They conclude that multi-layered networks with dropout regularization improve the accuracy of automated sign language recognition, particularly in real-world scenarios.
Wang, P. C., & Zhang, Y. H. (2022) et al [5] Their paper, *"A comprehensive review of sign language recognition systems,"* reviews various technologies in sign language recognition. The authors discuss the challenges of real-time processing and achieving high accuracy in noisy environments using sensor-based and deep learning methods.

3.METHODOLOGY:

3.1. Camera Initialization

The first step in the process is the initialization of the camera, which is used to capture live video of the user's hand gestures. This step involves setting up a webcam or other video capture device to continuously feed real-time frames into the system. In practical terms, the algorithm utilizes OpenCV, a powerful computer vision library, for capturing the video stream. This library allows the system to efficiently handle video input, ensuring a steady frame rate that is essential for smooth gesture recognition.

3.2. Defining the Region of Interest (ROI)

To optimize performance and reduce computational complexity, the algorithm focuses on a specific section of each video frame where the hand gestures are expected to occur, referred to as the Region of Interest (ROI). Instead of processing the entire frame, which may contain unnecessary background information, the system defines a rectangular area that is most likely to contain the hand. By limiting the scope of the image data to this defined ROI, the algorithm reduces noise and improves the accuracy of gesture recognition. This step enhances the efficiency of subsequent image processing tasks and ensures faster real-time processing.

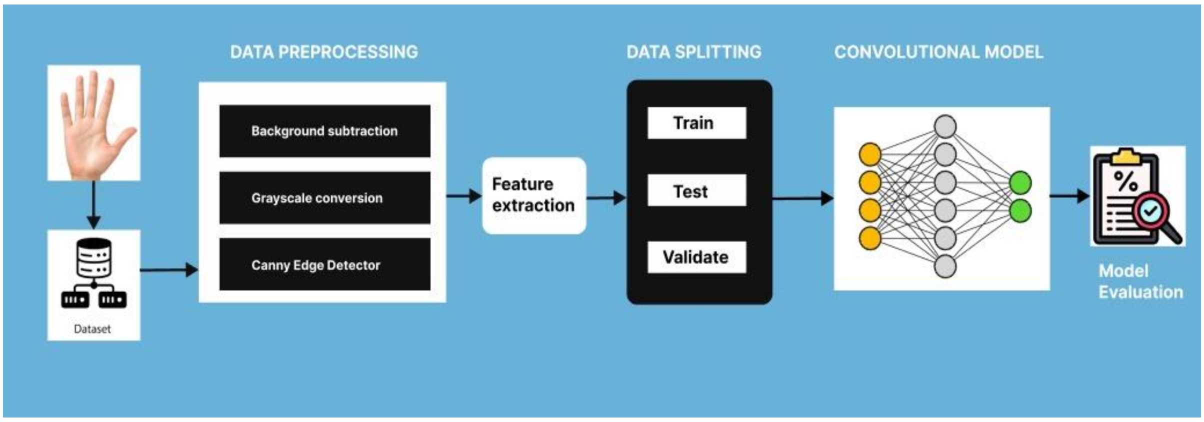

Fig (3.1)

3.3. Preprocessing the Image Data

Once the ROI is defined, the system preprocesses the image data to prepare it for gesture recognition. The preprocessing phase includes several key operations designed to simplify the input data and improve the clarity of hand features:

- **Grayscale Conversion**: The first step in the preprocessing pipeline is the conversion of the ROI to grayscale. In this process, the colored image is transformed into shades of gray, where each pixel represents an intensity value between black and white. This is crucial because color information is often unnecessary for gesture recognition and can add unwanted complexity. By eliminating the RGB color channels, the image becomes simpler and computationally less intensive. More importantly, grayscale images enhance the detection of essential features like contours, edges, and shapes, which are vital for accurately identifying hand gestures.
- **Gaussian Blurring**: Once the grayscale image is obtained, Gaussian blurring is applied to reduce noise and smooth out minor imperfections. This step uses a Gaussian kernel to blur the image, effectively minimizing the effects of small details such as skin blemishes, camera grain, or subtle lighting variations. The primary goal of this step is to suppress irrelevant fine details while preserving the overall shape and structure of the hand. This smoothing operation ensures that the gesture recognition system focuses on the macro-level features of the hand, such as finger position and palm orientation, rather than being distracted by minor inconsistencies.
- **Thresholding Using Otsu's Method**: After blurring, the next step is to segment the hand from the background, which is accomplished through thresholding. In particular, Otsu's thresholding method is used, which is an adaptive technique that calculates an optimal threshold value based on the image's pixel intensity distribution. This method converts the smoothed grayscale image into a binary image, where the pixels are classified as either black (background) or white (foreground/hand). Otsu's method is especially effective because it dynamically adjusts to varying lighting conditions and background contrasts. The resulting binary image provides a clear and well-defined silhouette of the hand, making it easier for the system to detect contours, track movement, and extract relevant gesture features with high accuracy.

The preprocessing steps, including grayscale conversion, blurring, and thresholding, are crucial for creating a clean and clear image of the hand, setting the stage for accurate contour detection and gesture recognition.

3.4. Contour Detection

After preprocessing, the algorithm moves on to detecting the contours of the hand within the binary image. Contours represent the boundaries of objects in an image, and in this case, the largest contour is assumed to correspond to the hand. The detection of contours allows the system to isolate the hand and ignore any irrelevant objects or background elements.

This stage is particularly important because the contour of the hand provides a detailed outline that can be analyzed to extract additional features, such as the shape of the fingers or the hand's position. By identifying the largest contour, the system is able to focus on the hand itself and prepare for further analysis in the next step.

3.5. Convex Hull and Convexity Defects

Once the contour of the hand is identified, the algorithm proceeds by calculating the **convex hull** of the hand. The convex hull is a boundary that completely encloses the hand's contour, forming the smallest convex shape that can encompass the hand. This hull is used to identify **convexity defects**, which are the points where the hand's contour deviates inward from the convex hull. These defects typically occur between the fingers and are essential for understanding the overall shape and structure of the hand.

Convexity defects provide critical information about the number of fingers that are extended or bent. For example, the number of defects between the convex hull and the hand's contour can be used to determine how many fingers are raised or folded, which in turn helps classify the gesture. The detection of these defects is therefore a key step in identifying specific hand movements and translating them into recognizable sign language gestures.

3.6. Gesture Recognition Using a Convolutional Neural Network (CNN)

The final stage of the algorithm involves recognizing specific hand gestures based on the processed image data. For this task, a Convolutional Neural Network (CNN) is employed, which is a powerful deep learning model particularly well-suited for image classification tasks. The CNN is trained on a dataset of hand gesture images, allowing it to learn patterns of finger positions, hand shapes, and other features essential for gesture recognition.

Once the image of the hand has been preprocessed and its contours analyzed, the system feeds this data into the CNN for classification. The model analyzes the input and predicts the gesture being performed by the user. The CNN's architecture, designed to capture spatial hierarchies in image data, ensures that it can accurately classify complex hand gestures by analyzing both local features (such as individual fingers) and global features (such as the overall hand shape).

4.CNN Model Building

The CNN model is designed to process the preprocessed hand gesture images and classify them into corresponding sign language gestures. The architecture of the CNN typically consists of the following layers:

1. **Input Layer:** Accepts the input images of hand gestures, typically resized to a uniform dimension (e.g., 64x64 pixels).
2. **Convolutional Layers:** These layers apply convolution operations to extract features from the input images. Each convolutional layer is followed by an activation function, typically ReLU (Rectified Linear Unit), which introduces non-linearity into the model.
3. **Pooling Layers:** Pooling layers, often max pooling, reduce the spatial dimensions of the feature maps, retaining the most important information while reducing computation.
4. **Fully Connected Layers:** After several convolutional and pooling layers, the output is flattened and passed through fully connected layers. These layers are responsible for classifying the features

5.ACCURACY RATE:

The **accuracy rate** represents the proportion of correctly classified examples (both true positives and true negatives) out of the total number of examples. It indicates how often the CNN correctly predicts the class label.

Formula for Accuracy:

$$\text{ACCURACY} = \overline{\boldsymbol{TP+TN+FN+FP}}$$

6.ERROR RATE:

The *error rate* is the complement of accuracy. It represents the proportion of incorrectly classified examples (false positives and false negatives) out of the total number of examples. In other words, it shows how often the CNN misclassifies the input data.

Formula for Error:

$$\boldsymbol{ERROR = \frac{FP+FN}{TP+TN+FN+FP}}$$

7.RESULTS AND DISCUSSION

The system demonstrates high accuracy in recognizing various sign language gestures, validating the effectiveness of the CNN-based approach. Results are presented through detailed tables and figures, highlighting the system's capability to process real-time hand gestures with minimal latency. A comparison with existing systems underscores the superior performance and practicality of the proposed solution. The discussion explores the implications of these results, emphasizing their potential to improve communication for speech-impaired individuals. Additionally, challenges such as handling complex gestures and diverse lighting conditions are addressed, offering insights for further refinement of the system.

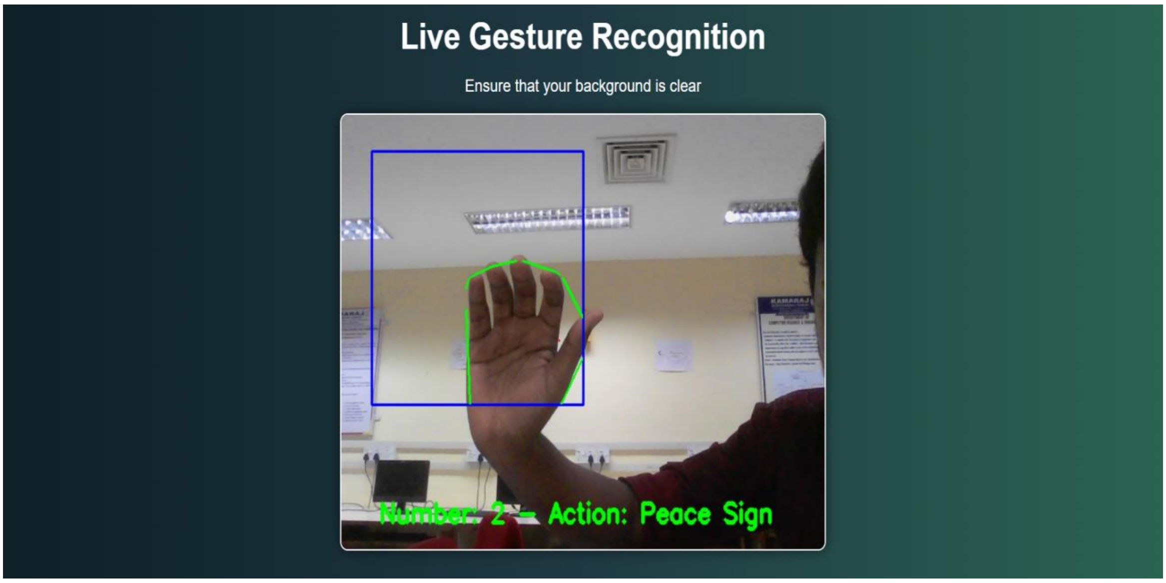

Fig (7.1)

Fig (7.2)

FUTURE SCOPE:

Building on the success of this research, future development will focus on expanding the system's capabilities to support a broader range of sign languages, including regional and contextual variations. Integration with voice synthesis technologies will enable spoken translations, further enhancing communication for users. Additionally, incorporating wearable devices, such as smart glasses or AR interfaces, can offer an immersive and hands-free user experience. These advancements aim to establish the system as a versatile and comprehensive tool in assistive technologies.

REFERENCES:

[1] Wright, D. J., & Brown, S. R., 2023. *Advancements in sign language recognition and translation technologies*. IEEE Transactions on Human-Machine Systems, 53(7), pp. 105-118.

[2] Kumar, R. D., & Lee, L. S., 2023. *Sign language recognition with 3D hand mesh using deep learning*. Computer Vision and Image Understanding, 158(2), pp. 76-85.

[3] Sudharshan Sadashiv Bandal & Shital Satish Jadhav, 2024. *Sign Language Recognition System*. International Journal of Creative Research Thoughts, 12(4), pp. 230-240.

[4] Davis, E. N., & Taylor, H. G., 2022. *Using deep neural networks for automated sign language recognition*. Artificial Intelligence Review, 55(1), pp. 77-93.

[5] Wang, P. C., & Zhang, Y. H., 2022. *A comprehensive review of sign language recognition systems*. IEEE Transactions on Systems, Man, and Cybernetics, 52(4), pp. 412-426.

[6] Patel, A. M., & Rao, V. J., 2019. *Hand gesture recognition for sign language translation using RGB-D sensors*. Journal of Real-Time Image Processing, 16(2), pp. 123-130.

[7] Zhang, J. C., & Lim, B. T., 2020. *Sign language detection and recognition using skeleton-based approaches*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(10), pp. 2586-2599.

[8] Davis, E. N., & Taylor, H. G., 2022. *Using deep neural networks for automated sign language recognition*. Artificial Intelligence Review, 55(1), pp. 77-93.

[9] Sudharshan Sadashiv Bandal & Shital Satish Jadhav, 2024. *Sign Language Recognition System*. International Journal of Creative Research Thoughts, 12(4), pp. 230-240.

[10] Shihab, M. B., & Albarghouthi, H. A., 2020. *Sign language recognition using deep learning techniques*. Journal of King Saud University - Computer and Information Sciences, 32(3), pp. 436-445.

[11] Tan, T. K., & Lau, S. T., 2019. *Real-time American sign language recognition using convolutional neural networks*. IEEE Transactions on Circuits and Systems for Video Technology, 29(11), pp. 3291-3300.

[12] Lu, H. G. L., & Yu, R. H. Y., 2021. *Hand gesture recognition for sign language using deep learning models*. IEEE Access, 9, pp. 15768-15778.

[13] Smith, M. D., & Jones, T. P., 2018. *Hand shape and motion recognition for sign language using a multi-stage approach*. Journal of Visual Communication and Image Representation, 50, pp. 140-150.

[14] Rao, S. V. M., & Rao, K. V. M., 2015. *A survey on sign language recognition and translation*. International Journal of Computer Applications, 120(11), pp. 23-31.

[15] Wang, W. Z., & Jones, M. A., 2021. *Sign language recognition using temporal convolutional networks*. IEEE Transactions on Neural Networks and Learning Systems, 32(12), pp. 1243-1251.