A Two-Stream Convolutional Attention Network for Hand Gesture Recognition and Classification

Chaovarit Janpirom
Department of Information Technology, Faculty of Digital Technology and Innovation, Southeast Bangkok University (SBU), Thailand
chaovarit@sbu.southeast.ac.th

Bunthida Chunngam
Department of Computer Engineering, Faculty of Industrial Education, Rajamangala University of Technology Suvarnabhumi (RMUTSB), Thailand
bunthida.c@rmutsb.ac.th

Tanchanok Phewkham
Department of Computer Engineering, Faculty of Industrial Education, Rajamangala University of Technology Suvarnabhumi (RMUTSB), Thailand
tanchanok.ph@rmutsb.ac.th

Aekkarat Suksukont
Department of Computer Engineering, Faculty of Industrial Education, Rajamangala University of Technology Suvarnabhumi (RMUTSB), Thailand
aekkarat.s@rmutsb.ac.th (corresponding author)

Received: 17 September 2025 | Revised: 6 November 2025 | Accepted: 15 November 2025

 DOI: https://doi.org/10.48084/etasr.14865

ABSTRACT

This study presents a two-stream convolutional attention network for hand gesture recognition and classification, with dual branches that effectively extract a range of spatial features. To further improve its capabilities, Convolutional Neural Network (CNN), Residual Blocks (RB), Attention Mechanism (AM), and Convolutional Block Attention Module (CBAM) are integrated, enabling the network to focus on specific regions of the input while reducing background noise. This structure allows the model to capture variations in hand gestures more effectively. Experiments utilized the Hand-Sign-Images Dataset (HSID) and the Hand Gesture Dataset (HGD), each offering a diverse array of hand gesture patterns. The results indicate that the model achieved high training accuracies of 100% and 98.68%, demonstrating its effectiveness in recognizing complex hand gestures. The proposed approach not only enhances recognition and classification accuracy but also offers strong adaptability, making it suitable for applications such as robot control and assistive communication systems.

Keywords-convolutional neural network; attention mechanism; convolutional block attention module; hand gesture recognition; hand gesture classification

I. INTRODUCTION

The development of hand gesture recognition methods depends on hand shape analysis, along with image processing and computer vision [1]. Early methods focused on analyzing skin color, removing backgrounds, or identifying regions of interest to describe features such as shape, position, and size, particularly those related to palm contours or finger joints. Feature extraction algorithms were also employed to cluster and categorize data, enabling systems to interpret gestures accurately [2-4]. As hand movements have become more complex and varied, traditional methods have limitations in recognition accuracy. In response, Deep Learning (DL) has increasingly improved classification and recognition performance, driving significant progress in the field. In recent years, DL has played a vital role in hand gesture classification and recognition, providing substantial benefits in managing variations in data collection, background images, and hand

shapes. These features enhance the efficiency of model training and testing. In [5], proposed radar-based and vision-based methods often suffer from environmental noise, illumination, or background complexity. In contrast, wearable sensors offer greater accuracy and reliability. This study proposes a multimodal fusion framework that combines flex sensors with gyroscope and accelerometer signals, employing various fusion strategies. Experimental results show that multiple kernel learning fusion achieved the highest performance, with recognition accuracies of 99.87% for static gestures and 97.59% for both static and dynamic gestures, confirming the effectiveness of multi-modal sensor fusion in practical gesture recognition systems. In [6], a dynamic adaptive CNN is proposed to handle variations in hand shape, size, and orientation. Using the sign language MNIST dataset and extensive preprocessing, data augmentation, and morphological operations, the model achieved 99% recognition accuracy.

These results highlight the potential of adaptive DL frameworks to enhance gesture recognition accuracy and robustness in real-world HCI applications. In [7], the authors proposed a 3D hand-gesture classification approach using Mask R-CNN. This method comprises a backbone, a Region Proposal Network (RPN), Regions of Interest (ROIs), a network head, and a loss function. Specifically, it employs ResNet50 paired with a feature pyramid network to facilitate effective multi-scale feature extraction. In [8], the authors proposed a framework employing surface electromyography (sEMG) and an auto-label-refining CNN to address label noise, thereby improving robustness to noisy datasets. In [9], a DL is presented that integrates a continuous wavelet transform, a CNN, and a Temporal Convolution Network (TCN) to extract features from the time-frequency domain of VR-based positional data, enabling accurate recognition of dual-hand gestures in 3D virtual reality. In [10], a multichannel radar and a multistream fusion 1D convolutional neural network were proposed for frequency-modulated continuous-wave radar. This approach processes raw radar data through four parallel Inception branches to extract gesture features, followed by LSTM (Long Short-Term Memory) layers and a dense layer with a softmax activation function for multi-class gesture learning and classification [11, 12].

The adaptation of publicly available networks to new gesture datasets has been extensively explored using models such as VGG16 [13], the deep residual network (ResNet) [14], Inception-v3 [15], transformer networks [16], and transfer learning approaches [17, 18]. Additionally, research continues to focus on real-time hand gesture detection and classification [19]. Collectively, these studies emphasize the importance of creating deep neural networks with streamlined architectures that balance model complexity and performance to enable robust, adaptable gesture recognition systems. Research in deep learning for hand gesture recognition has resulted in models capable of handling increasingly complex data. Examples include Mask R-CNN, which uses ResNet50 and a feature pyramid network to extract multi-scale features from images [7], and CWT-CNN-TCN, which combines time-frequency features for gesture classification in virtual reality environments [9]. While these models demonstrate strong accuracy and learning ability, challenges such as network complexity, high computational costs, real-time processing limitations, and sensitivity to noise persist. To address these issues, this paper introduces an approach for hand gesture recognition and classification based on a proposed two-stream convolutional attention network. The network architecture is built on a CNN framework, enhanced with RB, AM, and CBAM modules, which together improve the network's capacity to learn hierarchical and attention-aware representations, as shown below:

- This study combines two-stream convolutional attention networks for hand gesture recognition and classification. The proposed model is fine-tuned using mini-batch learning and trained to optimize accuracy across various gesture patterns.
- This study combines RB, AM, and CBAM with CNN to improve the network's ability to recognize and classify hand gesture shapes.

II. RESEARCH THEORY

This section presents the theoretical foundations used in the experiment, along with details of their implementation and application in the study.

A. Convolutional Neural Network

The CNN focuses on processing and extracting features of 2D data [9, 14, 20]. It is a classifier that uses convolutional layers with filters to detect spatial patterns and applies the ReLU function to introduce nonlinearity, thereby helping extract complex features. The feature map dimensions decrease while retaining essential information. These extracted features are then fed into fully connected layers for classification, and the softmax function in the final layer produces class probabilities. The operational details are outlined as:

$$A(i,j) = \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} I(i+m, j+n) K(m,n) \quad (1)$$

where I I is an image of size $M\ x\ N$ and there is a filter K of size $f\ x\ f$, the output feature map value at position $A_{i,j}$ is calculated by sliding the filter $K(m,n)$ across the image $I(i,j)$.

An element-wise product is computed between the filter and the corresponding region of the image, and the results are summed to produce the feature value on the feature map.

B. Residual Blocks

Residual Blocks (RB) are designed to address the vanishing gradient problem in DL, allowing Neural Networks (NN) to learn deeper representations more effectively. In this study, RB is used for its ability to retain important information across layers and to reduce feature loss during training [21, 22]. The operational details are described as:

$$y = F(x) + x \quad (2)$$

where x is the input of the RB, it computes $F(x)$ as learned features. This skip connection maintains essential information and improves learning in networks.

C. Convolutional Block Attention Module

The CBAM improves feature representation by sequentially inferring channel and Spatial Attention Maps (SAM), enabling

the network to focus on what and where to emphasize during feature learning [23, 24]. The CBAM architecture consists of two submodules: the Channel Attention Module (CAM) and the SAM. The channel attention mechanism leverages inter-channel dependencies by applying both Global Average Pooling (GAP) and Global Max Pooling (GMP) to aggregate spatial information into two descriptors, which are then processed by a shared multilayer perception to generate the CAM.

$$M_c(F) = \sigma(MLP(GAP(F) + MLP(GMP(F))) \quad (3)$$

where σ denotes the sigmoid activation function, F is the input feature map, and $M_c(F)$ is the learned channel attention mask that emphasizes informative feature channels.

Subsequently, the SAM focuses on where to emphasize by aggregating channel information using both GAP and GMP operations, followed by a convolutional layer with a 7×7 kernel and a sigmoid activation to produce a spatial attention map.

$$M_s(F) = \sigma\big(ConV_{7\times7}([AvgP(F); MaxP(F)])\big) \quad (4)$$

Finally, the refined feature map is obtained by sequentially applying channel and SAM to the input feature map as:

$$F' = M_s(F) \odot M_c(F) \odot F \quad (5)$$

where $\odot$ denotes element-wise multiplication. This sequential attention mechanism effectively suppresses irrelevant background features while enhancing discriminative representations, thereby improving the recognition and classification of hand gestures.

D. Attention Mechanism

The AM is a concept that allows NN to selectively focus on the most relevant parts of input features while processing information [25, 26]. Theoretically, it is based on human visual attention, in which the brain focuses on significant areas rather than processing all details equally. Mathematically, attention is represented as a weighted sum of input features, with the weights indicating the importance of each element. Given an input sequence $x = [x_1, x_2, \ldots, x_n]$the attention output as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

where Q, K and V represent the Query, Key, and Value matrices, and d_k are the dimensionality of the key vectors used for normalization. The softmax ensures that all attention weights sum to one, forming a probability distribution over the input elements. This helps the model focus on relevant information and ignore irrelevant details, thereby improving the representation of contextual dependencies and enhancing overall accuracy and the model's ability to generalize.

III. PROPOSED APPROACH

This study proposes a two-stream CNN attention architecture designed explicitly for multi-class hand gestures. As shown in Figure 1, the input images are fed into Blocks A and C, which serve as initial feature-extraction layers, as described in Equation 1. Block A contains two convolutional layers (32 and 32 filters) with 3×3 kernels and 2×2 MaxPooling, taking 50×50-pixel input images. The extracted features are input into Block B, which was designed with two convolutional layers comprising 64 filters each, a 3×3 kernel, and a 2×2 MaxPooling operation.

Additionally, a residual block, as described in Equation 2, is incorporated to retain the essential features identified in Block A and to enhance the network's learning efficiency. The result is then processed by a 64-filter convolutional layer with a 3×3 kernel, followed by a GAP layer that converts spatial feature maps into compact feature representations. In the other branch, Block C has two convolutional layers (64 filters each) with a 3×3 kernel and 2×2 MaxPooling, processing 100×100 pixel input images. The features extracted are then passed to Block D, which contains two convolutional layers (64 filters each) with a 3×3 kernel and 2×2 MaxPooling, combined with residual layers, as described in Equation 2, that receive feature maps from Block C. The output is then processed through a 128-filter convolutional layer and the CBAM, as described in Equations 3, 4, and 5. The CBAM modules act as spatial attention mechanisms, using GAP and GMP to capture the average and most essential features of the feature maps, allowing the model to effectively focus on the most relevant regions of the image. Attention-refined features are then passed to the final GAP layer to produce the overall spatial representations before proceeding to the concatenation stage. The outputs from Blocks B and D are then merged through a concatenation operation (Concat), combining feature maps from both branches. In this context, the goal is to achieve feature diversity, including spatial and channel features, so that when learning new representations, the network can improve overall performance by learning comprehensive representations from various perspectives. Then, the merged features are fed into the AM, as described in Equation 6; this process emphasizes important features and suppresses unimportant ones. This means they automatically assign adaptive weights to features that help focus on spatial regions and channels during the network's learning process. After this, the attention-refined features are fed into fully connected layers comprising two dense layers (with 512 and 128 nodes) with ReLU activations, followed by a softmax for final image classification and recognition.

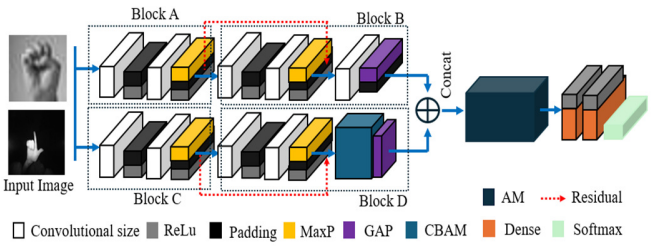


Fig. 1. Proposed method.

IV. EXPERIMENTAL PART

A. Dataset

In this study, the Hand-Sign-Image Dataset (HSID), which consists of 24 classes excluding J and Z due to their reliance on

motion [27]. The dataset initially contained 27,455 grayscale images, as shown in Figure 2. To cover the entire alphabet, the dataset was augmented with 3,592 images of J and Z from [28], resulting in a total of 31,047 images across all 26 classes. Additionally, they included the Hand Gesture Dataset (HGD) [29], which contains 38,219 images across 10 classes (see Figure 3). The combined datasets were split using TensorFlow into an 80% training set (60% for training and 20% for validation) and a 20% testing set.

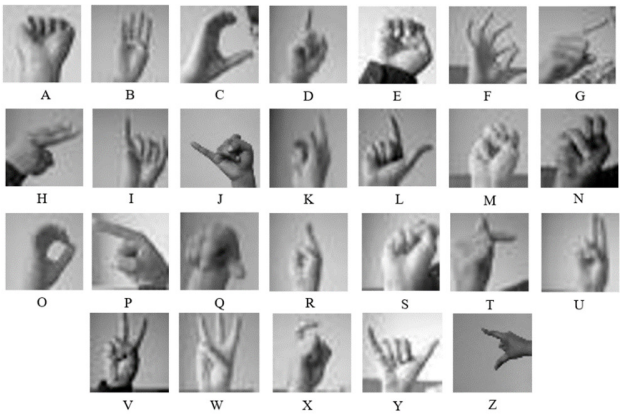


Fig. 2. HSID sample images.

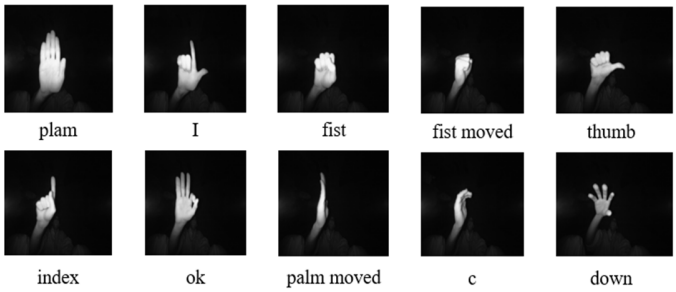


Fig. 3. HGD sample images.

TABLE I. TRAINING PARAMETERS

Parameter	Value
Image size	100×100×3, 50×50×3
Learning rate	10^{-3}, 10^{-4}, 10^{-5}
Epoch	30
Batch size	64
Loss function	Sparse categorical crossentropy
Optimization	Adam

B. Experimental Setup

The experiments were conducted on a Windows-based system equipped with an Intel Core i5-12400F processor and 32 GB of DDR5 RAM running at 5600 MHz. The proposed network was trained using an NVIDIA RTX 4070 GPU with 12 GB of VRAM and 5,888 CUDA cores. The implementation relied on NumPy and TensorFlow as the primary tools for model design and training. All training and testing parameters were standardized to ensure consistency, with details summarized in Table I.

V. EVALUATION

This study measures accuracy, precision, recall, F1, and AUC in Equations 7, 8, 9, 10, and 11 to assess effectively the model differentiates hand gestures. It also indicates learning efficiency and is evaluated alongside loss to track error reduction during training.

$$Accuracy = \frac{TP+FN}{TP+FP+FN+TN} \quad (7)$$

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

$$F1 = \frac{Precision \times Recall}{Precision+Recall} \quad (10)$$

$$AUC = \int_0^1 TP(FP)d(FP) \quad (11)$$

when true positive *(TP)* occurs when the model correctly identifies a hand gesture, true negative *(TN)* is when the model correctly identifies that no specific hand gesture is present, false positive *(FP)* is when the model incorrectly predicts a hand gesture that isn't there and false negative *(FN)* is when the model fails to detect an actual hand gesture and misclassified.

A. Training Results

The training results of the proposed network using HSID, as depicted in Figure 4(a), indicate that 10^{-3} results in successful learning, reaching a training accuracy of 96.98%. However, this is lower than the 100% accuracy obtained with 10^{-4}, while 10^{-5} yields the lowest accuracy at 85.70%. The validation accuracy in Figure 4(b) also confirms that 10^{-4} delivers the best performance, highlighting its suitability for enabling the model to effectively learn and retain gesture patterns. This is further supported by the loss comparison in Figure 4(c), where 10^{-4} yields the lowest loss of 4.8532×10^{-3}, indicating better learning stability. Figure 5(a) shows that while a lower Learning Rate (LR) increases training time, 10^{-3} completes in just 2 minutes and 33 seconds, and 10^{-5} takes 3 minutes and 38 seconds, 10^{-4} strikes an optimal balance between training efficiency and time. Overall, the results showed that 10^{-4} is the most effective LR for the proposed network when trained with noise-including hand gesture data. Additionally, the training results of the proposed network on the HGD, as shown in Figure 4(d), indicate that higher LR values of 10^{-3} and 10^{-4} enable the model to learn faster during the initial training phase, but its final accuracy is slightly lower than that with 10^{-5}. The 10^{-5} achieves the highest accuracy at 98.68%. In Figure 4(e), the validation accuracy peaks at 99.69% with 10^{-4}. Figure 4(f) shows that the loss consistently decreases across all LR, with 10^{-5} yielding the lowest loss of 0.0276, indicating better prediction stability than 10^{-3}, which has the highest loss of 0.0425. Figure 5(b) shows that lower LR results in longer training, 10^{-3} completes in 6 minutes and 19 seconds, while 10^{-5} takes the longest at 7 minutes and 59 seconds. These findings suggest that smaller LR improves model precision but requires more training time due to slower weight adjustments.

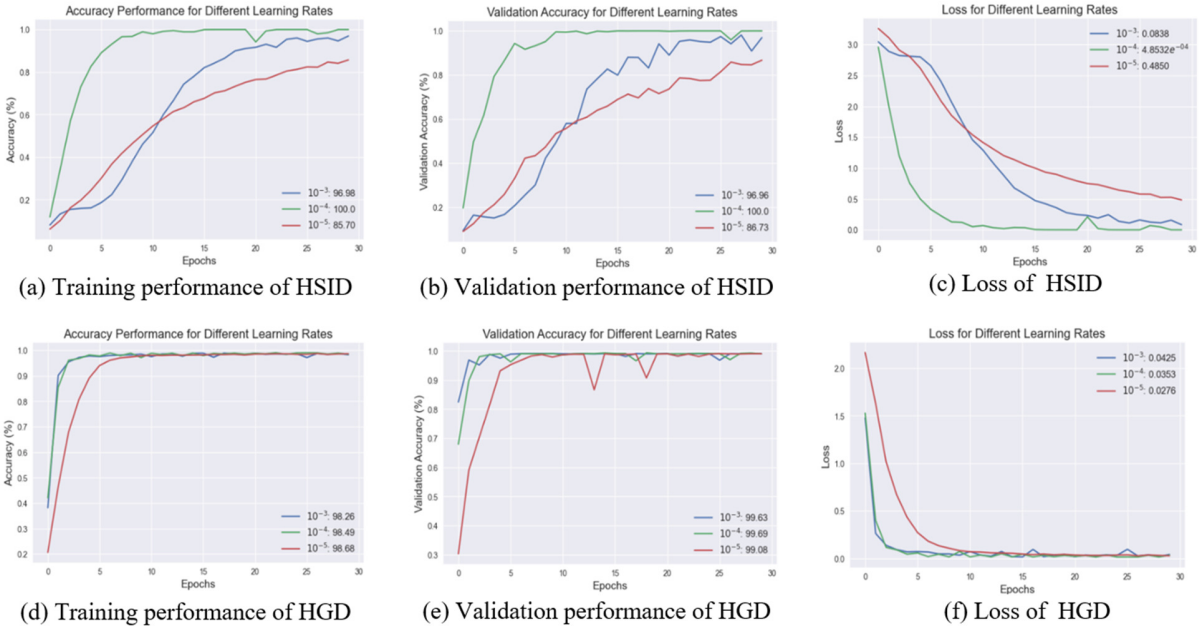


Fig. 4. Comparative training performance.

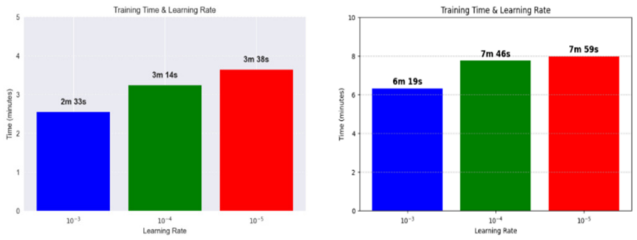


Fig. 5. Training time of HSID and HGD.

In Table II, the results from the HSID and HGD datasets demonstrate the model's exceptional performance during both training and testing with LR at 10^{-4}, achieving the highest training accuracy. For the HSID, all metrics were 100%, confirming perfect classification; for the HGD, precision, recall, and F1 were 99.25%, and AUC was 99.99%, thus assuring excellent generalization and stable convergence during training. The proposed model achieved accuracies of 95.09% and 98.98% on the HSID and HGD datasets, respectively, during testing, confirming its robustness and adaptability to unseen data.

B. Recognition Results

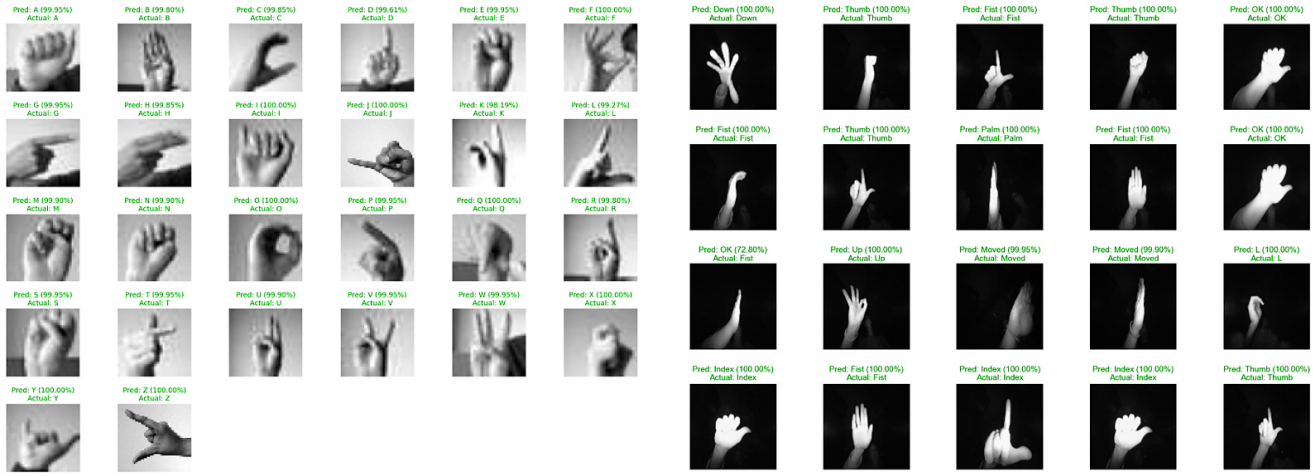
In Figure 6(a), the HSID for recognition of alphabet letters shows strong performance, with accuracy ranging from 98% to 100%. This highlights the model's ability to learn and effectively distinguish the unique features of each gesture. Overall, the results confirm that the proposed network exhibits high accuracy and strong potential for reliable hand gesture recognition. Furthermore, as shown in Figure 6(b), the HGD recognizes most gestures, achieving up to 100.00% accuracy in several cases where the predicted and actual gestures match perfectly. However, there are still a few instances of misrecognition, for example, the model predicting "OK" instead of the actual "Fist." These errors suggest that the model can occasionally struggle with gestures that share similar visual characteristics, particularly under challenging lighting conditions or viewpoints where features overlap.

TABLE II. EXPERIMENTAL PERFORMANCE RESULT

Dataset	**Training (%)**				**Testing (%)**
	Precision	Recall	F1	AUC	Acc
HSID	100.0	100.0	100.0	100.0	95.09
HGD	99.25 %	99.25	99.25	99.99	98.98

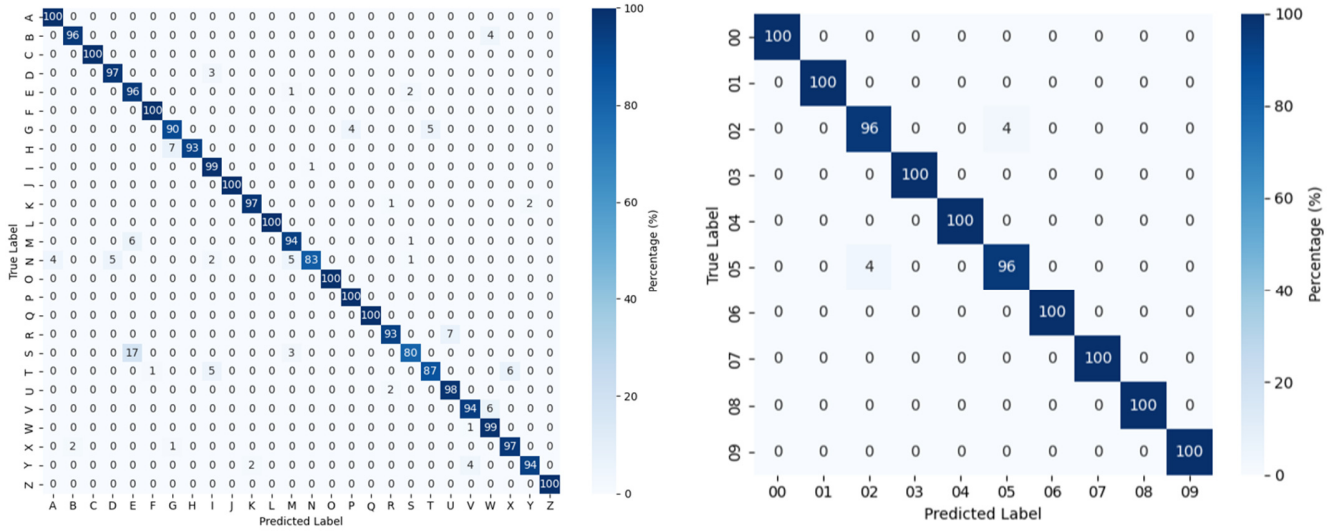
C. Classification Results

In Figure 7, the confusion matrices of HSID and HGD show that most classes were correctly classified. For HSID, as shown in Figure 7(a), the strong diagonal line indicates that the model correctly predicted most gesture classes. For HGD, as shown in Figure 7(b), the nearly perfect diagonal pattern shows that the model is stable and reliable in recognizing different hand gestures.



(a) Results of HSID recognition. (b) Results of HGD recognition.

Fig. 6. Recognition performance of HSID and HGD.



(a) HSID result (b) HGD result

Fig. 7. Confusion matrix of HSID and HGD.

D. Comparison Method

In this section, previous studies are compared, as shown in Table III. Earlier research has employed various DL techniques for hand gesture recognition, such as combining CWT with TCN for VR interactions, which is still underexplored [9]; using CNN-based architectures like VGG16, ResNet18, and InceptionV3 [13, 15]; integrating fuzzy logic with CNN transfer learning [17]; applying hybrid CNN–RNN frameworks [18]; and adopting reinforcement learning models such as DQN and DDQN for gesture control [30]. These approaches have achieved significant progress in improving recognition and classification accuracy. In contrast, the two-stream CNN, which integrates RB and CBAM to emphasize critical spatial–channel features, allows the network to focus on discriminative gesture regions. The proposed model attains accuracies of 98.68% and 100%, demonstrating that the proposed model is not only competitive with existing methods but also capable of delivering superior performance for diverse hand gesture data, demonstrating its robustness and adaptability across different applications In addition, the two-stream network captures both fine and global information from various input sizes, providing better generalization and higher recognition and classification accuracy than the single-stream network [6, 9, 14-16, 18, 31-33].

VI. DISCUSSION

The proposed two-stream CNN attention model employs dual-scale feature extraction and an adaptive attention mechanism to handle varying input sizes. This study uses

50×50 and 100×100 hand gesture images, applying convolutional layers, pooling, and RB to capture both detailed and overall features. The CBAM module enhances both spatial and channel attention, while concatenation merges the two streams to increase feature diversity. The AM module reweights important features before classification using dense layers with ReLU and softmax activations. The proposed method aligns with those presented in [9] and [34].

The model achieved 98.68% and 100% accuracy on the HGD and HSID datasets, respectively, surpassing or nearly matching previous approaches such as DACNN [6], CNN–RNN TL [18], and ResNet18 with SVM [31]. These results demonstrate that the proposed method generalizes effectively and maintains stable learning behavior across different visual features within each dataset. Additionally, the Grad-CAM heatmaps, shown in Figure 8, visually illustrate how the network makes its decisions. The highlighted regions primarily cover the fingers and palms, indicating that the model correctly focuses on key spatial areas defining each gesture. This interpretability enhances the reliability of the proposed architecture and supports its potential application in real-time gesture recognition and embedded systems.

TABLE III. COMPARISON OF DEEP LEARNING MODEL

Paper	Model	Result
[6]/ 2024	DACNN	99%
[7]/ 2021	Mask-RCNN with GH	88.16 and 88.19%
[8]/ 2022	ALR-CNN	95.50, 95.85, and 85.58%
[9]/ 2024	CNN combined with CWT and TCN	98.73%
[13]/2024	VGG16	96%
[14]/ 2022	ResNet18	96.62, 93.85, and 93.73%
[15]/ 2024	InceptionV3	83.66%
[16]/ 2023	Deep Transformer Network	88.22 and 99.10%
[17]/ 2024	Fuzzy Logic and CNN Transfer Learning	98.78, 97.94, 98.36, 90.46, and 98.34%
[18]/ 2025	CNN, RNN with Transfer Learning	99.01 and 96.84%
[30]/ 2024	ResNet18 with SVM	99.16, 90.88, and 91.66%
[31]/ 2024	DQN and DDQN in HGR	83.26, 82.87, 91.95, and 91.98%
Proposed	Two-stream CNN with AM	98.68% and 100%

VII. CONCLUSION

This research proposed a two-stream Convolutional Neural Network (CNN) based on the Residual Block (RB), Convolutional Block Attention Module (CBAM), and Attention Module (AM) for detecting and classifying hand gestures. The architecture is designed to obtain rich spatial and relational features using convolutional operators, RB, and a two-stream representation. The addition of CBAM also enhances the network's ability to focus on information-rich regions by sequentially applying channel and spatial attention, enabling it to target important features and selectively reduce background noise. Moreover, the AM combines the functionalities of all the modules to support the integration of coherent feature interactions and improve the overall efficiency of the network. The experimental results show that the model trained on different gesture categories accurately identified more instances than the baseline models did on two datasets. In this context, the proposed model achieved accuracies of 100.0% on the Hand-Sign-Images Dataset (HSID) and 98.69% on the Hand Gesture Dataset (HGD), outperforming or matching the performance of standard DL approaches.

(a) Grand-Cam heatmap of HSID

(b) Grand-Cam heatmap of HGD

Fig. 8. Grand-Cam heatmap of hand gesture.

In the future, they recommend enhancing the network to support real-time applications by optimizing its size for deployment on embedded systems with lightweight architectures. Additionally, further evaluation using larger and more diverse datasets is suggested to reduce potential overfitting and improve model generalization, as well as to compare different Spatial Attention Block (SAB) techniques to examine their impact on performance and training efficiency.

REFERENCES

[1] S. Yadav and S. Jain, "Gesture Recognition System for Human-Computer Interaction using Computer Vision," in *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Mar. 2024, pp. 1–4, https://doi.org/10.1109/ICRITO61523.2024.10522212.

[2] A. S. Editya, N. Kurniati, M. M. Alamin, A. Lisdiyanto, and A. L. Pramana, "Realtime of Hand Gesture Recognition for Telerobotics Controller Based Leap Motion Using Random Forest," in *2023 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, Aug. 2023, pp. 764–768, https://doi.org/10.1109/ICAMIMIA60881.2023.10427773.

[3] J. Li, C. Li, J. Han, Y. Shi, G. Bian, and S. Zhou, "Robust Hand Gesture Recognition Using HOG-9ULBP Features and SVM Model," *Electronics*, vol. 11, no. 7, Jan. 2022, Art. no. 988, https://doi.org/10.3390/electronics11070988.

[4] M. A. A. Razak, F. Y. A. Rahman, R. Mohamad, S. Shahbuddin, Y. W. M. Yusof, and S. I. Suliman, "Hand Gesture Recognition based on

Convolution Neural Network (CNN) and Support Vector Machine (SVM)," in *2023 IEEE 14th Control and System Graduate Research Colloquium (ICSGRC)*, Dec. 2023, pp. 123–126, https://doi.org/10.1109/ICSGRC57744.2023.10215427.

[5] H. G. Doan and N. T. Nguyen, "Fusion Machine Learning Strategies for Multi-modal Sensor-based Hand Gesture Recognition," *Engineering, Technology & Applied Science Research*, vol. 12, no. 3, pp. 8628–8633, June 2022, https://doi.org/10.48084/etasr.4913.

[6] A. O. Hashi, S. Z. M. Hashim, and A. B. Asamah, "Dynamic Adaptation in Deep Learning for Enhanced Hand Gesture Recognition," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 15836–15841, Aug. 2024, https://doi.org/10.48084/etasr.7670.

[7] F. S. Khan, M. N. H. Mohd, D. M. Soomro, S. Bagchi, and M. D. Khan, "3D Hand Gestures Segmentation and Optimized Classification Using Deep Learning," *IEEE Access*, vol. 9, pp. 131614–131624, 2021, https://doi.org/10.1109/ACCESS.2021.3114871.

[8] A. Fatayer, W. Gao, and Y. Fu, "sEMG-Based Gesture Recognition Using Deep Learning From Noisy Labels," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 9, pp. 4462–4473, Sept. 2022, https://doi.org/10.1109/JBHI.2022.3179630.

[9] T. D. Qi, F. L. Cibrian, M. Raswan, T. Kay, H. M. Camarillo-Abad, and Y. Wen, "Toward Intuitive 3D Interactions in Virtual Reality: A Deep Learning- Based Dual-Hand Gesture Recognition Approach," *IEEE Access*, vol. 12, pp. 67438–67452, 2024, https://doi.org/10.1109/ACCESS.2024.3400295.

[10] L. Qu, H. Wu, T. Yang, L. Zhang, and Y. Sun, "Dynamic Hand Gesture Classification Based on Multichannel Radar Using Multistream Fusion 1-D Convolutional Neural Network," *IEEE Sensors Journal*, vol. 22, no. 24, pp. 24083–24093, Sept. 2022, https://doi.org/10.1109/JSEN.2022.3216604.

[11] A. Osman Hashi, S. Zaiton Mohd Hashim, and A. Bte Asamah, "A Systematic Review of Hand Gesture Recognition: An Update From 2018 to 2024," *IEEE Access*, vol. 12, pp. 143599–143626, 2024, https://doi.org/10.1109/ACCESS.2024.3421992.

[12] S. Jiang, P. Kang, X. Song, B. P. L. Lo, and P. B. Shull, "Emerging Wearable Interfaces and Algorithms for Hand Gesture Recognition: A Survey," *IEEE Reviews in Biomedical Engineering*, vol. 15, pp. 85–102, 2022, https://doi.org/10.1109/RBME.2021.3078190.

[13] V. Madaan, N. Sharma, D. Gupta, and S. Aluvala, "Hand Gesture Detection Using VGG-16," in *2024 International Conference on Information Science and Communications Technologies (ICISCT)*, Aug. 2024, pp. 491–495, https://doi.org/10.1109/ICISCT64202.2024.10956830.

[14] Y. Gu *et al.*, "WiGRUNT: WiFi-Enabled Gesture Recognition Using Dual-Attention Network," *IEEE Transactions on Human-Machine Systems*, vol. 52, no. 4, pp. 736–746, Dec. 2022, https://doi.org/10.1109/THMS.2022.3163189.

[15] D. R. T. Hax, P. Penava, S. Krodel, L. Razova, and R. Buettner, "A Novel Hybrid Deep Learning Architecture for Dynamic Hand Gesture Recognition," *IEEE Access*, vol. 12, pp. 28761–28774, 2024, https://doi.org/10.1109/ACCESS.2024.3365274.

[16] M. Garg, D. Ghosh, and P. M. Pradhan, "Multiscaled Multi-Head Attention-Based Video Transformer Network for Hand Gesture Recognition," *IEEE Signal Processing Letters*, vol. 30, pp. 80–84, 2023, https://doi.org/10.1109/LSP.2023.3241857.

[17] M. A. A. Mosleh and A. H. Gumaei, "An Efficient Bidirectional Android Translation Prototype for Yemeni Sign Language Using Fuzzy Logic and CNN Transfer Learning Models," *IEEE Access*, vol. 12, pp. 191030–191045, 2024, https://doi.org/10.1109/ACCESS.2024.3512455.

[18] E. Yenisari and S. Yavuz, "Deep Learning-Based Sign Language Recognition Using Efficient Multi-Feature Attention Mechanism," *IEEE Access*, vol. 13, pp. 126684–126699, 2025, https://doi.org/10.1109/ACCESS.2025.3586096.

[19] J.-W. Choi, C.-W. Park, and J.-H. Kim, "FMCW Radar-Based Real-Time Hand Gesture Recognition System Capable of Out-of-Distribution Detection," *IEEE Access*, vol. 10, pp. 87425–87434, 2022, https://doi.org/10.1109/ACCESS.2022.3200757.

[20] M. Jaiswal, V. Sharma, A. Sharma, S. Saini, and R. Tomar, "Quantized CNN-based efficient hardware architecture for real-time hand gesture recognition," *Microelectronics Journal*, vol. 151, Sept. 2024, Art. no. 106345, https://doi.org/10.1016/j.mejo.2024.106345.

[21] Q. Shangguan *et al.*, "A Lightweight CNN Approach for Hand Gesture Recognition via GAF Encoding of A-Mode Ultrasound Signals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 33, pp. 3734–3743, 2025, https://doi.org/10.1109/TNSRE.2025.3608180.

[22] X. Dai, Z. Zhang, G. Liu, and J. Cai, "Application of CNN-LSTM feature fusion architecture based on self-made data glove in gesture recognition," *Biomedical Signal Processing and Control*, vol. 112, Feb. 2026, Art. no. 108573, https://doi.org/10.1016/j.bspc.2025.108573.

[23] M. Zakariah and A. Alnuaim, "Recognizing human activities with the use of Convolutional Block Attention Module," *Egyptian Informatics Journal*, vol. 27, Sept. 2024, Art. no. 100536, https://doi.org/10.1016/j.eij.2024.100536.

[24] B. Niu, J. Li, and Y. Wang, "A Simplified Convolutional Block Attention Module for Robust Hand Gesture Recognition with High Density Surface Electromyography," in *2025 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, July 2025, pp. 1–7, https://doi.org/10.1109/AIM64088.2025.11175815.

[25] J. P. Sahoo, S. P. Sahoo, S. Ari, and S. K. Patra, "Hand Gesture Recognition Using Densely Connected Deep Residual Network and Channel Attention Module for Mobile Robot Control," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, 2023, Art. no. 5008011, https://doi.org/10.1109/TIM.2023.3246488.

[26] Y.-Y. Yang, H.-H. Yang, and J.-K. Yang, "YOLO-LRHG: Long Range Hand Gesture detection using YOLO with attention mechanism," in *2024 IEEE 7th International Conference on Electronic Information and Communication Technology (ICEICT)*, July 2024, pp. 985–990, https://doi.org/10.1109/ICEICT61637.2024.10671140.

[27] A. Singh. "Hand-Sign-Images." Kaggle. https://www.kaggle.com/datasets/ash2703/handsignimages

[28] A. Khan. "Sign Language Gesture Images Dataset." Kaggle. https://www.kaggle.com/datasets/ahmedkhanak1995/sign-language-gesture-images-dataset

[29] T. Mantecón, C. R. del-Blanco, F. Jaureguizar, and N. García, "Hand Gesture Recognition Using Infrared Imagery Provided by Leap Motion Controller," in *Advanced Concepts for Intelligent Vision Systems*, Cham, 2016, pp. 47–57, https://doi.org/10.1007/978-3-319-48680-2_5.

[30] C. G. Bastidas, K. A. Pérez, Á. L. V. Caraguay, L. I. B. López, and M. E. Benalcázar, "Comparison of Hand Gesture Recognition Models Combining Supervised Learning and Reinforcement Learning," in *2024 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, Aug. 2024, pp. 1–11, https://doi.org/10.1109/LA-CCI62337.2024.10814786.

[31] R. Sebbah and F. Z. Chelali, "Using Machine Learning and Deep Learning for Static and Dynamic Hand Gesture Recognition," in *2024 International Conference on Advances in Electrical and Communication Technologies (ICAECOT)*, July 2024, pp. 1–6, https://doi.org/10.1109/ICAECOT62402.2024.10828747.

[32] A. Dey, S. Biswas, and L. Abualigah, "Umpire's Signal Recognition in Cricket Using an Attention based DC-GRU Network," *International Journal of Engineering*, vol. 37, no. 4, pp. 662–674, 2024, https://doi.org/10.5829/IJE.2024.37.04A.08.

[33] A. Dey, S. Biswas, and D.-N. Le, "Recognition of Wh-Question Sign Gestures in Video Streams using an Attention Driven C3D-BiLSTM Network," *Procedia Computer Science*, vol. 235, pp. 2920–2931, Jan. 2024, https://doi.org/10.1016/j.procs.2024.04.276.

[34] A. S. M. Miah, Md. A. M. Hasan, and J. Shin, "Dynamic Hand Gesture Recognition Using Multi-Branch Attention Based Graph and General Deep Learning Model," *IEEE Access*, vol. 11, pp. 4703–4716, 2023, https://doi.org/10.1109/ACCESS.2023.3235368.