



OPEN Improving sign Language recognition system for assisting deaf and dumb people using pathfinder algorithm with representation learning model

Nadhem Nemri^{1✉}, Mohammed Yahya Alzahrani², Wided Bouchelligua³ & Amani A. Alneil^{4,5}

For many individuals, communication through sign language (SL) is the primary means of interacting with the world, and the potential applications of effective SL Recognition (SLR) systems are vast and far-reaching. SLR is a research area dedicated to the automatic analysis of hand gestures and other visual signs used in communication among individuals with speech or hearing impairments. Despite significant advancements, the automated detection and interpretation of human signs remain a complex and multidisciplinary challenge that is yet to be fully addressed. Recently, various approaches have been explored, including the application of machine learning (ML) models in SLR. With advancements in deep learning (DL), sign recognition systems have become more accurate and adaptable, helping to bridge the communication gap for individuals with hearing impairments. Building upon these developments, the present study introduces a novel approach by integrating an advanced optimization strategy with a representation learning model, aiming to improve the robustness, accuracy, and real-world effectiveness of SLR systems. This paper proposes a Pathfinder Algorithm-based Sign Language Recognition System for Assisting Deaf and Dumb People Using a Feature Extraction Model (PASLR-DDPFEM) approach. The aim is to enhance SLR techniques to help individuals with hearing challenges communicate effectively with others. Initially, the image pre-processing phase is performed by using the Gaussian filtering (GF) model to improve image quality by removing the noise. Furthermore, the PASLR-DDPFEM approach utilizes the SE-DenseNet model for feature extraction. Moreover, the Elman neural network (ENN) model is implemented for the SLR classification process. Finally, the parameter tuning process is performed by using the Pathfinder Algorithm (PFA) model to enhance the classification performance of the ENN classifier. An extensive set of simulations of the PASLR-DDPFEM method is accomplished under the American SL (ASL) dataset. The comparison study of the PASLR-DDPFEM method revealed a superior accuracy value of 98.80% compared to existing models.

Keywords Pathfinder algorithm, Sign language recognition, Deaf and dumb people, Feature extraction, Elman neural network

According to the World Health Organisation (WHO), approximately 70 million people worldwide have hearing loss. A high number of people with hearing and speech impairments may struggle to write or read in everyday language¹. SL is one of the non-verbal languages used by deaf people for day-to-day communication among themselves. SL primarily relies on gestures more than voice to convey messages, incorporating the use of facial expressions, finger shapes, and hand movements². The following are the essential defects in this language: a limited vocabulary, difficulties in learning, and frequent hand movements. In addition to this, people who are

¹Department of Information Systems, Applied College at Mahayil, King Khalid University, Khalid, Saudi Arabia.

²Faculty of Computing and Information, Al-Baha University, Al-Baha, Saudi Arabia. ³Applied College, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11432, Saudi Arabia. ⁴Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia. ⁵King Salman Centre for Disability Research, Riyadh 11614, Saudi Arabia. ✉email: nnemri@kku.edu.sa

not deaf and mute are unaware of SL, while disabled people face significant problems in communicating with individuals³. These individuals with disabilities need to utilize a device translator for communicating with able-bodied individuals, which is achieved through the development of glove equipment with electronic circuits and sensors⁴. Many efforts were made to create an SLR method last year. In SLR, there are two major classifications, namely continuous sign classification and isolated SL. The hidden Markov model (HMM) functions on continuous SLR, which allows the segmentation of an information stream⁵. The SLR design is characterized into two essential types depending on its input, namely vision-based and data glove-based⁶. Vision-based SLR methods use cameras to detect hand gestures. Glove-based SLR technique utilizes smart gloves to measure locations, velocity, orientation, and other parameters, which employ sensors and microcontrollers.

Computer vision (CV)-based SLR methods commonly depend on removing characteristics such as gesture detection, edge detection, shape detection, and skin colour segmentation, among others⁷. In recent years, the use of a vision-based approach has become increasingly common, utilizing input from a camera. Many of the studies in SLR depend on DL methods that were achieved on SLs, unlike any Indian SL⁸. Currently, these fields are gaining more popularity among scholarly experts. The past reporting work on SLR primarily depends on ML models. These techniques result in lower accuracy because they do not automatically remove characteristics⁹. Automatic feature engineering is the primary objective of DL methods. The idea behind this is to spontaneously study a group of attributes from raw information used to recognize SL by individuals with hearing loss¹⁰. A communication gap exists between hearing-impaired individuals and those who are speech-impaired, as well as the general populace. Conventional tools for bridging this gap, such as sensor-based gloves, can be costly, inconvenient, or limited in scope; hence, it becomes crucial for intelligent, real-time solutions to interpret SL naturally and accurately. DL is considered effective in this area, enabling systems to automatically learn intrinsic patterns in gestures and facial cues without manual feature extraction. Its success in image and sequence recognition makes it ideal for advancing disability detection and improving communication accessibility.

This paper proposes a Pathfinder Algorithm-based Sign Language Recognition System for Assisting Deaf and Dumb People Using a Feature Extraction Model (PASLR-DDPFEM) approach. The aim is to enhance SLR techniques to help individuals with hearing challenges communicate effectively with others. Initially, the image pre-processing phase is performed by using the Gaussian filtering (GF) model to improve image quality by removing the noise. Furthermore, the PASLR-DDPFEM approach utilizes the SE-DenseNet model for feature extraction. Moreover, the Elman neural network (ENN) model is implemented for the SLR classification process. Finally, the parameter tuning process is performed by using the Pathfinder Algorithm (PFA) model to enhance the classification performance of the ENN classifier. An extensive set of simulations of the PASLR-DDPFEM method is accomplished under the American SL (ASL) dataset. The key contribution of the PASLR-DDPFEM method is listed below.

- The PASLR-DDPFEM model incorporates GF to enhance image quality and reduce noise in SL input images, thereby ensuring cleaner visual data. This process enhances the visibility of significant features, facilitating more precise feature extraction and ultimately improving the overall performance and reliability of the recognition system.
- The PASLR-DDPFEM method employs the SE-DenseNet-based DL approach for efficient and discriminative feature extraction, capturing both spatial and channel-wise data. This enhances the model's ability to focus on the most relevant features, resulting in improved recognition accuracy and robustness across varying SL image conditions.
- The PASLR-DDPFEM approach utilizes the ENN technique for effective SLR classification, employing its feedback connections to capture temporal patterns. This enables the model to better comprehend sequential dependencies in sign gestures, improving classification accuracy and adaptability to dynamic inputs.
- The PASLR-DDPFEM methodology utilizes the PFA model to tune ENN parameters, thereby enhancing overall classification accuracy by efficiently exploring the solution space. This optimization improves the convergence speed and stability of the ENN model, producing more reliable and precise SLR results.
- Thus, a novel hybrid framework is introduced which is required for ASL as it effectively handles image noise, captures discriminative spatial-temporal features, adapts to gesture discrepancies, and optimizes model parameters for accurate and robust recognition. This unique integration leverages the strengths of each component to address threats in SLR. The model enhances accuracy, robustness, and adaptability, and the novelty is in the synergistic use of DL, recurrent networks, and metaheuristic optimization.

Literature of works

Rethick et al.¹¹ presented an innovative model of online hand gesture detection and classification methods. CNN is utilized to present effective and intuitive methods of communication for individuals. The primary goal is to provide deaf individuals with access to real-world gesture recognition technology. This method utilizes a robust CNN framework, specifically designed for precise hand gesture detection, and trained on a meticulously curated dataset. Assiri and Selim¹² developed a model by utilizing an Adaptive Bilateral Filtering (ABF) model for noise reduction, the Swin Transformer (ST) technique for effective feature extraction, a hybrid CNN and Bi-directional Long Short-Term Memory (CNN-BiLSTM) model for accurate classification, and the Secretary Bird Optimiser Algorithm (SBOA) for optimal hyperparameter tuning. Kumar, Reddy, and Swetha¹³ presented a reliable and real-time Hindi SL (HSL) recognition system by utilizing CNNs for spatial feature extraction and recurrent neural networks (RNNs) for temporal sequence modelling of hand movements and facial expressions. Harshini et al.¹⁴ proposed an accurate SLR system by using ML models. Specifically, the Random Forest Classifier (RFC) is incorporated with a conversational AI (CAI) bot powered by the Google Gemini Model (GGM). Allehaibi¹⁵ presented a Robust Gesture SLR Utilizing Chicken Earthworm Optimiser with DL (RSLR-CEWODL) technique. The proposed approach utilizes the ResNet-101 method for feature extraction. For the optimum hyperparameter

tuning process, the projected model leverages the CEWO model. Moreover, the presented model employs a whale optimizer algorithm (WOA) with a deep belief network (DBN) for SLR. Kumar et al.¹⁶ introduced a new technique for enhancing the detection of Indian SL (ISL) by integrating Deep CNN with physically intended aspects. It employs the capability of DL with CNN to autonomously attain distinctive features from unprocessed data. This model involves a multi-stage process, in which the deep CNN gathers progressive features from unprocessed ISL images. In contrast, the manually intended aspects provide additional data to improve the recognition process. Hariharan et al.¹⁷ developed a highly accurate American SL (ASL) recognition system by utilizing advanced image pre-processing techniques, a modified Canny edge detection for segmentation, and a Modified CNN (MCNN) based on the deep Residual Network 101 (ResNet-101) architecture for classification. Almajally and Almukadi¹⁸ proposed an advanced SLR system that utilizes bilateral filtering (BF) for noise reduction, ResNet-152 for feature extraction, and a Bi-directional Long Short-Term Memory (Bi-LSTM) method for sequence modelling. The Harris Hawk Optimisation (HHO) technique is employed to tune the hyperparameters of the Bi-LSTM optimally.

Kaur et al.¹⁹ developed a real-time SL to speech conversion system by utilizing a pre-trained InceptionResNetV2 DL technique integrated with hand keypoint extraction techniques. The model is examined by using the ASL dataset. Almajally et al.²⁰ introduced a model utilizing advanced image pre-processing techniques, including Contrast-Limited Adaptive Histogram Equalisation (CLAHE) and Canny Edge Detection (CED). The model also incorporates multiple feature extractors, including ST, ConvNeXt-Large, and ResNet50, combined with a hybrid CNN and Bi-LSTM with Attention (CNN-BiLSTM-A) for precise classification. Jagdish and Raju²¹ proposed a technique by utilizing image processing and DL models, specifically CNN. Maashi, Iskandar, and Rizwanullah²² presented a Smart Assistive Communication System for the Hearing-Impaired (SACHI) methodology, utilizing BF for noise reduction, an improved MobileNetV3 for effective feature extraction, and a hybrid CNN with a Bi-directional Gated Recurrent Unit and Attention (CNN-BiGRU-A) method for accurate SLR. The Attraction-Repulsion Optimisation Algorithm (AROA) approach is used to tune the classifier's hyperparameters optimally. Ilakkia et al.²³ proposed a real-time ISL recognition system by utilizing DL techniques, specifically the Residual Network-50 (ResNet-50) architecture. Mosleh et al.²⁴ introduced a bidirectional real-time Arabic SL (ArSL) translation system by utilizing transfer learning (TL) with CNNs and fuzzy string-matching techniques. Thakkar, Kittur, and Munshi²⁵ presented a robust multilingual SL Translation (SLT) system by integrating advanced CV techniques like YOLOv5 for gesture detection, combined with Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models for machine translation across English, Hindi, and French. The model also used RF classifiers with frameworks such as OpenCV and MediaPipe. Dhaarini, Sanjai, and Sandosh²⁶ developed a real-time SL Detection and Assistive System (SLDAS) by utilizing advanced CV techniques and the You Only Look Once version 10 (YOLOv10) object detection model. Choudhari et al.²⁷ developed a platform-independent web application for real-time ISL recognition by utilizing a CNN with Leaky Rectified Linear Unit (Leaky ReLU) activation and Adam optimizer. Table 1 summarises the existing studies on SLR systems for assisting the deaf and dumb.

Though the existing studies are effectual in the SLR recognition process, several approaches depend on limited or domain-specific datasets, mitigating generalizability across diverse SLs. Various techniques lack robust handling of dynamic sequences and non-manual features such as facial expressions and real-time responsiveness is often compromised due to intrinsic architectures or high computational overhead. Most systems lack end-to-end bidirectional communication capabilities though the integration of DL techniques namely CNN, BiLSTM, and ST has illustrated promising results. A notable research gap exists in forming lightweight, scalable models optimized for edge deployment and cross-lingual adaptability. Furthermore, insufficient multimodal integration and limited interpretability in decision-making highlight further research gap in building inclusive and user-friendly SLR platforms.

Materials and methods

This paper designs and develops a PASLR-DDPFEM technique. The primary objective is to enhance SLR techniques to help individuals with hearing challenges communicate effectively with others. To accomplish this, the PASLR-DDPFEM model involves several stages, including image pre-processing, feature extraction, classification, and parameter tuning. Figure 1 depicts the overall working flow of the PASLR-DDPFEM model.

Dataset description

Table 2 consists of 78,000 samples with 26 classes, representing the letters A–Z³¹. Each image is sized at 200 × 200 pixels, making it appropriate for training DL techniques in gesture recognition. The dataset also includes a small test set with real-world examples to promote robust model evaluation.

GF-based image pre-processing

Initially, the image pre-processing phase is performed using the GF model to enhance image quality by removing noise². This model is chosen for its simplicity, efficiency, and efficiency in mitigating high-frequency noise while preserving crucial edge details in SL images. Unlike more complex filtering methods, the GF model presents a good balance between noise suppression and computational cost, making it appropriate for real-time applications. The technique helps improve the quality of input data without introducing distortions and also smoothens the image uniformly. This ensures that crucial gesture features remain intact for accurate downstream processing. Compared to median or bilateral filters, GF provides faster execution and consistent results. Its integration improves the reliability of feature extraction and overall recognition accuracy.

GF is a vital pre-processing stage in SLR for reducing noise and improving image quality while conserving essential features. It uses a Gaussian function to blur an image, minimizing high-frequency deviations that may occur due to illumination variations or sensor noise. This aids in improving feature extraction and edge

Ref. No.	Objective	Method	Dataset	Measures
Rethick et al ¹¹ .	To develop a real-time hand gesture recognition system to empower the deaf and mute community.	CNN, Real-time gesture detection, Custom CNN architecture, Fine-tuning for accuracy	Diverse ASL gesture images (29 classes, 87,000 images)	Accuracy of 99.11%, Real-Time Performance
Assiri and Selim ¹²	To develop a robust hand gesture recognition system for helping hearing-impaired individuals.	ABF, ST, CNN-BiLSTM, SBOA	Traffic Police Gesture Dataset	Accuracy of 99.25%, Performance Validation
Kumar, Reddy, and Swetha ¹³	To develop a real-time system for converting gestures into text and speech to enhance communication for the Deaf and Hard of Hearing community.	CNN, RNN	Hindi SL Gesture Dataset	Accuracy, Real-Time Performance
Harshini et al ¹⁴ .	To develop an SLR system for SL users in digital environments.	RF, Comparison with CNN and KNN, Integration with Conversational AI	Diverse SL Gesture Dataset	Accuracy of 0.9961, User Responsiveness
Allehaibi ¹⁵	To develop a robust SLR system using optimized DL models for accurate gesture classification.	RSLR-CEWODL, ResNet-101, WOA, DBN	SL Gesture Datasets	Accuracy, Performance Evaluation
Kumar et al ¹⁶ .	To enhance ISL recognition by integrating DL with manually designed features for improved accuracy and robustness.	Deep CNN, DL	Extensive ISL Dataset	Recognition Accuracy, Robustness
Hariharan et al ¹⁷ .	To develop an accurate ASL recognition system.	MCNN, ResNet-101	ASL Hand Gesture Images (36 Signs)	Accuracy (97%), False Positive Rate (0.05%)
Almjally and Almkadi ¹⁸	To develop an optimized DL technique for accurate and automatic SLR.	BF, ResNet-152, Bi-LSTM, HHO	SL Dataset	Precision, Recall, Accuracy, and F1-Score of 94.72%, 94.74%, 98.95, and 94.72%
Kaur et al ¹⁹ .	To develop a real-time SL-to-speech conversion system.	InceptionResNetV2 DL, Hand keypoint extraction, Python image processing, Model training with epochs	7200 Images, 24 Alphabet Classes (Excluding 'j' and 'z')	Training And Validation Accuracy
Almjally et al ²⁰ .	To enhance accurate and real-time SLR using an attention-driven hybrid DL technique with feature fusion.	CLAHE, CED, ST, ConvNeXt-Large, ResNet50, CNN-BiLSTM-A	SL Dataset	Accuracy of 98.10%, Precision of 95.28%, Recall of 95.28%, and F1-Score of 95.28%
Jagdish and Raju ²¹	To develop a DL-based system for accurate detection and recognition of SL gestures.	CNN Model Training, Sign-to-Text Conversion, Voice Output Integration	SL Gesture Images	Accuracy, Accessibility
Maashi, Iskandar, and Rizwanullah ²²	To develop an intelligent SLR system using a hybrid DL method to assist hearing-impaired individuals.	SACHI, MobileNetV3, CNN-BiGRU-A, AROA	ISL Dataset	Precision of 91.54%, Recall of 93.21%, Accuracy of 99.19%, and F-Score of 91.87%, respectively.
Ilakkia et al ²³ .	To develop a real-time ISL recognition system to translate ISL gestures into text for the deaf community.	DL, ResNet-50	Unique ISL Dataset	Accuracy, Real-Time Performance
Mosleh et al ²⁴ .	To develop a robust, real-time, bidirectional ArSL translation system to enhance communication for deaf individuals.	ArSL, CNN	ArSL Dataset and Arabic Data Dictionary	Accuracy, Processing Efficiency
Thakkar, Kittur, and Munshi ²⁵	To develop a multilingual SL translation system to facilitate communication between hearing, visually impaired, and auditory-impaired individuals.	YOLOv5, LSTM-GRU, RF, OpenCV & MediaPipe integration, Auto-tokenizer and Adam optimizer	Multilingual SL Images	Translation Accuracy, Processing Speed
Dhaarini, Sanjai, and Sandosh ²⁶	To develop a real-time SL Detection and Assistive System for deaf and mute individuals.	SLDAS, CV, YOLOv10	ASL Gesture Dataset	Accuracy, Real-Time Performance
Choudhari et al ²⁷ .	To develop a platform-independent web-based system for real-time ISL translation into text.	CNN, Leaky ReLU, Adam Optimiser	1200 Images, 35 Classes (26 Alphabets + 9 Numbers)	Accuracy of 97%, Real-time Performance

Table 1. Summary of existing studies on SLR systems for assisting hearing and speech-impaired individuals.

detection by decreasing unwanted artefacts. The filter functions by conveying advanced weights to vital pixels and gradually declining weights to surrounding pixels, which ensures a natural smoothing effect. In SLR, GF enhances hand and gesture segmentation, which makes it simpler for ML methods to identify key patterns. Properly adjusting the Gaussian kernel size is crucial to strike a balance between reducing noise and preserving detail.

SE-DenseNet-based feature extraction model

Furthermore, the PASLR-DDPFEM method involves a feature extraction process, which is executed by the SE-DenseNet model²⁸. This supervised DL method is chosen for its superior capability in capturing both spatial and channel-wise feature representations, improving the discriminative power of extracted features. The dense connectivity of the model promotes feature reuse. It reduces vanishing gradient issues, while the SE blocks adaptively recalibrate channel-wise responses, allowing the model to concentrate on the most informative features. Compared to conventional CNNs, SE-DenseNet presents enhanced efficiency and accuracy with fewer parameters. This integration yields richer feature hierarchies and improved generalization. Its integration ensures more robust and precise recognition of complex SL gestures under varying conditions.

DenseNet is an enhanced CNN-based model that calculates dense multiscale attributes from the object classifier's convolution layer. This dense calculation of characteristics from the entire image may speed up training. This structure utilizes dense links, connecting the output of all layers to the input of all succeeding layers, thus decreasing the parameter counts and computing costs without influencing performance. The densely

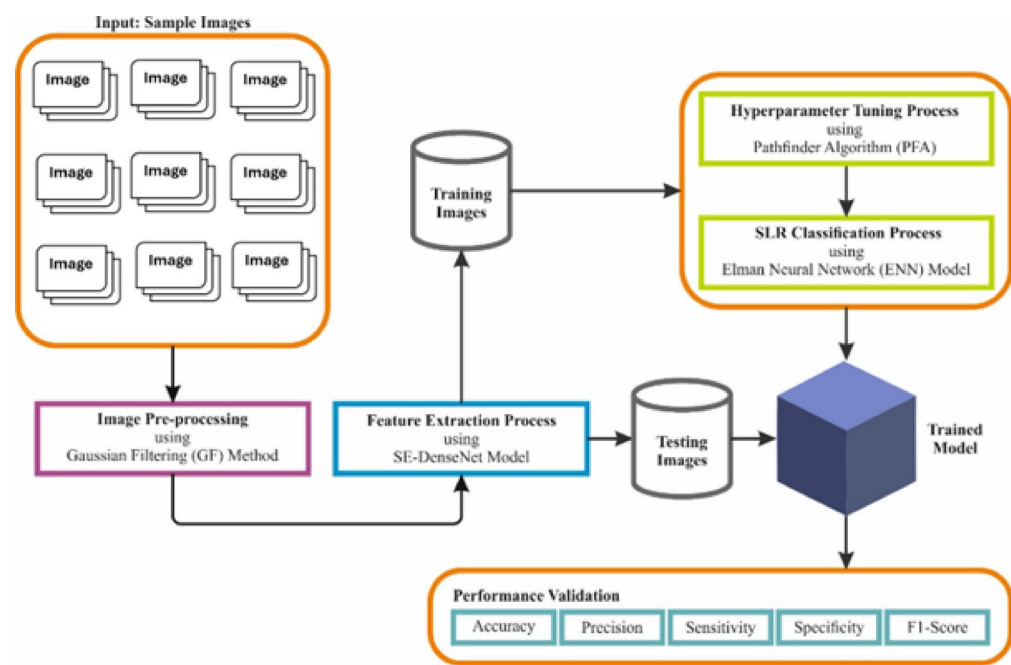


Fig. 1. Overall working process of the PASLR-DDFEM model.

Signs	Labels	No. of Count	Signs	Labels	No. of Count
A	L-1	3000	N	L-14	3000
B	L-2	3000	O	L-15	3000
C	L-3	3000	P	L-16	3000
D	L-4	3000	Q	L-17	3000
E	L-5	3000	R	L-18	3000
F	L-6	3000	S	L-19	3000
G	L-7	3000	T	L-20	3000
H	L-8	3000	U	L-21	3000
I	L-9	3000	V	L-22	3000
J	L-10	3000	W	L-23	3000
K	L-11	3000	X	L-24	3000
L	L-12	3000	Y	L-25	3000
M	L-13	3000	Z	L-26	3000
		Total	Count	78,000	

Table 2. Details of the dataset.

linked architecture improves the gradients and information flow; alleviating difficulties associated with vanishing gradients. The DenseNet’s basic notion is to achieve powerful feature representation and gradient propagation by minimizing information loss, thereby increasing the system’s performance. Equation (1) is applied to signify the initial layer input of DenseNet.

$$x_l = H_l([x_0, x_1, x_2, \dots, x_{l-1}]) \tag{1}$$

H_l denote a non-linear transformation function that consists of ReLU, convolutional, and batch normalization (BN) layers. $[x_0, x_1, x_2, \dots, x_{l-1}]$ signifies coordinated output from layers 0 to layer $l - 1$. This structure typically comprises Dense-Block and Transition modules that utilize dense links and smaller parameters to mitigate computational complexity. The transition unit includes Pooling, BN, Convolution, and ReLU layers. This Transition component connects adjacent dense blocks and reduces the feature mapping size over the pooling layer, underscoring the significance of higher-level feature representation in improving compression efficacy. The DenseNet model contains 4 DenseBlock units and 3 Transition components.

Attention mechanism (AM) is a data processing model in ML, which is extensively utilized in different areas of DL recently. AMs are separated into mixed-domain, spatial, and channel attentions. During this work of channel attention, a novel framework was developed, concentrating on channel relations in CNNs and presented

a novel structural component named “Squeeze and Excitation” (SE) blocks, which dynamically adjusts the feature remarks about channels by mimicking their interdependence. They considered the nondimensionality-decreasing local cross-channel interaction tactic and an adaptive model to select the dimensions of 1D convolutional kernels, thereby achieving performance growth. This study utilizes SENet for learning global feature information and remarkably improves the main characteristics. Initially, input X is converted into feature U using the transformation function F_{tr} , where $X \in R^{h \times w \times c_1}$ and $U \in R^{h \times w \times c_2}$.

Then, the squeezing module F_{sq} utilizes the global average pooling to condense the feature U into $R^{1 \times 1 \times c_2}$, demonstrating the global supply of replies on the feature networks. Formerly, $F_{ex}(\bullet, w)$ creates weights for every feature channel utilizing parameter w . By re-weighting, the excitation output weight was determined as the significance of all feature channels. At last, the weights are utilized for the preceding feature channels to recalibrate the new features.

This study proposes an original network method, named SE-DenseNet for SLR, which primarily consists of four DenseBlocks, three Transitions, and three SENets. The input model is $H \in R^{T \times E \times C}$. To speed up convergence and prevent gradient vanishing issues, this work carried out either the activation function BN or the process after every 2D convolution. The component in the DenseNet model presents a hyperparameter named growth rate, which is assigned a value of 12. This parameter controls the channel counts added in all convolutional layers, allowing the system to get the balance between model performance and complexity. All DenseBlocks are made from numerous Bottleneck layers. Every Bottleneck is created from a ReLU, a 1×1 convolutional layer, a BN layer, a 3×3 convolutional layer, a ReLU, and a BN layer sequentially. The DenseNet network, containing 4 DenseBlock units and 3 Transition modules, is rejected by seven successive processes: ReLU, BN layer, BN layer, ReLU, 1×1 convolutional layer, Dropout layer, and 3×3 convolutional layer. The Dropout layer aims to prevent overfitting. Inserting SENet among Transition and DenseBlock for learning the significance of every channel and improving valuable performance.

ENN-based classification model

Moreover, the ENN model is employed for the SLR classification process²⁹. This model is chosen for its dynamic memory capability, which effectively captures temporal dependencies in sequential data, such as SL gestures. This technique comprises context units that retain data from prior time steps, making it appropriate for recognizing patterns over time, unlike feedforward networks. This is specifically beneficial in SLR, where gestures follow a temporal sequence. Compared to conventional classifiers such as SVM or basic CNNs, ENN presents an enhanced performance on time-series data without requiring intrinsic architectures. Its ability to model contextual information results in more accurate and consistent classification results. Figure 2 portrays the structure of the ENN technique.

ENNs are an ML approach that is designed for processing time-independent data. Unlike conventional feedforward NNs, ENNs have relations that make managed cycles, permitting them to maintain a model of sequential data efficiently.

The advanced ENN consists of four layers: the input layer, denoted as i , j , representing the hidden layer (HL), the context layer, specified as c , and the output layer, embodied as o . Every layer is linked utilizing weight. The i th layer is provided with the hydrologic inputs. The i th layer includes ten hidden neurons. The advanced

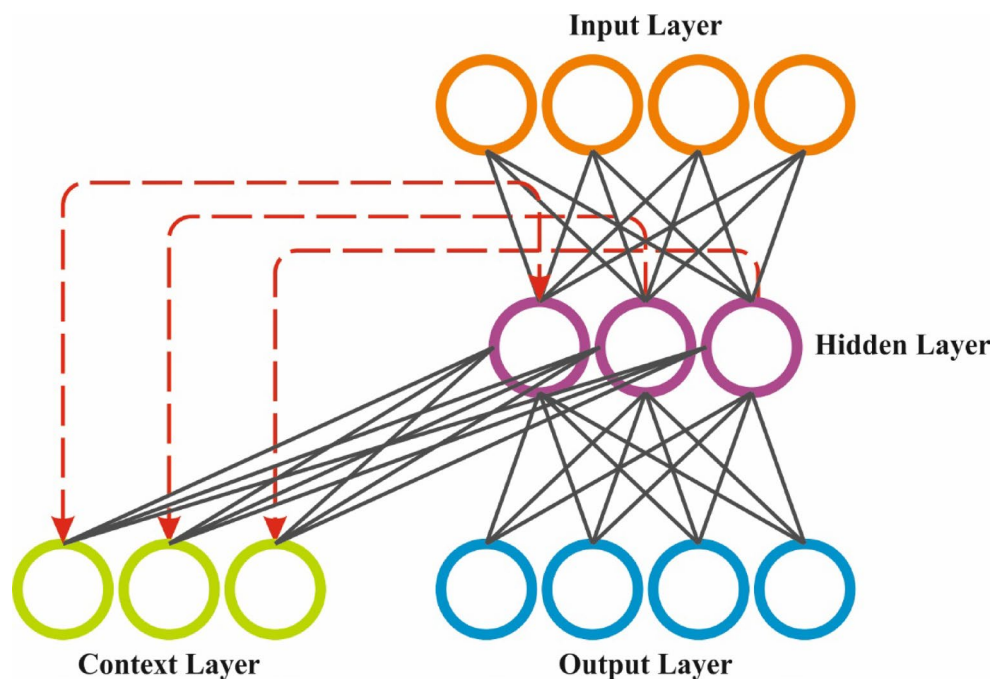


Fig. 2. Structure of the ENN model.

ENN method contains a c layer that is otherwise recognized as a layer of feedback. The objective of the c layer is to maintain the data from the previous stage, which assists in examining the patterns from the preceding data. The training and functional procedure are provided as shown.

The node in this layer and the input layer are specified as:

$$x_i^{(1)}(n) = f_i^{(1)}\left(\text{net}_i^{(1)}(n)\right) \quad (2)$$

Whereas $x_i^{(1)}(n)$ denotes the output data of the i th layer. The node in the i th layer is provided as:

$$x_j^{(2)}(n) = S\left(\text{net}_j^{(2)}(n)\right) \quad (3)$$

$$\text{net}_j^{(2)}(n) = \sum_i w_{ij} \times x_i^{(1)}(n) + \sum_k w_{kj} \times x_k^{(3)}(n) \quad j = 0, 1, 2, \dots, 9, k = 0, 1, 2, \dots, 9 \quad (4)$$

The function of the sigmoid, $S(x) = 1/(1 + e^{-x})$, was utilized in NNs for mapping input values to the range between (0,1) that might characterize possibilities. It was distinguishable and had a basic derivative,

$$S'(x) = S(x)(1 - S(x)) \quad (5)$$

The i th layer is linked utilizing the neuron with weightings w_{li} , and w_{ki} represents neuron weights. The nodes in these contextual layers are provided as:

$$x_k^{(3)}(n) = \alpha x_k^{(3)}(n-1) + x_j^{(2)}(n-1) \quad (6)$$

From Eq. (6), α refers to the gain of feedback that is located between $0 \leq \alpha \leq 1$. The node in the output layer was signified as:

$$\gamma_l^{(4)}(n) = f_l^{(4)}\left(\text{net}_l^{(4)}(n)\right) \quad (7)$$

$\gamma_l^{(4)}(n)$ Provides the forecast output of the presented method. The weighting update of the advanced ENN-based forecasting method occurs layer-to-layer; the weighted upgrade of linking neuron weight w_{il} is provided as:

$$w_{il}(n+1) = w_{il}(n) + \xi_1 \Delta w_{il} \quad (8)$$

In a weighted update, ξ_1 characterizes the rate of training of the zero layer. The novel weight of w_{ki} is provided as:

$$w_{kj}(n+1) = w_{kj}(n) + \xi_2 \Delta w_{kj} \quad (9)$$

In the weight update, ξ_2 embodies the rate of training of the i th layer. The novel weight of w_{ij} is specified as:

$$w_{ij}(n+1) = w_{ij}(n) + \xi_3 \Delta w_{ij} \quad (10)$$

In the weight update, ξ_3 refers to the rate of training of an input layer. The advanced ENN method was trained utilizing the backpropagation (BP) model, an expansion of the normal BP model applied in feedforward NNs. The BP methods consider the temporal dependences by describing the network over time and fine-tuning weights as a result.

PFA-based parameter tuning model

Ultimately, the parameter tuning process is conducted using the PFA model to enhance the classification performance of the ENN classifier³⁰. This model is chosen for its robust global search capability, fast convergence, and ability to avoid local optima during optimization. This technique is inspired by the collective movement of agents in a search space. Additionally, it demonstrates efficiency in balancing exploration and exploitation, making it ideal for fine-tuning intrinsic models, such as ENN. Compared to conventional methods such as grid search or other metaheuristics, namely PSO or GA, PFA illustrates better stability and solution quality in high-dimensional spaces. Its adaptability and computational efficiency improve the overall performance of the classification model. This results in a more accurate and reliable SLR.

PFA simulates the random behaviour and drive of the animal, which emulates its head to a neighbouring site in search of sustenance or prey. Modifications in a leader are probable while the goal of searching is not accomplished. The head of a group and its competitors collaborate to determine the most effective path to the destination. Depending on the direction and force in the multidimensional region, the path's direction is improved. At some point, the contestant in the optimum position is considered the swarm's head. This candidate is specified as the Pathfinder. During these existing iterations, Pathfinder and its location are viewed as the finest solution, and another competitor acquires it. A vector representing the movement position of competitors in multiple sizes is employed to manage the recommended solutions. To control how the rival performs in the exploration phase, four parameters are adjusted. Every cycle concurrently creates the vibration of competitor

ν and oscillating frequency τ . The attraction factor α fine-tunes the random area of separation, and the communication factor σ upholds the movement regarding the neighbouring competitor.

$$C(i + \Delta i) = C^0(i) \cdot d + E_f + K_J + \nu \quad (11)$$

The term C indicates the vector for a position, d signifies the identity vector, K_J specifies the force that is reliant on the position of the Pathfinder, i specifies the period, and E_f is the communication that arises between dual rivals C_k and c_f .

$$C_J(i + \Delta i) = C_J(i) + \Delta C + \tau \quad (12)$$

The term Δc represents the value assessed by deliberating the region among the dual diverse locations of the Pathfinder, and C_J is the vector position of the Pathfinder.

$$\vec{C}_f^{o+1} = \vec{C}_f^o + \vec{Q}_1 * \left(\vec{C}_k^o - \vec{C}_f^o \right) + \vec{Q}_2 * \left(\vec{C}_j^o - \vec{C}_f^o \right) + \nu \quad (13)$$

The terms \vec{Q}_1 and \vec{Q}_2 are dual vectors of the trajectory in arbitrary coordinates. The value of $\vec{Q}_1 = \alpha \cdot q_1$ and $\vec{Q}_2 = \sigma \cdot q_2$, where q_1 and q_2 indicate the arbitrary movement created homogeneously. The values of q_1

and q_2 range from $(-1, 1)$. The term \vec{C}_f^o and \vec{C}_k^o the vector position of dual rival f and k at the existing iteration 0. The value of ν is described.

$$\nu = \left[1 - \left(\frac{o}{O} \right) \right] p_1 \cdot N_{os}; N_{os} = \| C_f - C_k \| \quad (14)$$

The term O is the suggested maximum number of iterations, 0 specifies the existing iteration, and N_{os} is the separation distance between the dual rivals. The factor of attraction α and the factor of communication σ values are altered. Every rival moves randomly and independently within the region, whereas σ and α equal 0. Every rival stop moving and loses the path of the swarm's head while σ and α are equal to ∞ . If α and σ are both lower than 1 and higher than 2, then an affiliate rival cannot generate an optimum solution. Thus, it is significant that the value of α and σ must be (1, 2).

$$\vec{C}_j^{o+1} = \vec{C}_j^o + 2q_3 \cdot \left(\vec{C}_j^o - \vec{C}_j^{o-1} \right) + \tau \quad (15)$$

The term q_3 represents an arbitrary vector of rivals. Whether the terms $\vec{Q}_1 * \left(\vec{C}_k^o - \vec{C}_f^o \right)$ and $\vec{Q}_2 * \left(\vec{C}_j^o - \vec{C}_f^o \right)$ in Eq. (11) or the term $2q_3 \cdot \left(\vec{C}_j^o - \vec{C}_j^{o-1} \right)$ become zero, subsequently τ and ν can randomly move each rival with proper values through various paths. The oscillating frequency τ is calculated.

$$\tau = p_2 \cdot \exp \left(\frac{-20}{O} \right) \quad (16)$$

The term p_2 signifies an arbitrary value within $(-1, 1)$. The convergence and divergence of PFA are derived from the values τ and ν . It can accelerate or slow down the technique. To accomplish this without diverging among them in every iteration, values ν and τ must be (1, 2). The contestant can quickly leave their locations without discovering a solution if either p_1 or p_2 is beyond the range $[-1, 1]$.

The PFA model generates a fitness function (FF) for achieving improved classification performance. It describes a positive number to describe the better efficiency of the candidate solution. Here, the classification rate of error reduction is designated as FF, as defined in Eq. (16).

$$\begin{aligned} \text{fitness}(x_i) &= \text{ClassifierErrorRate}(x_i) \\ &= \frac{\text{no. of misclassified samples}}{\text{Total no. of samples}} \times 100 \end{aligned} \quad (17)$$

Proposed methodology

The performance evaluation of the PASLR-DDPFEM technique is examined under the ASL dataset³¹. The method runs on Python 3.6.5 with an Intel Core i5-8600 K CPU, 4GB GPU, 16GB RAM, 250GB SSD, and 1 TB HDD, using a 0.01 learning rate, ReLU activation, 50 epochs, 0.5 dropout, and a batch size of 5. The chosen dataset includes signs performed under varied conditions such as diverse hand positions, lighting, and

Class Labels	<i>Accu_y</i>	<i>Prec_n</i>	<i>Sens_y</i>	<i>Spec_y</i>	<i>F1_{Score}</i>	Kappa
TRAPA (70%)						
L-1	98.64	82.34	82.34	99.29	82.34	82.40
L-2	98.85	84.69	85.74	99.38	85.21	85.27
L-3	98.79	83.41	85.40	99.32	84.39	84.44
L-4	98.82	84.58	84.54	99.38	84.56	84.63
L-5	98.91	85.34	86.00	99.42	85.67	85.75
L-6	98.87	85.80	85.72	99.41	85.76	85.83
L-7	98.90	86.09	85.73	99.44	85.91	85.96
L-8	98.90	84.81	86.72	99.38	85.75	85.80
L-9	98.71	82.77	82.40	99.34	82.59	82.65
L-10	98.76	83.21	84.76	99.32	83.98	84.05
L-11	98.70	85.01	80.62	99.43	82.75	82.80
L-12	98.86	85.78	83.85	99.45	84.80	84.87
L-13	98.78	83.60	85.41	99.32	84.50	84.58
L-14	98.65	83.64	80.98	99.36	82.28	82.36
L-15	98.71	83.83	81.63	99.38	82.72	82.78
L-16	98.85	84.53	86.05	99.37	85.28	85.35
L-17	98.84	83.48	87.61	99.30	85.49	85.55
L-18	98.88	86.23	84.64	99.46	85.43	85.51
L-19	98.89	87.00	83.52	99.50	85.23	85.29
L-20	98.60	82.10	80.96	99.30	81.52	81.58
L-21	98.80	84.32	85.15	99.36	84.74	84.80
L-22	98.81	84.12	85.24	99.36	84.67	84.73
L-23	98.96	86.27	86.61	99.45	86.44	86.51
L-24	98.87	86.84	83.02	99.50	84.89	84.94
L-25	98.84	84.66	85.11	99.38	84.88	84.96
L-26	98.65	81.00	85.11	99.20	83.01	83.07
Average	98.80	84.44	84.42	99.38	84.42	84.48
TESPA (30%)						
L-1	98.81	84.51	84.23	99.39	84.37	84.43
L-2	98.81	83.37	86.16	99.32	84.74	84.82
L-3	98.79	85.16	83.19	99.42	84.16	84.22
L-4	98.70	82.05	84.96	99.25	83.48	83.54
L-5	98.96	86.71	87.18	99.45	86.94	87.00
L-6	98.97	84.23	87.56	99.39	85.87	85.94
L-7	98.95	86.46	85.17	99.49	85.81	85.88
L-8	98.82	83.89	86.11	99.33	84.98	85.04
L-9	98.66	85.15	82.19	99.38	83.65	83.71
L-10	98.63	80.39	85.45	99.16	82.84	82.90
L-11	98.66	83.82	80.34	99.39	82.04	82.11
L-12	98.82	85.45	84.24	99.41	84.84	84.90
L-13	98.70	81.83	83.89	99.28	82.84	82.90
L-14	98.88	86.19	83.97	99.47	85.07	85.12
L-15	98.67	84.09	82.48	99.35	83.28	83.34
L-16	98.78	83.44	84.75	99.33	84.09	84.16
L-17	98.74	83.18	83.37	99.34	83.28	83.34
L-18	99.00	87.02	86.53	99.49	86.78	86.84
L-19	98.88	85.99	84.56	99.45	85.27	85.33
L-20	98.48	80.07	81.21	99.18	80.63	80.71
L-21	98.75	83.28	82.89	99.36	83.08	83.15
L-22	98.76	84.12	83.56	99.37	83.84	83.90
Continued						

Class Labels	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	$F1_{Score}$	Kappa
L-23	99.03	88.76	86.15	99.56	87.44	87.51
L-24	98.94	88.12	84.05	99.54	86.04	86.10
L-25	98.83	84.88	84.97	99.39	84.93	85.00
L-26	98.50	79.33	81.82	99.16	80.55	80.62
Average	98.79	84.29	84.27	99.37	84.26	84.33

Table 3. Overall SLR outcome of PASLR-DDPFEM model at 70%TRAPA and 30%TESPA.

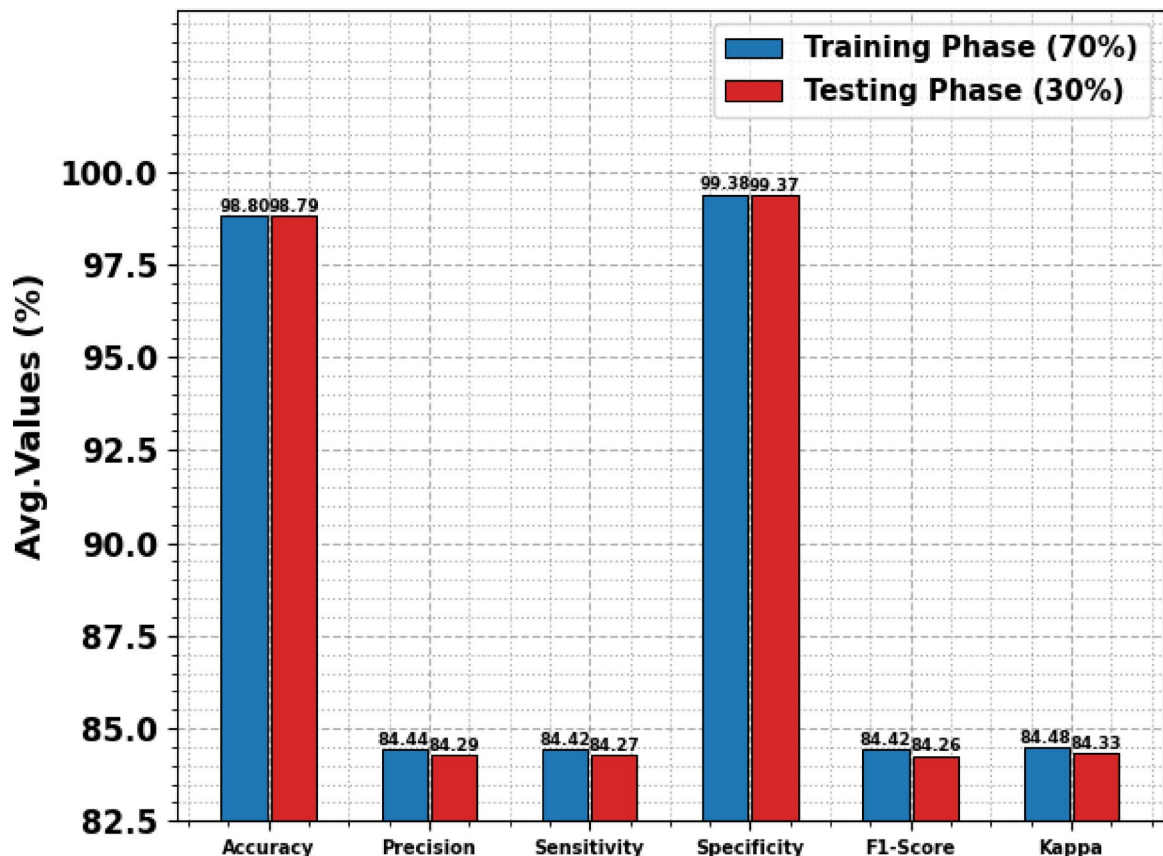


Fig. 3. Average outcome of PASLR-DDPFEM model at 30%TESPA.

backgrounds, assisting the dataset capture some real-world variation in gesture appearance for robust model training and evaluation.

In Table 3; Fig. 3, a brief overview of the overall SLR outcome for the PASLR-DDPFEM approach is presented, covering 70% of the training phase (TRAPA) and 30% of the testing phase (TESPA). The tabulated values indicate that the PASLR-DDPFEM methodology accurately identifies the 26 samples. The results suggest that the PASLR-DDPFEM approach can effectively recognize the samples. For under 70% of TRAPA, the PASLR-DDPFEM method obtains an average $accu_y$, $prec_n$, $sens_y$, $spec_y$, $F1_{score}$, and Kappa of 98.80%, 84.44%, 84.42%, 99.38%, 84.42%, and 84.48%, respectively. Likewise, under 30% of TESPA, the PASLR-DDPFEM method obtains an average $accu_y$, $prec_n$, $sens_y$, $spec_y$, $F1_{score}$, and Kappa of 98.79%, 84.29%, 84.27%, 99.37%, 84.26%, and 84.33%, respectively.

In Fig. 4, the TRA $accu_y$ (TRAAY) and validation $accu_y$ (VLAAY) analysis of the PASLR-DDPFEM technique is illustrated. The figure highlights that the TRAAY and VLAAY values exhibit a rising trend, indicating the model's ability to achieve higher performance over various iterations. Additionally, the TRAAY and VLAAY remain closer throughout an epoch, which results in minimal overfitting and optimal performance of the PASLR-DDPFEM technique.

In Fig. 5, the TRA loss (TRALO) and VLA loss (VLALO) curve of the PASLR-DDPFEM approach is displayed. The TRALO and VLALO analyses exemplify a decreasing trend, indicating the capacity of the PASLR-DDPFEM approach in balancing trade-offs. The constant decrease also guarantees the enhanced performance of the PASLR-DDPFEM model.

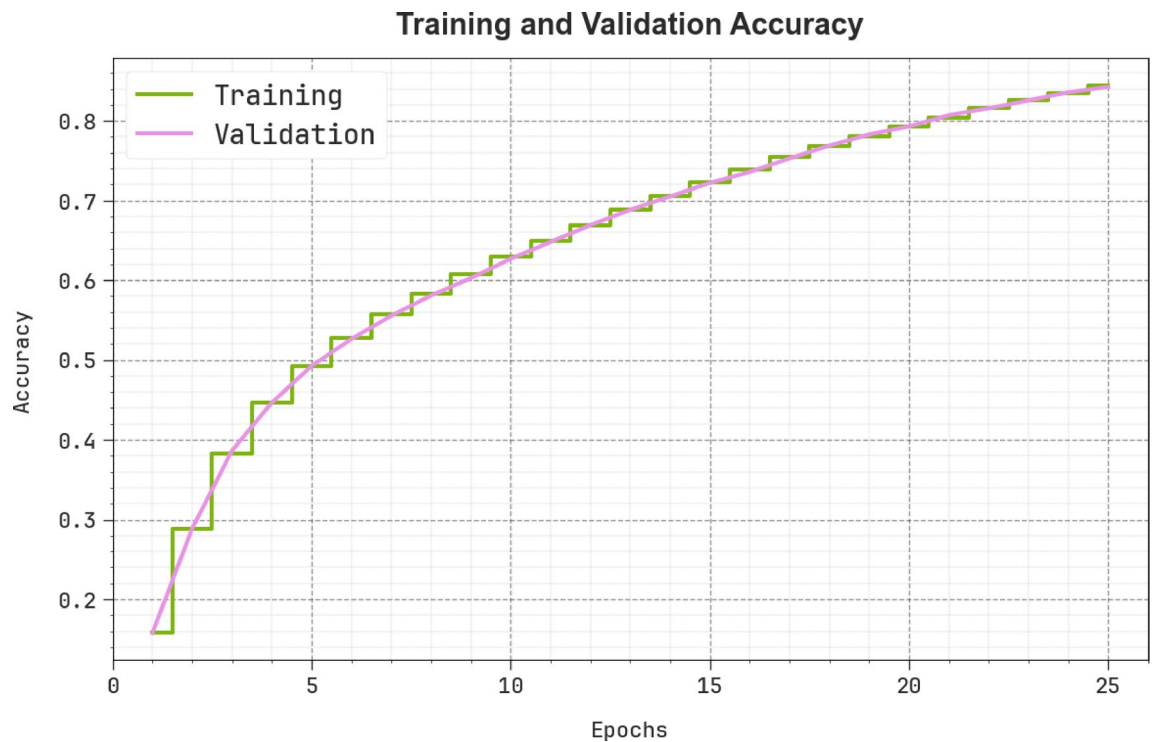


Fig. 4. $Accu_y$ curve of the PASLR-DDPFEM method.

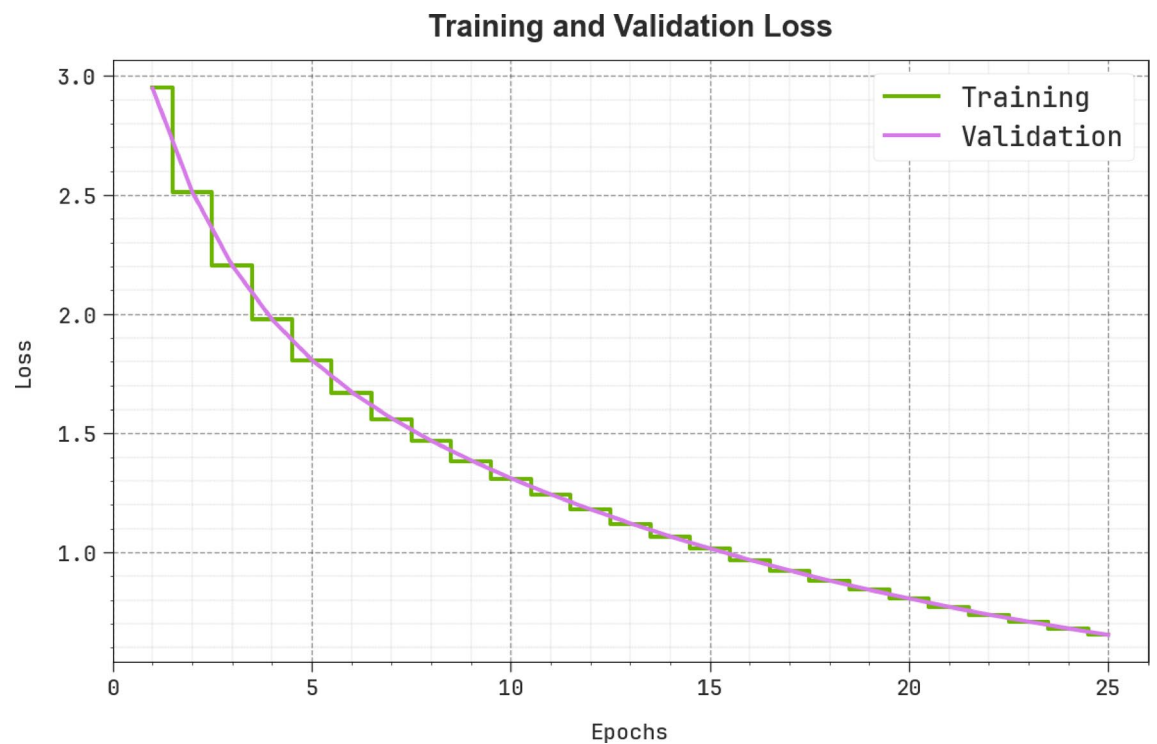


Fig. 5. Loss curve of the PASLR-DDPFEM method.

In Fig. 6, the PR graph analysis of the PASLR-DDPFEM methodology provides clarification into its results by plotting Precision beside Recall for every class label. The steady rise in PR values across all class labels depicts the efficacy of the PASLR-DDPFEM approach in the classification process.

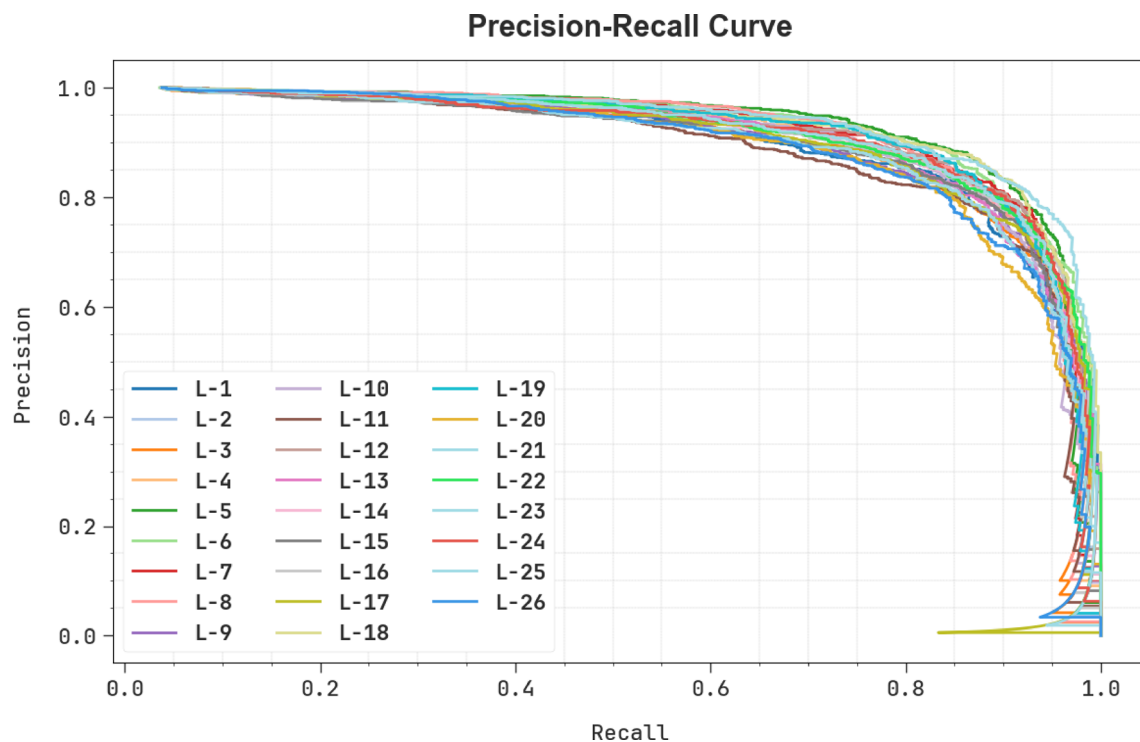


Fig. 6. PR curve of the PASLR-DDPFEM model.

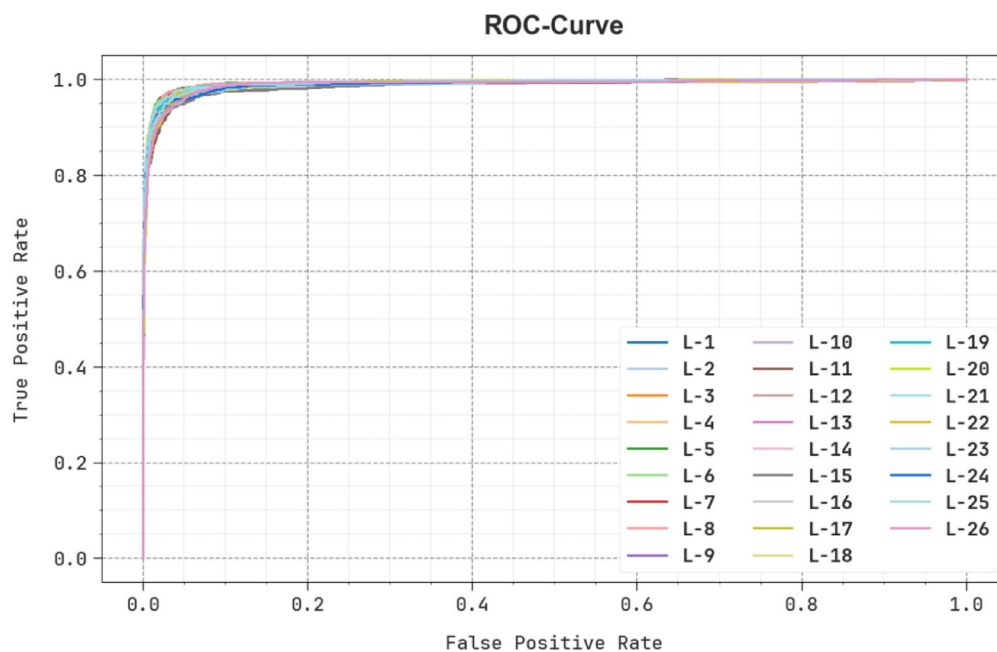


Fig. 7. ROC curve of the PASLR-DDPFEM model.

In Fig. 7, the ROC analysis of the PASLR-DDPFEM approach is examined. The results suggest that the PASLR-DDPFEM technique achieves optimal ROC results across all classes, effectively representing the vital capacity to distinguish between class labels. This dependable tendency of better values of ROC across several class labels signifies the proficient efficiency of the PASLR-DDPFEM technique on predicting class labels, highlighting the classification procedure.

To demonstrate the proficiency of the PASLR-DDPFEM technique, a comprehensive comparison study is presented in Table 4^{32,33}.

Methodology	<i>Accu_y</i>	<i>Prec_n</i>	<i>Sens_y</i>	<i>Spec_y</i>	<i>F1_{score}</i>
CNN Classifier	95.54	81.69	76.33	93.95	75.21
VGG16 Method	89.00	81.59	77.28	90.94	81.33
EfficientNet V2	86.92	75.28	81.09	95.38	76.67
MobileNetV2	88.55	82.73	84.00	99.12	77.20
SignLan-Net	84.72	77.94	81.80	96.63	75.86
Faster R-CNN	95.87	78.55	76.58	97.60	77.49
Inception V3	91.66	81.84	83.95	92.09	78.24
PASLR-DDPFEM	98.80	84.44	84.42	99.38	84.42

Table 4. Comparative study of the PASLR-DDPFEM technique with existing models.

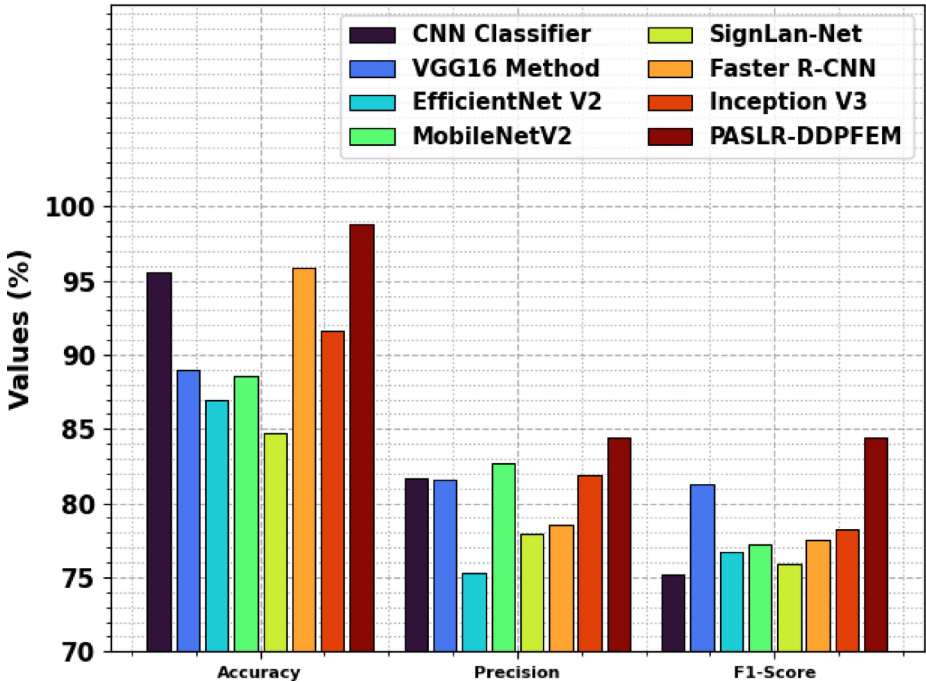


Fig. 8. *Accu_y*, *prec_n*, and *F1_{score}* outcome of PASLR-DDPFEM technique with existing models.

In Fig. 8, a comparative *accu_y*, *prec_n*, and *F1_{score}* results of the PASLR-DDPFEM technique are provided. The results indicate that the SignLan-Net, EfficientNet V2, and MobileNetV2 methodologies have shown worse values of *accu_y*, *prec_n*, and *F1_{score}*. At the same time, the VGG16 and Inception V3 methods have achieved slightly maximal *accu_y*, *prec_n*, and *F1_{score}*. Meanwhile, the Faster R-CNN and CNN methodologies have established closer values of *accu_y*, *prec_n*, and *F1_{score}*. However, the PASLR-DDPFEM approach results in optimal performance with *accu_y*, *prec_n*, and *F1_{score}* of 98.80%, 84.44%, and 84.42%, respectively.

In Fig. 9, a comparative *sens_y* and *spec_y* results of the PASLR-DDPFEM approach are provided. The results indicate that the CNN, Faster R-CNN, and VGG16 techniques have shown lower values of *sens_y* and *spec_y*. At the same time, the EfficientNet V2 and SignLan-Net approaches have achieved slightly maximum *sens_y* and *spec_y*. Meanwhile, the Inception V3 and MobileNetV2 techniques have established closer values of *sens_y* and *spec_y*. On the other hand, the PASLR-DDPFEM model results in superior performance, with *sens_y* and *spec_y* of 84.42% and 99.38%, respectively.

Table 5; Fig. 10 present the computational time (CT) analysis of the PASLR-DDPFEM approach compared to existing methods. The CT clearly demonstrates the efficiency of the PASLR-DDPFEM approach, which records the lowest CT of 6.34 s among all evaluated models. In contrast, conventional models like the CNN and VGG16 require 22.35 and 21.78 s, respectively, reflecting significantly higher CTs. EfficientNet V2 and Faster R-CNN exhibit enhanced performance with CT values of 11.19 and 12.62 s, while MobileNetV2 and Inception V3 require 20.81 and 19.78 s. SignLan-Net performs well with 9.95 s, yet the PASLR-DDPFEM method outperforms all others, presenting a reduction of over 70% in CT compared to the highest value, making it highly appropriate for real-time applications.

Table 6; Fig. 11 present the error analysis of the PASLR-DDPFEM methodology in comparison to existing models. The evaluation results indicate that the PASLR-DDPFEM methodology, with an *accu_y* of 1.20%, *prec_n* of 15.56%, *sens_y* of 15.58%, *spec_y* of 0.62%, and *F1_{score}* of 15.58%, illustrates relatively lower performance

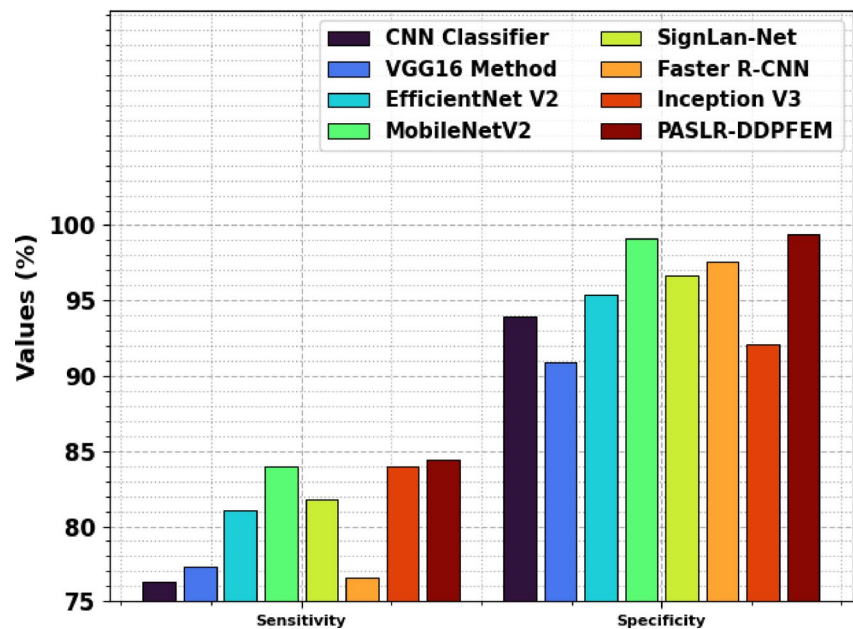


Fig. 9. *Sens_y* and *spec_y* outcome of the PASLR-DDPFEM technique with existing models.

Methodology	CT (sec)
CNN Classifier	22.35
VGG16 Method	21.78
EfficientNet V2	11.19
MobileNetV2	20.81
SignLan-Net	9.95
Faster R-CNN	12.62
Inception V3	19.78
PASLR-DDPFEM	6.34

Table 5. CT analysis of the PASLR-DDPFEM approach with existing methods.

compared to other approaches. For instance, SignLan-Net demonstrates the highest *accu_y* of 15.28% and a robust *F1_{score}* of 24.14%, while EfficientNet V2 achieves an *accu_y* of 13.08% and a *prec_n* of 24.72%. Although the PASLR-DDPFEM model maintains a consistent balance between *prec_n* and *sens_y*, its overall *accu_y* and *spec_y* remain limited. This suggests that the model may struggle to distinguish between certain sign classes effectively, and further optimization may be necessary. The error analysis highlights the importance of deeper feature learning and better class representation to improve recognition results.

Table 7 depicts the ablation study of the PASLR-DDPFEM technique. Without PFA tuning, the ENN with GF preprocessing and SE-DenseNet feature extraction attained an *accu_y* of 98.18%, *prec_n* of 83.79%, *sens_y* of 83.65%, *spec_y* of 98.77%, and *F1_{score}* of 83.71%. By combining the PFA tuning resulted with an *accu_y* of 98.80%, *prec_n* of 84.44%, *sens_y* of 84.42%, *spec_y* of 99.38%, and *F1_{score}* of 84.42%, highlighting the efficiency of the tuning process in improving the overall performance.

Table 8 illustrates the computational efficiency of diverse object detection models³⁴. The PASLR-DDPFEM methodology exhibits the lowest Floating-Point Operations (FLOPs) at 4.09 G, minimal Graphics Processing Unit (GPU) memory usage of 589 MB, and the fastest inference time of 1.07 s, significantly outperforming YOLOv3-tiny-T, ShuffleNetv2-YOLOv3, YOLOv5I, and YOLOv7 in terms of both speed and resource efficiency.

Conclusion

This paper designs and develops a PASLR-DDPFEM model. The aim is to enhance SLR techniques to help individuals with hearing challenges communicate with others. Initially, the image pre-processing phase is performed using the GF model to improve image quality by removing noise. Furthermore, the PASLR-DDPFEM method is employed by the SE-DenseNet model for the feature extraction process. Moreover, the ENN method is used for the SLR classification process. Finally, the parameter tuning process is performed through PFA to improve the classification performance of the ENN classifier. An extensive set of simulations of the PASLR-DDPFEM method is accomplished under the American SL (ASL) dataset. The comparison study of the PASLR-DDPFEM method revealed a superior accuracy value of 98.80% compared to existing models. This method can

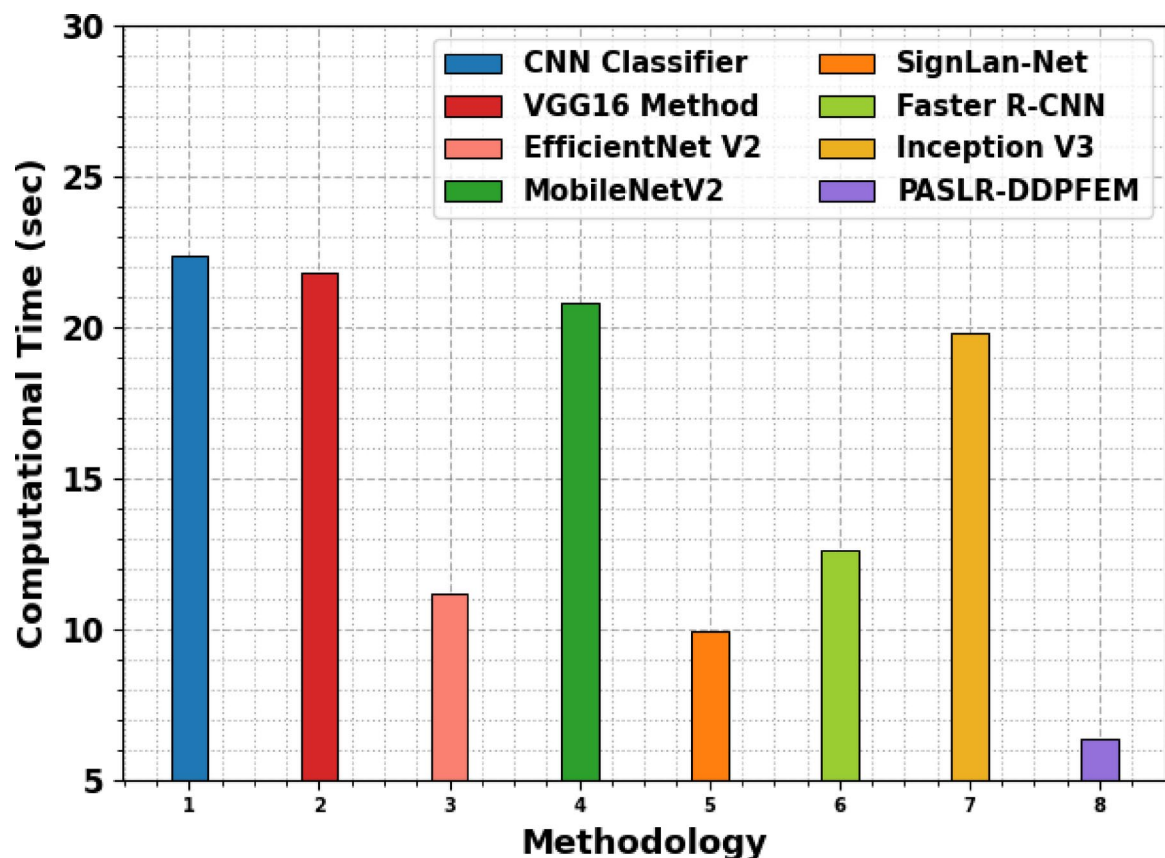


Fig. 10. CT analysis of the PASLR-DDPFEM approach with existing methods.

Methodology	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	$F1_{Score}$
CNN Classifier	4.46	18.31	23.67	6.05	24.79
VGG16 Method	11.00	18.41	22.72	9.06	18.67
EfficientNet V2	13.08	24.72	18.91	4.62	23.33
MobileNetV2	11.45	17.27	16.00	0.88	22.80
SignLan-Net	15.28	22.06	18.20	3.37	24.14
Faster R-CNN	4.13	21.45	23.42	2.40	22.51
Inception V3	8.34	18.16	16.05	7.91	21.76
PASLR-DDPFEM	1.20	15.56	15.58	0.62	15.58

Table 6. Error analysis of the PASLR-DDPFEM methodology with existing models.

be deployed in real-time applications due to optimized feature extraction and tuning, enabling fast and accurate gesture recognition suitable for mobile and embedded devices. The limitations of the PASLR-DDPFEM method comprise the dependence on a single dataset, which may restrict the generalizability of the findings across diverse SL discrepancies and real-world conditions. Furthermore, the model's performance could be affected by varying lighting conditions and complex backgrounds that were not extensively addressed. The existing approach also fails to integrate multimodal inputs, such as depth or motion data, which could enhance recognition accuracy. Computational needs, although optimized, may still pose threats for deployment on low-resource devices. Furthermore, user-specific adaptability and personalized learning were not explored. Addressing these limitations could open new avenues to improve robustness, inclusivity, and practical applicability in future research. Also, integrating multimodal cues such as facial expressions and lip movements could additionally improve the accuracy of recognition by providing further contextual data to discriminate similar gestures.

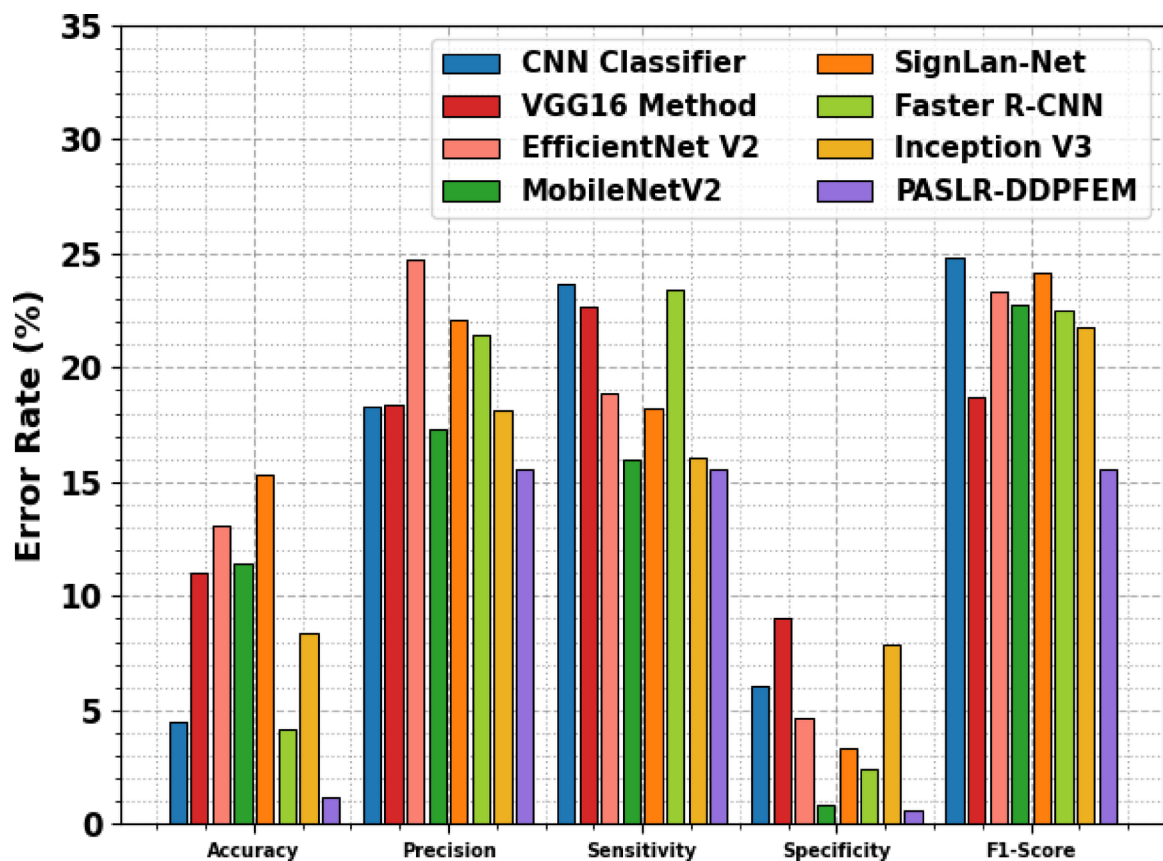


Fig. 11. Error analysis of the PASLR-DDPFEM methodology with existing models.

Methodology	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	$F1_{Score}$
ENN + GF + SE-Densenet (Without PFA Hyperparameter tuning process)	98.18	83.79	83.65	98.77	83.71
PASLR-DDPFEM (ENN with GF preprocessing and SE-DenseNet feature extraction and PFA hyperparameter tuning process)	98.80	84.44	84.42	99.38	84.42

Table 7. Comparative performance evaluation of the PASLR-DDPFEM technique through ablation study against existing models.

Methods	FLOPs (G)	GPU (M)	Inference Time (sec)
YOLOv3-tiny-T	7.20	2849	6.21
ShuffleNetv2-YOLOv3	42.50	2837	2.74
YOLOv5I	108.20	3489	6.55
YOLOv7	105.00	2917	4.62
PASLR-DDPFEM	4.09	589	1.07

Table 8. Computational efficiency comparison of the PASLR-DDPFEM technique in terms of FLOPs, GPU, and inference time.

Data availability

The data that support the findings of this study are openly available at [<https://www.kaggle.com/datasets/grassknotted/asl-alphabet>] (<https://www.kaggle.com/datasets/grassknotted/asl-alphabet>), reference number [32].

Received: 4 June 2025; Accepted: 26 December 2025

Published online: 29 December 2025

References

1. Wadhawan, A. & Kumar, P. Deep learning-based sign Language recognition system for static signs. *Neural Comput. Appl.* **32** (12), 7957–7968 (2020).
2. Cheok, M. J., Omar, Z. & Jaward, M. H. A review of hand gesture and sign Language recognition techniques. *Int. J. Mach. Learn. Cybernet.* **10**, 131–153 (2019).
3. Adaloglou, N. et al. A comprehensive study on deep learning-based methods for sign Language recognition. *IEEE Trans. Multimedia.* **24**, 1750–1762 (2021).
4. Rastgoo, R., Kiani, K. & Escalera, S. Hand sign language recognition using multi-view hand skeleton. *Expert Systems with Applications*, **150**, p.113336. (2020).
5. Katoch, S., Singh, V. & Tiwary, U. S. Indian Sign Language recognition system using SURF with SVM and CNN. *Array*, **14**, p.100141. (2022).
6. Lee, C. K. et al. American Sign Language recognition and training method with recurrent neural network. *Expert Systems with Applications*, **167** p.114403. (2021).
7. Sharma, S. & Singh, S. Vision-based hand gesture recognition using deep learning for the interpretation of sign language. *Expert Systems with Applications*, **182**, p.115657. (2021).
8. Al-Hammadi, M. et al. Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. *Ieee Access*, **8**, pp.192527–192542. (2020).
9. Camgoz, N. C., Koller, O., Hadfield, S. & Bowden, R. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10023–10033). (2020).
10. Abdel-Jaber, H., Devassy, D., Al Salam, A., Hidaytallah, L. & El-Amir, M. A review of deep learning algorithms and their applications in healthcare. *Algorithms*, **15**(2), p.71. (2022).
11. Rethick, R. S. et al. February. Deep learning-driven recognition and categorization of hand gestures for individuals with hearing and speech disabilities. In *AIP Conference Proceedings* (Vol. 3204, No. 1). AIP Publishing. (2025).
12. Assiri, M. S. & Selim, M. M. A swin transformer-driven framework for gesture recognition to assist hearing impaired people by integrating deep learning with secretary bird optimization algorithm. *Ain Shams Engineering Journal*, **16**(6), p.103383. (2025).
13. Kumar, M. M., Reddy, P. D. K. & Swetha, P. National sign Language real-time recognition to speech and text for deaf and dumb patient interaction. In *Recent Trends in Healthcare Innovation* (292–300). CRC. (2025).
14. Harshini, P. J., Rahman, M. A., Raj, J. A. & Durgadevi, P. May. Sign Language Recognition System for Seamless Human-AI Interaction. In *International Research Conference on Computing Technologies for Sustainable Development* (pp. 124–144). Cham: Springer Nature Switzerland. (2024).
15. Allehaibi, K. H. Artificial intelligence based automated sign gesture recognition solutions for visually challenged people. *Journal Intell. Syst. & Internet Things*, **14**(2). (2025).
16. Kumar, K. D., Ragul, K., Kumar, G. P. P. & Kumar, G. K. April. Enhancing Sign Language Recognition through Deep CNN and Handcrafted Features. In *2024 2nd International Conference on Networking and Communications (ICNWC)* (pp. 1–6). IEEE. (2024).
17. Hariharan, U. et al. Recognition of American sign Language using modified deep residual CNN with modified canny edge segmentation. *Multimedia Tools Applications*, pp.1–28. (2025).
18. Almjalj, A. & Almukadi, W. S. Deep computer vision with artificial intelligence based sign language recognition to assist hearing and speech-impaired individuals. *Scientific Reports*, **15**(1), p.32268. (2025).
19. Kaur, B., Chaudhary, A., Bano, S., Yashmita, Reddy, S. R. N. & Anand, R. Fostering inclusivity through effective communication: Real-time sign Language to speech conversion system for the deaf and hard-of-hearing community. *Multimedia Tools Appl.* **83** (15), 45859–45880 (2024).
20. Almjalj, A., Algamdi, S. A., Aljohani, N. & Nour, M. K. Harnessing attention-driven hybrid deep learning with combined feature representation for precise sign language recognition to aid deaf and speech-impaired people. *Scientific Reports*, **15**(1), p.32255. (2025).
21. Jagdish, M. & Raju, V. Multimodal Sign Language Recognition System: Integrating Image Processing and Deep Learning for Enhanced Communication Accessibility. *International Journal of Performability Engineering*, **20** (5), p.271. (2024).
22. Maashi, M., Iskandar, H. G. & Rizwanullah, M. IoT-driven smart assistive communication system for the hearing impaired with hybrid deep learning models for sign language recognition. *Scientific Reports*, **15**(1), p.6192. (2025).
23. Ilakkia, A., Kanchanadevi, P., Abirami, A., Rajan, R. & Muthukumaran, K. December. Sign Comm: A Real-Time Indian Sign Language Recognition System Using Deep Learning for inclusive communication. In *2024 International Conference on Emerging Research in Computational Science (ICERCS)* (pp. 1–8). IEEE. (2024).
24. Mosleh, M. A., Mohammed, A. A., Esmail, E. E., Mohammed, R. A. & Almuahaya, B. Hybrid Deep Learning and Fuzzy Matching for Real-Time Bidirectional Arabic Sign Language Translation: Toward Inclusive Communication Technologies. *IEEE Access*. (2025).
25. Thakkar, R., Kittur, P. & Munshi, A. June. Deep Learning Based Sign Language Gesture Recognition with Multilingual Translation. In *2024 International Conference on Computer, Electronics, Electrical Engineering & their Applications (IC2E3)* (pp. 1–6). IEEE. (2024).
26. Dhaarini, G., Sanjai, R. & Sandosh, S. May. Real-Time Sign Language Detection and Assistive System Using YOLOv10 for Enhanced Communication and Learning. In *2025 7th International Conference on Energy, Power and Environment (ICEPE)* (pp. 1–6). IEEE. (2025).
27. Choudhari, J., Bhagwat, S., Kundale, J. & Raorane, A. December. Real-Time Sign Language Recognition and Communication: Leveraging CNN for Deaf and Mute Communities. In *2024 International Conference on Innovation and Novelty in Engineering and Technology (INNOVA)* (Vol. 1, pp. 1–6). IEEE. (2024).
28. Xiang, Y., Zheng, W., Tang, J., Dong, Y. & Pang, Y. Gesture recognition from surface electromyography signals based on the SE-DenseNet network. *Biomedical Engineering/Biomedizinische Technik*, (0). (2025).
29. Wang, L. & Wang, L. Analysis and prediction of landslide deformation in water environment based on machine algorithm. *Hydrology Research*, p.nh2025098. (2025).
30. Sundaramoorthy, R. A. et al. Implementing heuristic-based multiscale depth-wise separable adaptive temporal convolutional network for ambient air quality prediction using real time data. *Scientific Reports*. **14** (1), p.18437. (2024).
31. <https://www.kaggle.com/datasets/grassknoted/asl-alphabet>
32. Kothadiya, D. R., Bhatt, C. M., Rehman, A., Alamri, F. S. & Saba, T. SignExplainer: an explainable AI-enabled framework for sign Language recognition with ensemble learning. *IEEE Access*. **11**, 47410–47419 (2023).
33. Nareshkumar, M. D. & Jaison, B. A Light-Weight deep Learning-Based architecture for sign Language classification. *Intelligent Autom. & Soft Computing*, **35**(3). (2023).
34. Chen, R. & Tian, X. Gesture detection and recognition based on object detection in complex background. *Applied Sciences*. **13** (7), p.4480. (2023).

Acknowledgements

The authors extend their appreciation to the King Salman center For Disability Research for funding this work through Research Group no KSRG-2024- 438.

Author contributions

Nadhem NEMRI: Conceptualization, methodology, validation, investigation, writing—original draft preparation, Mohammed Yahya Alzahrani: Conceptualization, methodology, writing—original draft preparation, writing—review and editing Wided Bouchelligua: methodology, validation, writing—original draft preparation Amani A Alneil: software, visualization, validation, data curation, writing—review and editing.

Funding

The authors extend their appreciation to the King Salman Centre for Disability Research for funding this work through Research Group no. KSRG-2024- 438.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to N.N.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025