CIT. Journal of Computing and Information Technology, Vol. 34, No. 2, June 2026, 133–146
doi: 10.20532/cit.2026.1006399


# Convolutional Neural Network with Class-Dependent Label Smoothing Method for Indian Sign Language Recognition

Keerthi Kumar Masanasetty and Parameshachari Bidare Divakarachari

Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Nitte (Deemed University), Bengaluru, India

Sign language is a significant way of communication for individuals with hearing impairments. To bridge a communication gap between the hearing impaired and the general public, it is important to recognize a sign accurately. The recognition of sign language using hand gesture images is a challenging task due to similar gesture images in different classes, which minimizes recognition performance. The Convolutional Neural Network (CNN) with the Class Dependent Label Smoothing (CDLS) technique is being developed to recognize the Indian Sign Language by hand gesture images. The CDLS is a regularization technique that allows the method to assign smoothing values based on the characteristics of gestures in various classes and enhances the CNN performance for sign language recognition. The ResNet-50-based feature extraction technique is utilized to extract deep features of gestures to identify variant gestures in different classes. The CNN with the CDLS technique reaches 99.6% accuracy, 0.99% precision, 0.99% recall, and a 0.99% F1-score on the Indian Sign Language (ISL) dataset when compared to CNN1.

*ACM CCS (2012) Classification:* Computing methodologies → Machine learning → Machine learning approaches

Computing methodologies → Artificial intelligence → Computer vision

*Keywords*: class-dependent label smoothing, convolutional neural network, Indian sign language, hand gesture, regularization

## 1. Introduction

Hand gestures act as a form of communication, developing a well-structured and organized language that facilitates efficient communication with impaired individuals [1]. Sign language consists of different gesture-generation methods and is a more efficient communication technique compared to the identification of lip movement [2-4]. Sign language is extensive and includes various gestures to comprehend messages properly; it is not just gestures utilizing the palms and fingers but also includes visual cues by the eyes and face [5, 6]. Additional components, such as facial expressions, involve expressing difficult meanings, which is more creative than the general verbal language [7]. Sign language is structured and simple, offering an effective communication method because of its formal structure [8]. The recognition of sign language as a research field includes matching patterns, Natural Language Processing (NLP), computer vision, Deep Learning (DL), and a design approach for identifying the sign language [9]. This is extended for human-computer interaction without the interface of voice. The system incorporates concepts from multiple disciplines and uses different algorithms, developing an integral part of the sign language recognition system [10]. The Indian Sign Language (ISL) includes various types of posture variability that makes recognition challenging, and hence, the recordings of appropriate postures are needed to frame the dataset [11]. Many researchers process sign language recognition, depending on Machine Learning (ML), which offers less accuracy due to the unavailability of automatic feature extraction [12]. Deep Learning (DL)-based algorithms address this issue in

various areas and show better performance than conventional algorithms in NLP, computer vision, and other areas of Artificial Intelligence (AI) [13]. The DL-based algorithms are developed for sign language recognition, offering enhanced performance compared to conventional methods, which improves the accuracy and effectiveness of recognition systems. The recognition of sign language using the hand gesture images is a challenging task due to similar gesture images in different classes, which minimizes the recognition performance [14]. Gangrade and Bharti [15] suggested an approach for the detection and segmentation of the hand region from the depth images utilizing a Microsoft Kinect sensor. The suggested approach performed well in cluttered environments, such as when the hand overlaps the face. A Convolutional Neural Network (CNN) was employed for the automatic extraction of features from the ISL signs, and these features were invariant to scaling and rotation. The suggested method was invariant to lighting conditions and effectively mitigated the hands' occlusion issue. However, the suggested method had difficulty in predicting the finger and hand postures accurately.

Kothadiya *et al.* [16] presented the Explainable Artificial Intelligence (XAI) technique, which provided interpretations of neural network predictions. The presented method was highly recommended for computer vision tasks in medical applications. The presented approach was integrated with XAI to recognize sign language. The method utilized an attention-based ensemble learning method for developing the prediction model. The presented method utilized the Residual Network-50 (ResNet-50) with a self-attention mechanism for designing the ensemble learning structure. The presented method was effective in smaller datasets, and its performance was optimized by lowering the learning rate. However, the presented method had an issue of overfitting during the training process. Kothadiya *et al.* [17] developed the Transformer Encoder for the recognition of sign language. The developed approach separated signs into a series of position embedding patches that were fed to the transformer block with four self-attention layers and a network of a multilayer perceptron. The developed method achieved better performance than the existing techniques. The developed method attained better accuracy with a smaller number of training epochs and layers. The developed method had difficulty in differentiating the gestures in various classes. Das *et al.* [18] introduced the Automated Indian Sign Language Recognition System for Emergency Words (AISLRSEW) method to recognize ISL words that were frequently utilized in emergencies. The introduced technique utilized the integration of handcrafted features and CNN for resolving the problem of detecting the SL words with the same hand orientation and multiple viewing angles that enhanced recognition accuracy. The introduced method consumed less training time and performed effectively in real time because of a keyframe extraction module. However, it did not capture the spatial and deep features that were significant to differentiate the various gestures in different classes. Naz *et al.* [19] implemented an effective and lightweight pose-dependent pipeline that leveraged the Graph Convolution Network (GCN) with the residual connections and the bottleneck architecture. The implemented method facilitated effective learning during training, which effectively improved accuracy and alleviated computational complexity. The implemented method utilized the residual connections and structures of the bottleneck to enhance the performance of recognition. But the implemented method did not eliminate the noise in the images, which degraded the model's performance. The existing algorithms have drawbacks, such as difficulty in predicting the finger and hand postures accurately, issues of overfitting, difficulty in differentiating the gestures in various classes, not capturing the spatial and deep features, and not eliminating the noise in the images. These drawbacks degrade a model's performance in sign language recognition. In this research, the median filter is used in the preprocessing stage to eliminate noise in images. Then, a ResNet-50-based feature extraction model is used, which extracts deep features from low-level to high-level features and additionally extracts the spatial information of hand gestures. The Class Dependent-Label Smoothing (CD-LS)-based regularization technique is used in CNN, which enhances CNN's performance to recognize the hand gestures and helps to differentiate the similar gestures in different classes with high recognition accuracy. The effective contributions of this article are given as follows.

- The CNN with CD-LS recognizes the Indian sign language by hand gesture images.
- The CD-LS-based regularization technique used in the CNN method allows assigning higher or lower smoothing for various classes depending on their characteristics in hand gestures.
- The ResNet-50-based feature extraction model is used to extract the meaningful features from hand gestures, which help to differentiate the different gestures in various classes.

This research paper is organized as follows: Section 2 explains the proposed CNN with the CD-LS method. Section 3 represents the outcomes of CNN with the CD-LS method. The conclusion of this research paper is given in Section 4.

## 2. Proposed Method

An effective DL-based algorithm is developed to recognize the Indian sign language from hand gesture images. The dataset used in this research is ISL data, and noises in the images are eliminated using the median filter, and the data is normalized by utilizing the min-max normalization method. Subsequently, the ResNet-50-based feature extraction model is utilized to extract deep meaningful features, which help to identify different gestures in the classes. Finally, the CNN with the CD-LS technique is used to recognize various gestures in different classes with high recognition accuracy. Figure 1 displays the process of Indian sign language recognition using CNN with the CD-LS method.

### 2.1. Dataset

The datasets considered for research are the Indian Sign Language (ISL) dataset [20] and the American Sign Language (ASL) [21]. The ISL dataset includes 36 classes of Indian signs from digits (0–9) and the alphabet (A–Z). Every class includes 1,200 images with three-channel images with a resolution size of 240 × 240. The ASL dataset includes 35 classes, and each class has 840 images with a resolution size of 400 × 400. Figure 2 represents sample images in the ISL dataset.

### 2.2. Preprocessing

The hand gesture image is given as input to a preprocessing phase, which removes the noise and normalizes the data to improve the data quality for further processing. In this research, the median filter is used to remove the noise, and min-max normalization is used to scale the pixel values in a certain range. These preprocessing techniques are explained below.

- Median filter — The median filter is used as a preprocessing approach to eliminate the noise from images [22]. This compares every pixel of the image to the neighborhood pixel. If any of the pixels vary from the neighborhood pixels, that is taken as

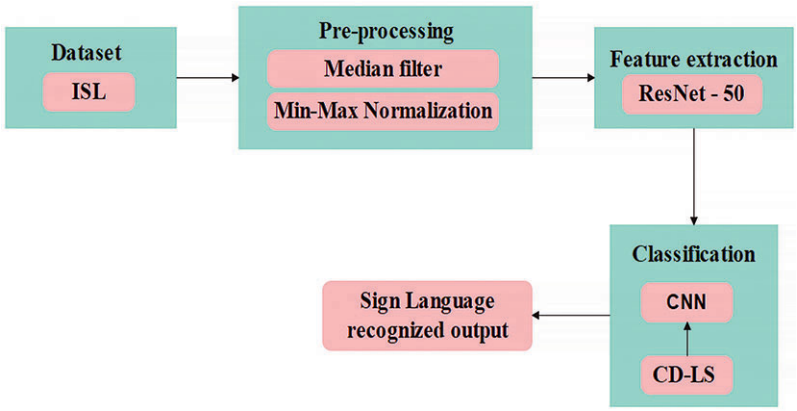


*Figure 1.* Process of Indian Sign Language Recognition.

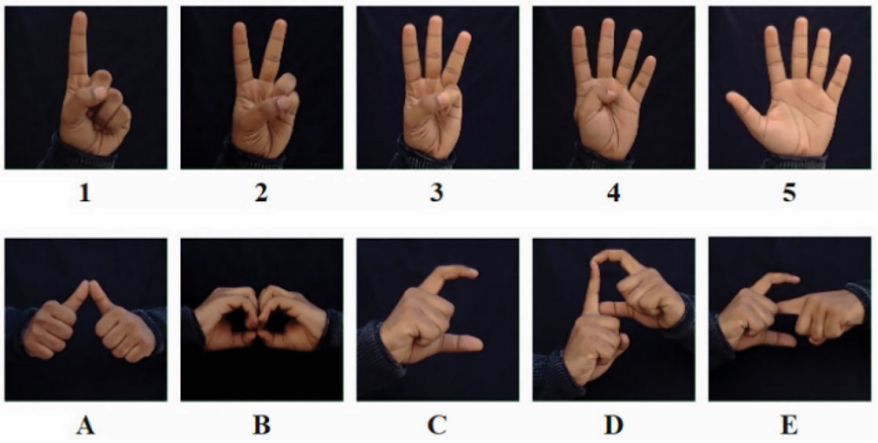


*Figure 2.* Sample images in the ISL dataset.

noise, and the median filter replaces the value of the noisy pixel with the median value of the neighborhood pixel.

- Min-Max Normalization — The min-max normalization scales the pixel values to a uniform range of [0,1] that supports model stability and convergence speed [23]. This process ensures that the method considers every pixel uniformly and mitigates the problem's relevance to different lighting conditions across images. The normalized images allow a feature extraction phase to concentrate on the original gesture. These preprocessed images are fed as input to the feature extraction phase to extract a meaningful feature to differentiate and identify the hand gestures.

## 2.3. Feature extraction

In this phase, the meaningful features from low-level to high-level are extracted using ResNet-50. Compared with other pretrained models, namely, Visual Geometry Group (VGG) and InceptionNet, ResNet-50 is less complex and includes a residual connection, which mitigates the vanishing gradient issue. The architecture of ResNet-50 includes five convolution blocks, a global average pooling layer, and a Fully Connected (FC) layer. Convolutional and pooling layers are utilized to extract deep hand gesture features [24, 25]. Figure 3 illustrates the architecture of ResNet-50.

The ResNet-50 model is initialized with ImageNet pretrained weights. The fully connected classification layer is removed, and the output of the global average pooling layer is utilized as a feature representation. These extracted feature maps are fed into the proposed CNN with CD-LS regularization. The network is trained in an end-to-end manner, allowing fine-tuning of ResNet-50 layers to adapt feature representations to the ISL dataset. This integration ensures efficient transfer learning while enabling task-specific feature refinement. The initial convolutional layers of ResNet-50 are frozen, and the final CNN classifier layers are trained on extracted feature representations. Freezing the initial layers preserves the low-level and mid-level visual features learned by the pre-

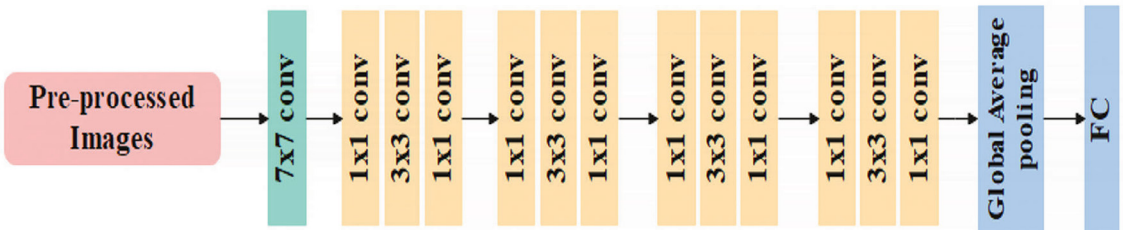


*Figure 3.* The ResNet-50 architecture.

trained network, while the proposed CNN classifier adapts these features for ISL recognition. This strategy minimizes computational complexity and helps to prevent overfitting during training.

#### 2.3.1. Convolutional Layer and Residual Blocks

Each convolution layer contains a fixed number of filters, which are considered feature detectors, and the features from pre-processed images. The convolutional layer starts with a large kernel size of (7 × 7) with two strides of size (1 × 1) and (3 × 3). The ResNet-50 includes multiple residual blocks, where every block has three convolutional layers with a size (1 × 1),(3 × 3) and (1 × 1). The first convolutional layer (1 × 1) minimizes the dimensionality, the (3 × 3) convolutional layer extracts deep features, and the (1 × 1) convolutional layer again restores dimensionality.

#### 2.3.2. Global Average Pooling Layer

After all the convolutional layers, there is a global average pooling layer used, which minimizes every feature map to a single value of a One-Dimensional (1D) feature vector, and it is used for recognizing the hand gestures. By using the ResNet-50 model, the deep meaningful features from low-level to high-level are extracted, as it is explained below. The low-level features, such as edges, corners, and textures, help identify the basic shape and structure of the hand. The mid-level features, namely, object parts and complex patterns, help identify the orientation of fingers and particular hand shapes to differentiate the signs. The high-level features, such as complex structures and entire objects that capture the entire gestures, allow the method to differentiate among various signs. Additionally, ResNet-50 captures positional data and relationships among different objects and scenes. These features are essential to differentiate the various signs of gestures, and these are given as input to the recognition phase for effective hand gesture recognition.

### 2.4. Sign Language Recognition by CNN with CD-LS

The recognition of the Indian sign language is performed by utilizing the CNN, as it gives effective performance in recognition. The CNN architecture has a convolution layer, max-pooling layers, one fully connected layer, and an output layer. Figure 4 demonstrates the architecture of a CNN with CD-LS. The convolutional layer $CV_1$, which includes an input matrix of 32 kernels of size 3 × 3. The mathematical expression for measuring the convolution matrix is given in Equation (1).

$$CV_i^{j-1}(m,n) = \sum_{y=1}^{x} \Omega\left(m\left(m - y + \frac{x+1}{2}\right)\right) U_i^j(y) + \alpha_i^j \tag{1}$$

In the Equation (1) above, the $(m,n)$ represents the coordinates in the convolution layer, the $\Omega$

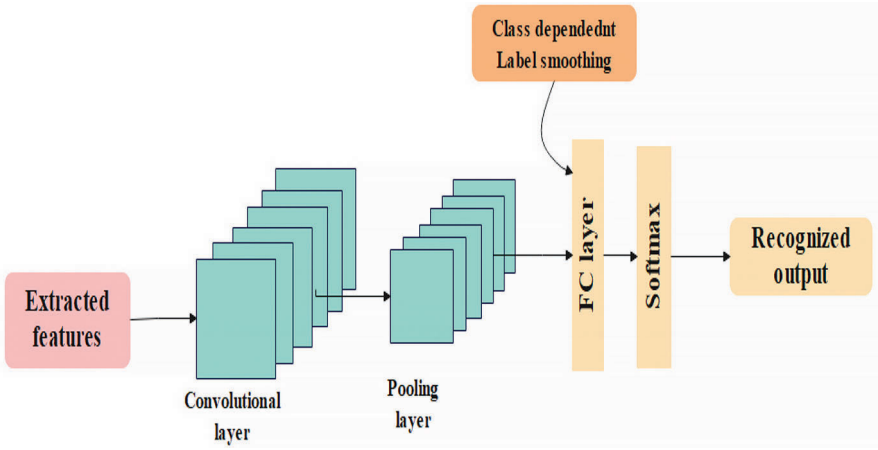


*Figure 4.* Architecture of a CNN with CD-LS.

represents the final layer maps and kernel size, the $U_i^j$ represents the $j$-th kernel of the $i$-th layer, and the $\alpha_i^j$ represents the bias of the convolution layer. The output of a convolution layer is fed to an initial layer of the max-pooling layer. The addition of bias and weight among the convolution and max pooling layers is measured by utilizing the Rectified Linear Unit (ReLU) activation function and then fed to the following layer. The mathematical expression for the pooling layer is given in Equation (2).

$$M_i^{j-1}(m,n) = \max\left(CV_i^j\left(m\left((n-1)\times(r,s)\right)\right)\right) \quad (2)$$

In Equation (2), the $(r-1)$ is the output layer, $s$ is the kernel, and $(m,n)$ represents the coordinate. The output from an initial pooling layer is fed into a second convolution layer, which has 64 kernels. This process is repeated in subsequent layers with 128 kernels, allowing for deep feature extraction. The mathematical expression for a final fully connected layer is given in Equation (3). In Equation (3) below, the $d_t^j U_{ju}^j$ represents the matrix with weights.

$$F_u^{j+1} = \sum_{y=1}^{x} \Omega\left(m\left(n-y+\frac{x+1}{2}\right)\right) U_i^j(y) + \alpha_i^j + \alpha_i^j\left(\sum_t d_t^j U_{ju}^j + \alpha_u^j\right) \quad (3)$$

Class Dependent – Label Smoothing: In the multiclass classification, the features are employed to hard labels, and the mathematical expression is given in Equation (4). In Equation (4) below, the $y_i$ ($i = 1, 2, \ldots, C$) is described as a one-hot vector, the $y_i = 1$ when the features belong to $i$th class. The CNN results in a probability that the input feature vector belongs to every class through the softmax function, and the Categorical Entropy (CE) loss is used as the classification loss. The mathematical expressions are given in Equations (5) and (6).

$$y_i = \begin{cases} 1, & i = \text{true} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

$$l_{CE} = -\sum_{i=1}^{C} y_i \log q_i \quad (5)$$

$$= -\log q_{\text{true}} \quad (6)$$

In hard label classification using CE loss, the predicted label is limited to either one for the correct class or zero for other classes. In the classification vector, the value of the correct class $p_i$ leads to infinity, making the method much more confident in its prediction. To overcome this issue, smooth labels are used instead of hard labels, and the mathematical formula is given in Equation (7) for class $j$.

$$y_i^{LS} = \begin{cases} 1-\varepsilon, & i = j \\ \dfrac{\varepsilon}{C-1}, & \text{otherwise} \end{cases} \quad (7)$$

In Equation (7), the $\varepsilon$ represents the fixed smoothing parameter and $C$ is the number of classes. Here, the label smoothing is class-dependent, *i.e.*, every class $j$ ($j = 1, 2, \ldots, C$), has various smoothing parameters $\varepsilon_j$. The mathematical expression for Class Dependent Label Smoothing (CDLS) is given in Equation (8).

$$y_i^{CDLS} = \begin{cases} 1-\varepsilon_j, & i = j \\ \dfrac{\varepsilon_j}{N-1}, & \text{otherwise} \end{cases} \quad (8)$$

This process allows the method to assign higher or lower smoothing for various classes depending on their characteristics. The $\varepsilon_j$ is a class-dependent smoothing parameter for class $j$. The probability mass employed to correct the class is $1 - \varepsilon_j$, while the remaining probability is equally distributed among $N - 1$ classes. The mathematical expression for total probability mass is given as Equation (9).

$$\left(1-\varepsilon_j\right) + (N-1)\frac{\varepsilon_j}{N-1} = 1 \quad (9)$$

Hence, the proposed CD-LS maintains that the total probability mass remains equal to one even when class-specific smoothing parameters are employed, ensuring probabilistic consistency across all classes. The smoothing parameter $\varepsilon_j$ for every class is defined depending on inter-class confusion statistics derived from the validation set. Consider $C_i$ that represents the normalized confusion value of the class $i$ calculated from the confusion matrix. Classes that show higher misclassification rates are employed with larger smoothing values to prevent overconfidence. The mathematical expression for computing the smoothing parameter is given in Equation (10).

$$\varepsilon_j = \lambda \times \frac{C_i}{\max(C)} \quad (10)$$

In Equation (10), $C_i$ is the confusion score of the class $i$, $\lambda$ is the maximum smoothing bound, and max($C$) represents a normalization term. This ensures that visually similar gesture classes have strong regularization, while well-separated classes retain lower smoothing. This approach ensures structured regularization aligned with dataset characteristics. CD-LS and Adaptive Label Smoothing (ALS) mitigate model overconfidence but differ in the smoothing parameter determined. In ALS, the smoothing parameter is dynamically adjusted for every training sample based on prediction confidence during training, making it instance-dependent. The proposed CD-LS employs a different smoothing parameter for every class, computed from inter-class confusion statistics and gesture ambiguity acquired from the validation set. Classes with higher misclassification rates receive larger smoothing values, while well-separated classes retain smaller smoothing. Hence, ALS adapts smoothing at the sample level, whereas CD-LS performs structured regularization at the class level, enabling better handling of visually similar gesture classes in sign language recognition. The noises in the images are eliminated by using the median filter, and the data is normalized by utilizing the min-max normalization method. Later, the ResNet-50-based feature extraction technique is utilized to extract deep, meaningful features, which helps to identify different gestures in the classes. Finally, the CNN with the CD-LS technique is used to recognize the various gestures in different classes with high recognition accuracy.

## 3. Experimental Results

The performance of CNN with the CD-LS method is evaluated in a Python environment, with the following configuration: Windows 10 OS, 8 GB RAM, and an i5 processor. The performance metrics considered to evaluate the CNN with the CD-LS method are accuracy, precision, recall, and F1-score. The mathematical expressions for performance metrics are given in Equations (11) to (14).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (11)$$

$$Recall = \frac{TN}{TN + FP} \quad (12)$$

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

$$F1 - score = \frac{2 \times precision \times recall}{precision + recall} \quad (14)$$

To prevent overfitting and ensure reliable performance validation, ISL datasets are partitioned by a stratified 70 : 15 : 15 split for training, validation, and testing. The split is performed to prevent data leakage.

In Table 1, the performance of the ResNet-50-based feature extraction model is evaluated with different feature extraction methods on metrics such as precision, F1-score, accuracy, and recall. Various feature extraction methods used to evaluate a ResNet-50 method are InceptionNet, VGG-16, VGG-19, and ResNet-50. Compared to ResNet-50, ResNet-18 is less complex, but it has only 18 layers, which is sufficient to capture

*Table 1*. Performance of the ResNet-50-based feature extraction model..

| Feature extraction methods | Accuracy (%) | Precision | Recall | F1-score |
|---|---|---|---|---|
| InceptionNet - CNN | 95.4 | 0.94 | 0.93 | 0.94 |
| VGG-16 - CNN | 96.0 | 0.95 | 0.94 | 0.94 |
| VGG-19 - CNN | 96.5 | 0.96 | 0.95 | 0.95 |
| ResNet-18 - CNN | 97.3 | 0.97 | 0.96 | 0.97 |
| ResNet-50 - CNN | 99.6 | 0.99 | 0.99 | 0.99 |

the deep features. Although VGG-16, VGG-19, and InceptionNet are deep architectures, their performance varies depending on depth, parameter complexity, and feature reuse mechanisms. The ResNet-50 has 50 layers that capture the deep features from low-level to high-level and include the residual connection, which facilitates the effective gradient flow and mitigates the gradient vanishing issue. To ensure a fair comparison of feature extraction backbones, all pretrained models are used as feature extractors by eliminating their original classification layers. The extracted feature maps are then fed into an identical CNN-based classifier under the same training settings, hyperparameters, optimizer, batch size, and learning rate. The ResNet-50-based feature extraction methods reach 99.6% accuracy, 0.99 precision, 0.99 recall, and a 0.99 F1-score when compared to various feature extraction methods.

In Table 2, the performance of the CD-LS regularization technique is evaluated with different conventional regularization techniques on various metrics such as precision, F1-score, accuracy, and recall. The various regularization techniques used for evaluating the CD-LS technique are Dropout, L2-regularization, Adaptive LS (ALS), and conventional LS. Compared to the CD-LS technique, the other regularization methods apply a uniform smoothing factor across all classes and then randomly drop the unit in the network, which is less effective for varying class difficulties. Unlike ALS, which adjusts smoothing dynamically at the sample level, the proposed CD-LS employs smoothing parameters in class levels, enabling better handling of inter-class similarity in gesture data. But the CD-LS technique sets the various smoothing parameters for every class and label smoothing to a particular task, prevents the overfitting issue, and effectively handles the variation in classes. The CD-LS-based regularization technique reaches 99.6% accuracy compared to various regularization techniques.

In Table 3, the performance of the CNN with the CD-LS method is evaluated with different standard classifiers on various metrics, namely, precision, F1-score, accuracy, and recall. The various classifiers used for evaluating the CNN with the CD-LS method are Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and standard CNN. While comparing a CNN with the CD-LS method, the other existing methods have difficulties in differentiating the classes with various characters or gestures. The CNN with the CD-LS technique sets different smoothing parameters for every class, which prevents the overfitting issue and effectively handles the variation in classes. CNN with the CD-LS method reaches 99.6% accuracy when compared to various classifiers.

Table 4 represents the ablation study, which represents the individual and combination of proposed models. The standard CNN obtains high performance because of limited depth and minimized ability to extract fine-grained spatial dependencies among visually similar gestures. ResNet-50 enhances performance by using residual learning that improves gradient flow and enables deeper semantic feature extraction. ResNet-50 and CNN facilitate multi-level fea-

*Table 2*. Performance of CD-LS-based regularization technique.

| Regularization techniques | Accuracy (%) | Precision | Recall | F1-score |
|---|---|---|---|---|
| Dropout | 96.4 | 0.93 | 0.92 | 0.92 |
| L2 | 97.1 | 0.94 | 0.94 | 0.94 |
| LS | 97.8 | 0.95 | 0.95 | 0.95 |
| ALS | 98.2 | 0.97 | 0.96 | 0.96 |
| CD-LS | 99.6 | 0.99 | 0.99 | 0.99 |

*Table 3*. Performance of CNN with CD-LS method.

| Methods | Accuracy (%) | Precision | Recall | F1-score |
|---|---|---|---|---|
| ANN | 95.0 | 0.91 | 0.90 | 0.90 |
| RNN | 95.7 | 0.92 | 0.91 | 0.92 |
| LSTM | 96.5 | 0.94 | 0.93 | 0.94 |
| CNN | 97.2 | 0.96 | 0.95 | 0.95 |
| CNN with CD-LS method | 99.6 | 0.99 | 0.99 | 0.99 |

*Table 4*. Ablation study of the proposed CNN with the CD-LS model using the ISL dataset.

| Methods | Accuracy (%) | Precision | Recall | F1-score |
|---|---|---|---|---|
| ResNet50 | 98.4 | 0.98 | 0.97 | 0.98 |
| CNN | 97.2 | 0.96 | 0.95 | 0.95 |
| CNN with ResNet50 | 98.9 | 0.98 | 0.98 | 0.98 |
| ResNet-50 with CD-LS | 99.1 | 0.98 | 0.99 | 0.99 |
| CNN with CD-LS | 99.3 | 0.99 | 0.99 | 0.99 |
| ResNet50 and CNN with CD-LS | 99.6 | 0.99 | 0.99 | 0.99 |

ture refinement, enhancing inter-class separability. The incorporation of CD-LS strengthens performance by modifying target distribution to minimize overconfidence and smooth class boundaries, especially for gesture classes. By employing class-dependent smoothing parameters, CD-LS reshapes the optimization landscape, stabilizes gradient updates, and enhances generalization. The higher accuracy obtained by combining ResNet-50, CNN, and CD-LS demonstrates that deep residual feature learning and probabilistically consistent class-wise regularization improve discriminative ability and convergence stability in multi-class sign language recognition. In Table 5 below, the evaluation metrics of precision, recall, and F1-score consistently perform for the proposed CNN with the CD-LS model.

Table 6 presents the statistical analysis to validate the performance improvement of the proposed CNN with the CD-LS method. Every experiment is repeated across multiple random seeds, and the standard deviation is computed to assess performance stability. The reduced standard deviation represents progressive enhanced convergence consistency and minimized variance across runs. The reported p-values are acquired by a paired t-test comparing every baseline model against the proposed CNN with the CD-LS model. All p-values are below the 0.05 significance threshold, indicating that performance differences are statistically significant. The proposed model obtains a lower p-value, which determines that its performance improvement is high because of random variation. The minimized standard deviation supports that

*Table 5*. Class-wise performance of the proposed CNN with the CD-LS model using numbers in the ISL dataset.

| Classes | Precision | Recall | F1-score |
|---|---|---|---|
| 1 | 1.00 | 1.00 | 1.00 |
| 2 | 1.00 | 1.00 | 1.00 |
| 3 | 0.98 | 0.98 | 0.98 |
| 4 | 1.00 | 1.00 | 1.00 |
| 5 | 1.00 | 1.00 | 1.00 |
| 6 | 0.98 | 0.97 | 0.97 |
| 7 | 0.98 | 0.98 | 0.98 |
| 8 | 0.98 | 0.97 | 0.97 |
| 9 | 0.99 | 0.99 | 0.99 |

*Table 6*. Class-wise performance of the proposed CNN with the CD-LS model using numbers in the ISL dataset.

| Methods | P-value from t-test | Standard Deviation | CI |
|---|---|---|---|
| ANN | 0.021 | 0.42 | ± 0.26 |
| RNN | 0.018 | 0.37 | ± 0.23 |
| LSTM | 0.012 | 0.31 | ± 0.19 |
| CNN | 0.009 | 0.24 | ± 0.15 |
| CNN with CD-LS method | 0.004 | 0.12 | ± 0.07 |

CD-LS stabilizes gradient updates, reshapes the optimization landscape, and improves generalization reliability in multi-class gesture recognition.

Figures 5 and 6 show the accuracy and loss curves, respectively, demonstrating stable convergence and strong generalization behavior of the proposed CNN with the CD-LS model. As illustrated in the accuracy plot, training and validation accuracies increase steadily, and convergence toward significant divergence represents minimal overfitting. The curves represent efficient regularization that is attributed to the CD-LS mechanism, which minimizes overconfidence and stabilizes gradient updates. The loss curves signify smooth decay for training, and validation indicates optimization stability and proper learning dynamics. Figure 7 represents the confusion matrix for the CNN with the CD-LS method.

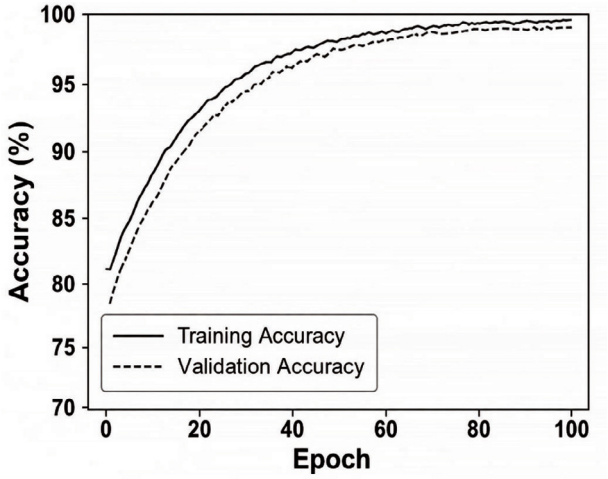


*Figure 5.* Accuracy vs. Epoch graph for the proposed CNN with the CD-LS method using the ISL dataset.

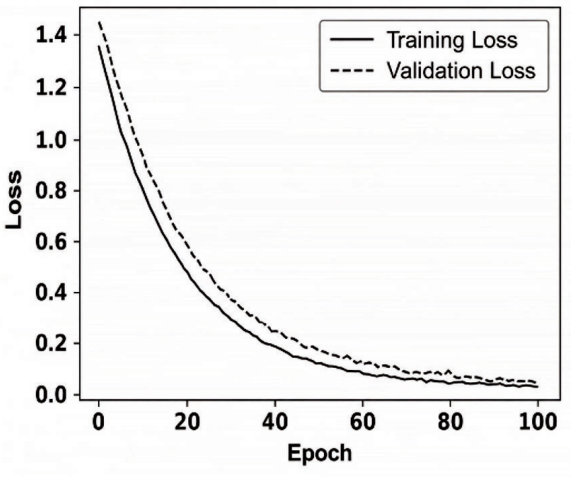


*Figure 6.* Loss vs. Epoch graph for the proposed CNN with the CD-LS method using the ISL dataset.

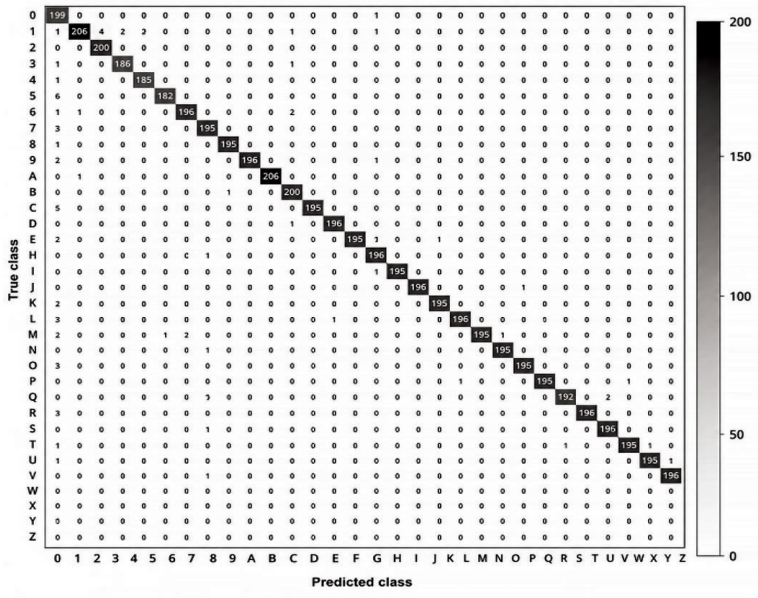


*Figure 7.* Confusion matrix for CNN with CD-LS method.

## 3.1. Comparative analysis

The performance of CNN with the CD-LS method for Indian sign language recognition is compared with existing methods such as CNN [15], ResNet-50 + Attention [16], and transformer encoder [17] on the ISL and ASL datasets. Accuracy, F1-score, recall, and precision metrics are used to evaluate the performance of CNN with the CD-LS method on Indian sign language recognition under a static background. When compared to the CD-LS technique, the other regularization methods apply a uniform smoothing factor across all classes and randomly drop units in the network, which is less effective for varying class difficulties. This process enhances the performance of CNN in Indian Sign Language recognition on the ISL and ASL datasets. CNN with the CD-LS method reaches 99.6% accuracy when compared to

*Table 7*. Comparative analysis of CNN with the CD-LS method.

| Methods | Dataset | Accuracy (%) | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| CNN [15] | ISL | 99.93 | 0.98 | 0.98 | NA |
| ResNet 50 + Attention [16] | ISL | 98.20 | 0.98 | 0.98 | 0.97 |
| | ASL | 98.0 | NA | NA | NA |
| Transformer encoder [17] | ISL | 99.29 | 0.99 | NA | 0.99 |
| | ASL | 98.31 | 0.98 | NA | 0.99 |
| Proposed CNN with CD-LS method | ISL | 99.6 | 0.99 | 0.99 | 0.99 |
| | ASL | 98.67 | 0.98 | 0.98 | 0.98 |

various existing methods. Table 7 presents the comparative analysis of CNN with the CD-LS method. The existing CNN [15] model values are taken from an actual study and correspond to its specific experimental configuration, including Kinect-based data acquisition, preprocessing, and training settings. Hence, the comparison presented in Table 7 is a dataset-level comparison from the literature rather than a direct evaluation under identical experimental conditions. While CNN [15] exhibits higher accuracy on the ISL dataset, the proposed CNN with the CD-LS model focuses on enhancing generalization ability and classification robustness by class-dependent label smoothing. The existing model obtains a recall of 98% and an F1-score of 97%, but the proposed model improves the per-class enhancement, so the class-wise metrics of recall and F1-score consistently obtain improved performance.

## 3.2. Discussion

The results of the CNN with the CD-LS method are evaluated with various feature extraction methods, regularization techniques, and various classifiers. The performance of CNN with the CD-LS method is evaluated on ResNet-18, VGG-16, VGG-19, and InceptionNet, with L2 regularization, dropout, standard LS, ALS, ANN, RNN, LSTM, and conventional CNN. The existing algorithms have drawbacks such as difficulty in predicting the finger and hand postures accurately, issues of overfitting, difficulty in differentiating the gestures in various classes, not capturing the spatial and deep features, and not eliminating the noise in the images. These drawbacks degrade the model's performance in sign language recognition. In this research, the median filter is used in the preprocessing phase to eliminate noise in images. Subsequently, a ResNet-50-based feature extraction model is used, which extracts deep features from low-level to high-level features and additionally extracts the spatial information of hand gestures. The CD-LS-based regularization technique is used in CNN, which enhances the CNN's performance to recognize the hand gestures and helps to differentiate the variant gestures in different classes with high recognition accuracy. The experimental results determine that integrating class-dependent label smoothing improves gesture recognition performance when compared with standard CNN architecture. The results show that class-wise smoothing efficiently mitigates overconfidence in visually similar gesture classes. This is significant in the ISL dataset, where subtle variations in finger positioning lead to higher inter-class similarity. The ResNet-50 outperforms existing models because of its deeper hierarchical feature extraction ability, which enables better representation of complex spatial patterns in hand gestures. The residual learning mechanism facilitates stable gradient propagation and supports refined feature learning across layers. The confusion matrix analysis indicates that misclassifications are mainly between vi-

sually similar alphabet gestures, while numerically different gestures show high performance. The CD-LS mechanism minimizes prediction sharpness for ambiguous classes by enhancing inter-class separability. The proposed model obtains high performance because it depends on proper estimation of class-dependent smoothing parameters. Overall, the proposed model demonstrates that structured regularization enhances gesture recognition performance while maintaining stable convergence behavior.

## 4. Conclusion

To bridge the gap between people with impairments and the general population, an effective DL-based sign language recognition method was developed. By using the median filter, the noise in images was removed, and then, the data were uniformly scaled using min-max normalization. Using ResNet-50 for feature extraction, low-level and high-level features, along with spatial information, were used to identify various gestures in the images. Then, CNN with the CD-LS method was used to recognize the sign language from hand gestures with high recognition accuracy. The CD-LS regularization method was used in the CNN method, which helped to differentiate similar gestures in different classes based on their characteristics. CNN with the CD-LS method achieved 99.6% accuracy and outperformed various existing methods. In the future, another variant of CNN can be used to further improve recognition performance for Indian Sign Language.

## Declaration of Competing Interests

The authors declare no conflict of interest.

## Funding

None declared.

## Data Availability

The datasets used in this study are publicly available. The Indian Sign Language (ISL) dataset is available at Kaggle (https://www.kaggle.com/datasets/vaishnaviasonawane/indian-sign-language-dataset), and the American Sign Language (ASL) dataset is available at Kaggle (https://www.kaggle.com/datasets/ayuraj/asl-dataset).

## References

[1] S. Das *et al.*, "A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier", *Expert Syst. Appl.*, vol. 213, no. part B, p. 118914, 2023. https://doi.org/10.1016/j.eswa.2022.118914

[2] R. Sreemathy *et al.*, "Continuous word-level sign language recognition using an expert system based on machine learning", *Int. J. Cognit. Comput. Eng.*, vol. 4, pp. 170–178, 2023. https://doi.org/10.1016/j.ijcce.2023.04.002

[3] G. Park, V. K. Chandrasegar, and J. Koh, "Accuracy enhancement of hand gesture recognition using CN", *IEEE Access*, vol. 11, pp. 26496–26501, 2023. https://doi.org/10.1109/ACCESS.2023.3254537

[4] H. Ansar *et al.*, "Hand gesture recognition for character understanding using convex hull landmarks and geometric features", *IEEE Access*, vol. 11, pp. 82065–82078, 2023. https://doi.org/10.1109/ACCESS.2023.3300712

[5] H. A. AbdElghfar e*t al.*, "A model for Qur'anic sign language recognition based on deep learning algorithms", *J. Sens.*, vol. 2023, no. 1, p. 9926245, 2023. https://doi.org/10.1155/2023/9926245

[6] A. M. Buttar *et al.*, "Deep learning in sign language recognition: a hybrid approach for the recognition of static and dynamic signs", *Mathematics*, vol. 11, no. 17, p. 3729, 2023. https://doi.org/10.3390/math11173729

[7] E. Rajalakshmi *et al.*, "Multi-semantic discriminative feature learning for sign gesture recognition using hybrid deep neural architecture", *IEEE Access*, vol. 11, pp. 2226–2238, 2023. https://doi.org/10.1109/ACCESS.2022.3233671

[8] A. Venugopalan and R. Reghunadhan, "Applying hybrid deep neural network for the recognition of sign language words used by the deaf Covid-19 patients", *Arabian J. Sci. Eng.*, vol. 48, no. 2, pp. 1349–1362, April 2022. https://doi.org/10.1007/s13369-022-06843-0

[9] S. Das, S. K. Biswas, and B. Purkayastha, "A deep sign language recognition system for Indian sign language", *Neural Comput. Appl.*, vol. 35, no. 2, pp. 1469–1481, 2022. https://doi.org/10.1007/s00521-022-07840-y

[10] R. Sreemathy, M. Turuk, I. Kulkarni, and S. Khurana, "Sign language recognition using artificial intelligence", *Education and Information Technologies*, vol. 28, no. 5, pp. 5259–5278, 2022. https://doi.org/10.1007/s10639-022-11391-z

[11] U. Nandi *et al.*, "Indian sign language alphabet recognition system using CNN with diffGrad optimizer and stochastic pooling", *Multimedia Tools Appl.*, vol. 82, no. 7, pp. 9627–9648, 2023. https://doi.org/10.1007/s11042-021-11595-4

[12] B. Natarajan, R. Elakkiya, and M. L. Prasad, "Sentence2SignGesture: a hybrid neural machine translation network for sign language video generation", *J. Ambient Intell. Hum. Comput.*, vol. 14, no. 8, pp. 9807–9821, 2022. https://doi.org/10.1007/s12652-021-03640-9

[13] A. A. Barbhuiya, R. K. Karsh, and R. Jain, "ASL hand gesture classification and localization using deep ensemble neural network", *Arabian J. Sci. Eng.*, vol. 48, no. 5, pp. 6689–6702, 2023. https://doi.org/10.1007/s13369-022-07495-w

[14] A. Dayal *et al.*, "Adversarial unsupervised domain adaptation for hand gesture recognition using thermal images", *IEEE Sens. J.*, vol. 23, no. 4, pp. 3493–3504, 2023. https://doi.org/10.1109/JSEN.2023.3235379

[15] J. Gangrade and J. Bharti, "Vision-based hand gesture recognition for Indian sign language using convolution neural network", *IETE J. Res.*, vol. 69, no. 2, pp. 723–732, 2023. https://doi.org/10.1080/03772063.2020.1838342

[16] D. R. Kothadiya *et al.*, "SignExplainer: an explainable AI-enabled framework for sign language recognition with ensemble learning", *IEEE Access*, vol. 11, pp. 47410–47419, 2023. https://doi.org/10.1109/ACCESS.2023.3274851

[17] D. R. Kothadiya *et al.*, "SIGNFORMER: deepvision transformer for sign language recognition", *IEEE Access*, vol. 11, pp. 4730–4739, 2023. https://doi.org/10.1109/ACCESS.2022.3231130

[18] S. Das, S. K. Biswas, and B. Purkayastha, "Automated Indian sign language recognition system by fusing deep and handcrafted features", *Multimedia Tools Appl.*, vol. 82, no. 11, pp. 16905–16927, 2022. https://doi.org/10.1007/s11042-022-14084-4

[19] N. Naz *et al.*, "Signgraph: An efficient and accurate pose-based graph convolution approach toward sign language recognition", *IEEE Access*, vol. 11, pp. 19135–19147, 2023. https://doi.org/10.1109/ACCESS.2023.3247761

[20] ISL dataset: https://www.kaggle.com/datasets/vaishnaviasonawane/indian-sign-language-dataset/data

[21] ASL dataset: https://www.kaggle.com/datasets/ayuraj/asl-dataset

[22] S. Rani *et al.*, "Efficient 3D AlexNet architecture for object recognition using syntactic patterns from medical images", *Comput. Intell. Neurosci.*, vol. 2022, no. 1, p. 7882924, 2022. https://doi.org/10.1155/2022/7882924

[23] X. Pei *et al.*, "Robustness of machine learning to color, size change, normalization, and image enhancement on micrograph datasets with large sample differences", *Mater. Des.*, vol. 232, p. 112086, 2023. https://doi.org/10.1016/j.matdes.2023.112086

[24] D. AlSaeed and S. F. Omar, "Brain MRI analysis for Alzheimer's disease diagnosis using CNN-based feature extraction and machine learning", *Sensors*, vol. 22, no. 8, p. 2911, 2022. https://doi.org/10.3390/s22082911

[25] Z. Zhang, Q. Shen, and Y. Wang, "Electromyographic hand gesture recognition using convolutional neural network with multi-attention", *Biomed. Signal Process. Control*, vol. 91, p. 105935, 2024. https://doi.org/10.1016/j.bspc.2023.105935

*Received:* July 2025
*Revised:* January 2026
*Accepted:* February 2026

*Contact addresses*:
Keerthi Kumar Masanasetty
Department of Electronics and Communication Engineering
Nitte Meenakshi Institute of Technology
Nitte (Deemed University)
Bengaluru
India
e-mail: keerthi.kumar@nmit.ac.in

Parameshachari Bidare Divakarachari
Department of Electronics and Communication Engineering
Nitte Meenakshi Institute of Technology
Nitte (Deemed University)
Bengaluru
India
e-mail: paramesh@nmit.ac.in

Keerthi Kumar Masanasetty is an Assistant Professor in the Department of Electronics and Communication Engineering at Nitte Meenakshi Institute of Technology (NMIT), Bengaluru, India. He is currently pursuing a Ph.D. and has over 12 years of teaching experience. His primary research interests include embedded systems, the Internet of Things (IoT), robotics, machine learning, and communication systems.

Parameshachari Bidare Divakarachari is a Professor and Head of the Department of Electronics and Communication Engineering at Nitte Meenakshi Institute of Technology, Bengaluru, India, and a Visiting Professor at the University of Rome Tor Vergata, Italy. He is recognized among Stanford University's World's Top 2% Scientists and serves as a Ph.D. supervisor at VTU, Belagavi. He has published more than 150 journal and conference papers and serves as Associate Editor for several international journals. His research interests include electronics, communication engineering, embedded systems, and artificial intelligence.