https://doi.org/10.53964/mit.2025001

https://www.innovationforever.com

Modern Intelligent Times

ISSN 2957-7942 (Online)

Open Access

Research Article

GestureTransformer: A Hybrid CNN-Transformer Model for Hand Gesture Recognition in Smart Educational Environments

Jielin Yang[1], Wanyi Li[1*], Mingqi Zheng[1]

[1]School of Computer Science, Guangdong University of Education, Guangzhou, Guangdong Province, China

[*]**Correspondence to: Wanyi Li, Ph.D, Asssociate Professor,** School of Computer Science, Guangdong University of Education, Guangzhou, 510303, Guangdong Province, China; Email: luther1212@163.com

Received: May 24, 2025 **Revised:** June 25, 2025 **Accepted:** June 30, 2025 **Published:** July 7, 2025

Abstract

Objective: With the increasing adoption of digital technologies in modern classrooms, there is a growing demand for intuitive and contactless modes of human-computer interaction. Traditional input methods such as keyboards and mice are often unsuitable in dynamic or inclusive educational environments. This study aims to address this need by developing a high-precision, real-time hand gesture recognition system, designed specifically for smart classroom applications. The goal is to empower educators and students to interact with digital content seamlessly through natural hand movements, thereby promoting engagement, accessibility, and efficiency in teaching and learning processes.

Methods: We propose GestureTransformer, a hybrid deep learning architecture that integrates Convolutional Neural Networks (CNNs) for effective local spatial feature extraction and Transformer modules with multi-head self-attention for modeling global semantic dependencies. A custom dataset of static hand gestures was constructed to support supervised training and evaluation. Additionally, a gesture-to-keyboard mapping system was developed to translate recognized gestures into predefined control commands, enabling hands-free operation of educational software tools.

Results: Experiments demonstrate that GestureTransformer achieves a classification accuracy of 98.79%, significantly outperforming several baseline models, including SimpleCNN, MiniVGG, Residual Neural Network (ResNet)-9, and MLPClassifiers. The model exhibits stable convergence and strong generalization, as shown by consistent training and validation performance. The integration with the gesture-mapping interface supports real-time feedback and system responsiveness, making it well-suited for both physical and virtual classroom settings. The confusion matrix analysis confirms the system's discriminative ability.

Conclusion: GestureTransformer effectively combines the strengths of CNNs and Transformers to deliver robust and accurate hand gesture recognition. Its integration with a gesture-driven control interface promotes hands-free interaction and accessibility, offering valuable applications in inclusive education and intelligent classroom management.

Keywords: gesture recognition, smart classroom, human-computer interaction, transformer, educational technology

https://doi.org/10.53964/mit.2025001

Citation: Yang J, Li W, Zheng M. GestureTransformer: A Hybrid CNN-Transformer Model for Hand Gesture Recognition in Smart Educational Environments. *Mod Intell Times*, 2025; 3: 1. DOI: 10.53964/mit.2025001.

1 INTRODUCTION

With the rapid advancement of computer vision and artificial intelligence, human-computer interaction (HCI) has evolved toward more intuitive, efficient, and contactless modalities[1,2]. Among these, hand gesture recognition has emerged as a critical technique for enabling natural interactions in diverse applications, including virtual reality, sign language translation[3], and smart education systems. In intelligent classrooms, non-intrusive gesture-based interfaces offer the potential to enhance engagement and interactivity, especially in remote or touchless teaching environments.

Convolutional Neural Networks (CNNs) have long been the backbone of visual recognition tasks due to their ability to extract hierarchical spatial features through local receptive fields and weight sharing. Since the introduction of LeNet[4], CNN architectures have progressed significantly. VGGNet[5] deepened networks using uniform 3×3 convolutions; GoogLeNet[6] introduced multi-scale Inception modules; and Residual Neural Network (ResNet)[7] mitigated gradient vanishing via residual connections, enabling the training of very deep networks. Lightweight models such as MobileNet and ShuffleNet have improved computational efficiency, making CNNs suitable for real-time and embedded scenarios.

CNNs have been widely adopted in gesture recognition due to their efficiency in extracting spatial features and their suitability for real-time applications. Numerous studies have demonstrated CNNs' effectiveness across visual and physiological gesture data. For instance, Tan et al.[8] introduced EDenseNet, an enhanced DenseNet model for static hand gesture classification. By improving feature propagation through dense connectivity and transition layer optimization, their model achieved 99.64% accuracy with data augmentation and 98.50% without on three benchmark datasets including ASL and NUS. Mujahid et al.[9] proposed a lightweight real-time gesture detector based on YOLOv3 and DarkNet-53, achieving 97.68% accuracy across five gesture classes. The model demonstrated strong robustness in cluttered environments and high inference speed without requiring extensive preprocessing. Chen et al.[10] converted surface electromyography signals into grayscale images and applied a multi-view CNN with asymmetric convolution kernels to enhance feature representation, reaching over 98% accuracy on multiple gesture tasks. Collectively, these studies reinforce CNNs' advantages in localized feature extraction, multi-scale modeling, and real-time deployability, making them highly suitable for gesture recognition across diverse scenarios.

However, CNNs are inherently constrained by their limited receptive fields and inductive biases toward local spatial patterns[11]. This restricts their ability to capture long-range dependencies or model temporal dynamics, which are essential for recognizing continuous or dynamic hand gestures in real-world scenarios. These limitations have led to increasing interest in architectures that can incorporate both local and global information.

Transformers, originally introduced for natural language processing by Vaswani et al. have been increasingly adapted for vision tasks due to their self-attention mechanism's ability to model long-range dependencies[12]. Vision Transformer (ViT), DEtection TRansformer[13], and Swin Transformer[14] demonstrated strong performance by treating images as sequences of patches and learning global representations.

To leverage the strengths of global context modeling, Garg et al.[15] proposed GestFormer, a lightweight, transformer-based network designed for dynamic gesture recognition. The computational effort is reduced while maintaining the temporal modelling capability by replacing attention with a non-parametric pooling-based token mixer and introducing a multiscale wavelet pooling mechanism. Besrour et al.[16] applied the Transformer model to inertial sensor data for dynamic gesture recognition and achieved high accuracy, surpassing several classical machine learning models. These results demonstrate the superior ability of Transformers to model complex temporal patterns in sensor-based data, strengthening their applicability in gesture-based human computer interaction systems. These advantages motivate us to integrate Transformer components into hybrid models to combine CNN's local feature extraction capabilities.

In light of this, hybrid architectures combining CNNs and Transformers have gained traction, aiming to leverage CNNs' efficiency in spatial feature extraction with Transformers' strength in global context modeling. Liu et al.[17]proposed CNN-ViT for EMG-based gesture recognition, which achieved strong generalization across datasets by integrating motion encoding mechanisms. Haq et al.[18] developed a CNN-Transformer hybrid model for gesture recognition that remains robust across various body postures and camera angles, achieving high accuracy in complex scenarios. These studies further demonstrate the superiority of the CNN-Transformer hybrid architecture. However, their work primarily focuses on recognition accuracy and sequence mapping, paying little attention to real-time interaction control or deployment in real educational settings.

To address these challenges, we propose Gesture-Transformer, a novel lightweight hybrid architecture combining CNN and Transformer components for accurate and real-time hand gesture recognition in intelligent

educational environments. While previous works have primarily focused on improving classification accuracy or architectural efficiency, they often lack integration with real-time interactive control systems and are rarely optimized for deployment in educational environments.In contrast, our work not only introduces a lightweight hybrid architecture,but also emphasizes end-to-end system design by mapping recognized gestures directly to executable commands, enabling practical use in smart classrooms.

The main contributions of this work are summarized as follows:

1.We propose GestureTransformer, a lightweight hybrid CNN-Transformer architecture that captures both local visual patterns and global dependencies, specifically tailored for gesture recognition.

2.We develop a real-time gesture-to-keyboard mapping system that translates predicted gesture classes into actual keyboard events, supporting closed-loop interaction for educational applications.

3.We demonstrate the deployability of the system on Windows platforms with high classification accuracy (98.79%) and efficient inference, validating its real-time performance in practical classroom-like scenarios.

These contributions collectively advance gesture-based HCI in the context of intelligent education, offering a feasible and deployable solution for real-time, touchless system control in smart classrooms.

The remainder of this paper is organized as follows: Section 2 presents the methods and materials, including model architecture and dataset description. Section 3 discusses the experimental results and comparative analysis. Section 4 provides an in-depth discussion, and Section 5 concludes the paper with future research directions.

2 METHODS AND MATERIALS

2.1 GestureTransformer Architecture

In this study, we propose a hybrid neural network architecture named GestureTransformer, which integrates the local feature extraction capabilities of CNNs with the global contextual modeling strength of the Transformer architecture. This hybrid design is tailored specifically for hand gesture recognition, addressing the dual need for fine-grained spatial understanding and semantic generalization across diverse gesture categories.The specific structure is shown in Figure 1.

The model begins with a feature extraction backbone composed of three convolutional layers, each followed by a max-pooling operation. These convolutional layers are responsible for detecting low-level image features, such as edges, contours, and textures, and for gradually building up mid-level and high-level semantic representations of the input gesture image. Max-pooling reduces the spatial dimensions of the feature maps while preserving dominant features, thereby lowering the computational cost and introducing translational invariance into the model.

Instead of dividing the image into patches as in ViT, we flatten the entire convolutional feature map into a single visual token, which is then projected into a high-dimensional embedding space suitable for Transformer processing, and finally format it as a sequence input compatible with Transformer architectures.

The core of the architecture is a Transformer module composed of two stacked layers of self-attention and feed-forward networks. Although it adopts the encoder-decoder format provided by PyTorch's nn.Transformer, we input the same feature sequence to both branches, effectively forming a simplified self-attention encoder. This design enables efficient global context modeling across the visual tokens.Each layer incorporates four-headed multi-head self-attention and feed-forward sublayers. Dropout is applied after the embedding layer to mitigate overfitting.

Each encoder layer enables interactions among all feature tokens, allowing the model to capture richer semantic relationships. The Transformer module's output thus encodes comprehensive global information from the entire input image, which is crucial for fine-grained gesture classification.

The output sequence of the Transformer encoder is passed through a global average pooling layer across the sequence dimension. This operation condenses the learned contextual information into a single, fixed-size global feature vector, reducing reliance on any particular spatial position and ensuring robustness against variations in hand position, size, or orientation.

Finally, the pooled features are fed into a fully connected layer that performs the classification task. This layer maps the high-level global features to a vector of probabilities, each corresponding to a predefined gesture class. The model outputs the class label with the highest probability as the predicted gesture.

2.2 Gesture-to-Keyboard Mapping System

To translate gesture recognition into practical HCI, we design and implement a gesture-to-keyboard mapping system that connects the output of the GestureTransformer model to real-time command execution on the Windows platform.

The system begins with real-time hand tracking and region extraction, using the MediaPipe Hands solution provided by Google. MediaPipe is a lightweight, high-precision framework for real-time hand landmark detection. It detects 21 3D hand keypoints from a live webcam feed and provides normalized coordinates for each joint. For each frame, the

https://doi.org/10.53964/mit.2025001

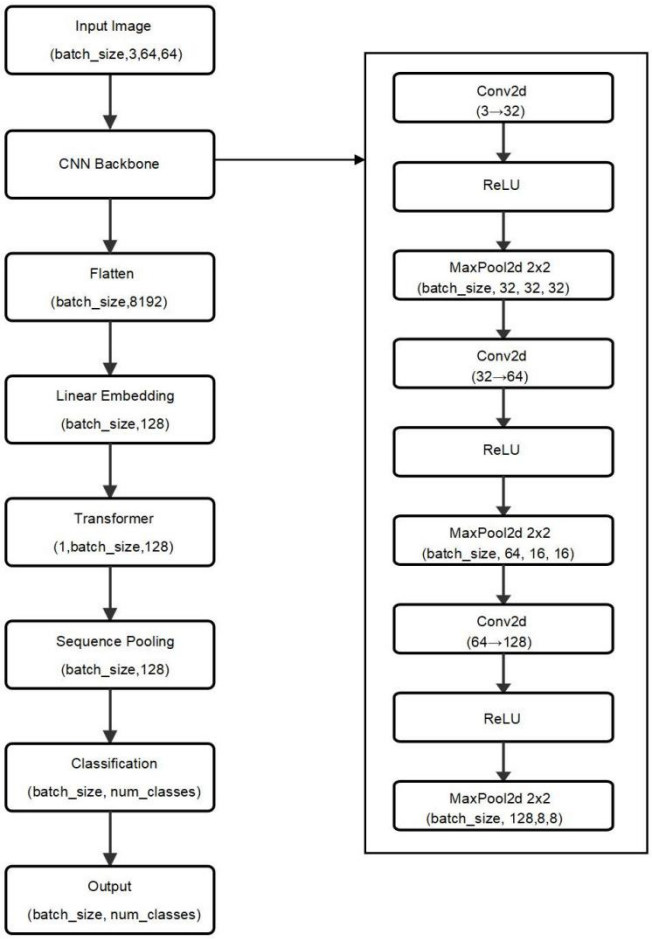

Figure 1. The network Architecture of GestureTransformer.

system calculates the minimum bounding rectangle that fully encompasses all detected hand keypoints. To ensure contextual integrity, this rectangle is slightly expanded in all directions, mitigating potential clipping of finger tips or palm edges due to minor detection inaccuracies.

This cropped hand region is resized and normalized to match the input requirements of the GestureTransformer model. Once passed through the model, a gesture class label is obtained. The system then maps this class label to a predefined command using a dictionary-style key mapping table (key_map), which associates each gesture category with a specific keyboard key or shortcut combination

After successful recognition, the system uses Python automation libraries—notably pyautogui and keyboard—to simulate the corresponding keystroke or hotkey. These libraries allow for high-level control of the keyboard and mouse without requiring administrative privileges. Before executing the simulated input, the system programmatically activates the target application window using process hooks or window handles, ensuring that the simulated input is directed to the correct application context. This solves the window focus problem, which commonly occurs in real-time input systems where the target window may not be in the foreground.

This system allows users to perform various actions, such as navigating slides, controlling media playback, launching applications, or typing frequently used phrases—all through intuitive, contactless hand gestures. Because it requires only a standard webcam and runs efficiently on common Windows PCs, the system exhibits strong applicability and extensibility in real-world environments.

2.3 Experimental Environment

All experiments were conducted on a personal computer running a 64-bit Windows 11 operating system. The Python environment was managed using Anaconda, and model training and evaluation were implemented with the PyTorch deep learning framework. Visual Studio Code was used as the integrated development environment for code development and model deployment.

2.4 Dataset

We constructed a hand gesture image dataset comprising 11 distinct gesture classes, with each class containing 2,400 images, yielding a total of 26,400 images. Each image captures a single gesture instance performed by different users under natural lighting conditions. The dataset is organized using PyTorch's ImageFolder structure, where

https://doi.org/10.53964/mit.2025001

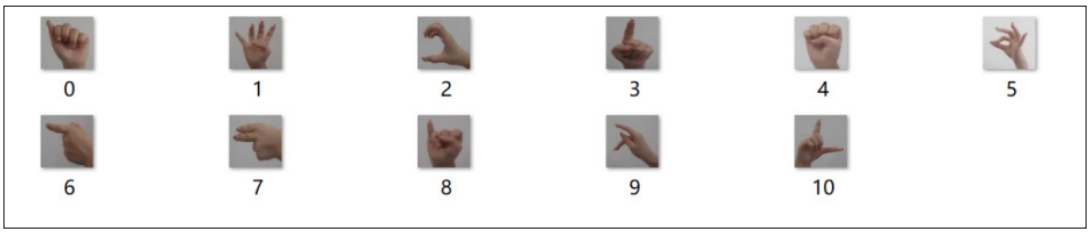

Figure 2. Sample Gesture Images from the Dataset, One Per Class.

Table 1. Dataset Partitioning

Dateset	Proportion	The number of images of each type (pieces)
Train_dataset	70%	1680
Val_dataset	10%	240
Test_dataset	20%	480

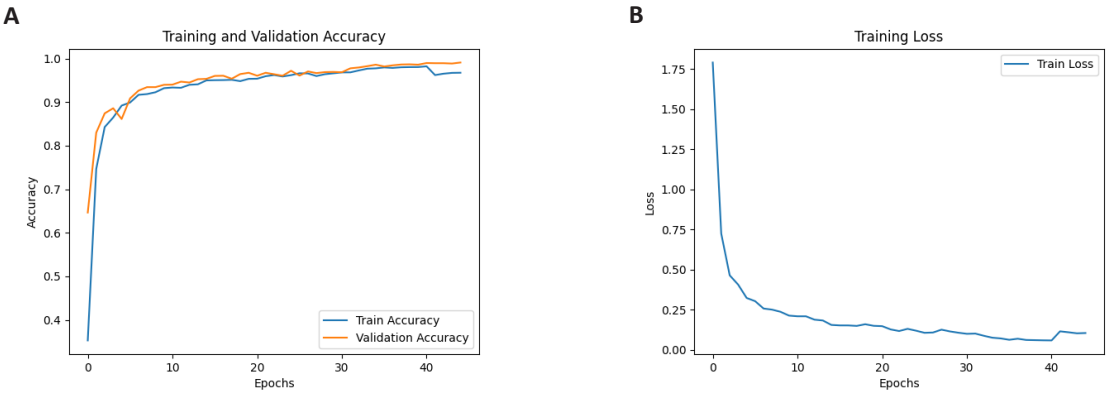

Figure 3. (A) Accuracy; (B) Training Loss.

each subdirectory represents one gesture class.

To provide a concrete illustration, Figure 2 displays a representative example from each gesture class. The dataset covers a diverse set of commonly used hand gestures that are intuitive and easy to recognize, such as a clenched fist, an open palm, a finger point, an "OK" sign, designed to reflect common interactive hand commands suitable for intelligent education systems.

All images were manually verified to ensure gesture clarity and class consistency. The dataset was then split into training, validation, and test subsets using a 7:1:2 ratio, implemented with PyTorch's random_split, ensuring that samples were non-overlapping across sets and uniformly distributed across classes.

The sample distribution across the three subsets is summarized in Table 1 ensuring class balance is maintained across all phases of training and testing.

Only the training set underwent data augmentation to enhance model robustness and simulate real-world variance. Augmentation operations included random rotation (±10°), resized cropping to 64×64 pixels, horizontal flipping, color jitter (brightness, contrast, saturation, hue), Gaussian blur, and random erasing.

The validation and test sets were left unaugmented to serve as unbiased measures of model generalization. All subsets were loaded using PyTorch's DataLoader with a batch size of 256 for training and 1024 for evaluation.

3 RESULTS

3.1 Performance Evaluation

The preprocessed training and validation sets were fed into the GestureTransformer model for training, and the best-performing model on the validation set was used for evaluation on the test set. The training configuration was as follows:The training was conducted with a batch size of 256 for the training set and 1024 for the validation and testing sets using the Adam optimizer with a learning rate of 0.001 the loss function was cross-entropy loss and the model was trained for a maximum of 100 epochs.

During training, both accuracy and loss curves were monitored in real time to ensure convergence stability and prevent overfitting. The training dynamics are visualized in Figure 3.

Figure 3A shows the accuracy curves for both the training and validation sets. The model achieves a rapid increase in accuracy within the first 10 epochs, followed by a stable convergence, demonstrating fast learning dynamics.

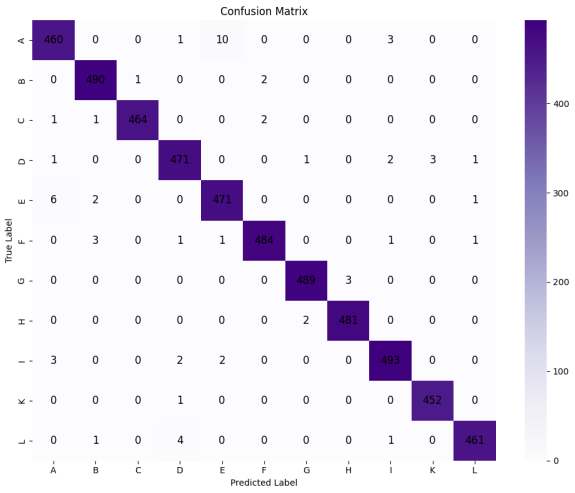

Figure 4. Confusion Matrix.

Throughout training, the training and validation accuracies remain closely aligned, indicating that the model does not suffer from overfitting or underfitting, and generalizes well across different data distributions.

Figure 3B shows the training loss curve, which follows a smooth and monotonic downward trajectory, indicating consistent gradient descent and absence of noisy optimization behavior. By the end of training, both the loss and accuracy stabilize, further affirming that the model does not overfit or underfit the data.

These results demonstrate the model's capability to learn complex gesture representations efficiently while maintaining generalization across unseen data.

To further evaluate the classification performance on the test set, Figure 4 presents the confusion matrix generated from the final predictions. As observed, most samples are correctly classified, with high density along the diagonal, indicating strong discriminative power across gesture categories. While a few misclassifications occur between certain similar classes, their frequency is relatively low and does not significantly impact overall accuracy.

3.2 Comparison with Baseline Models

To further assess the effectiveness and superiority of the proposed GestureTransformer, we conducted comparative experiments with four baseline models—SimpleCNN, MiniVGG, MLPClassifier, and ResNet-9—under strictly consistent training and evaluation settings.

SimpleCNN is a shallow convolutional network with three convolutional layers followed by max-pooling and fully connected layers. It is representative of fast, lightweight models suitable for embedded applications. MiniVGG is a simplified variant of the Visual Geometry Group(VGG) architecture, retaining its deep convolutional layers while reducing width and depth for efficiency. MLPClassifier lacks any spatial modeling capability, relying solely on dense layers and thus serves as a benchmark for non-convolutional approaches. ResNet-9, on the other hand, is a deeper architecture using residual blocks, and is expected to capture more complex features through identity mappings.

The evaluation metrics used include Precision, Recall, and F1-score, defined as follows: Precision measures the proportion of correct predictions among all instances predicted as a given class:

$$\text{Precision}=\frac{\text{True Positives}}{\text{True Positives}+\text{False Positives}} \quad (1)$$

Recall reflects the model's ability to correctly identify all actual instances of a given class:

$$Recall = \frac{\text{True Positives}}{\text{True Positives}+\text{False Negatives}} \quad (2)$$

F1-score is the harmonic mean of precision and recall, offering a balanced evaluation of classification performance:

$$F1-\text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

As shown in Table 2, the GestureTransformer significantly outperforms all baseline models. It achieves an overall accuracy of 98.79%, representing an improvement of approximately 2.4% over SimpleCNN (96.34%) and MiniVGG (96.16%). The performance gain is even more pronounced when compared to MLPClassifier (84.00%) and ResNet-9 (80.28%).

Despite the architectural diversity, GestureTransformer consistently outperformed all these baselines in terms of classification accuracy, precision, recall, and F1-score, as shown in Table 2. The accuracy gap of over 14% between GestureTransformer and ResNet-9 highlights that deeper CNNs alone are insufficient to model the global dependencies present in complex hand gestures. ResNet-9, while effective for general image tasks, may lose subtle contextual details during aggressive downsampling.

The comprehensive experiments presented in this section demonstrate the effectiveness of the proposed GestureTransformer in real - world gesture recognition tasks, with key findings including rapid convergence and stable training dynamics with minimal risk of overfitting, strong generalization capability confirmed by high accuracy on unseen test data, superior classification performance compared to multiple baseline models reflected in both overall accuracy and class - wise F1 - scores, and effective handling of gesture variability with low confusion between similar classes. By integrating CNNs and Transformers into a lightweight hybrid architecture, GestureTransformer strikes a balance between computational efficiency and expressive power, thus making it well - suited for real - time gesture - controlled systems.

https://doi.org/10.53964/mit.2025001

Table 2. Model Comparison

Class	Metric	GestureTransformer	SimpleCNN	MiniVGG	MLPClassifier	ResNet-9
A	Precision	0.9766	0.9549	0.9299	0.7683	0.8190
	Recall	0.9705	0.9119	0.9027	0.6489	0.5621
	F1-score	0.9735	0.9329	0.9161	0.7036	0.6667
	Support	474	488	514	470	475
B	Precision	0.9859	0.9767	0.9779	0.8155	0.6842
	Recall	0.9939	0.9618	0.9444	0.8405	0.8053
	F1-score	0.9899	0.9692	0.9609	0.8278	0.7398
	Support	493	523	468	489	452
C	Precision	0.9978	0.9890	0.9850	0.9463	0.8960
	Recall	0.9915	0.9912	0.9935	0.9443	0.8905
	F1-score	0.9946	0.9901	0.9893	0.9453	0.8933
	Support	468	455	464	485	484
D	Precision	0.9812	0.9444	0.9588	0.8433	0.8118
	Recall	0.9833	0.9325	0.9465	0.8187	0.7427
	F1-score	0.9823	0.9384	0.9526	0.8309	0.7757
	Support	479	474	467	480	482
E	Precision	0.9731	0.9256	0.9189	0.7715	0.6697
	Recall	0.9812	0.9664	0.9485	0.7286	0.7674
	F1-score	0.9772	0.9455	0.9335	0.7495	0.7153
	Support	480	476	466	468	473
F	Precision	0.9918	0.9836	0.9787	0.7940	0.8632
	Recall	0.9857	0.9678	0.9766	0.8783	0.8541
	F1-score	0.9888	0.9757	0.9777	0.8340	0.8587
	Support	491	497	471	452	473
G	Precision	0.9939	0.9629	0.9495	0.8579	0.8925
	Recall	0.9939	0.9873	0.9979	0.9035	0.8470
	F1-score	0.9939	0.9749	0.9731	0.8802	0.8692
	Support	492	473	471	508	549
H	Precision	0.9938	0.9872	0.9836	0.8985	0.7954
	Recall	0.9959	0.9706	0.9677	0.8685	0.8966
	F1-score	0.9948	0.9789	0.9756	0.8832	0.8430
	Support	483	477	495	479	464
I	Precision	0.9860	0.9250	0.9516	0.7984	0.7987
	Recall	0.9860	0.9427	0.9417	0.8067	0.7555
	F1-score	0.9860	0.9338	0.9466	0.8025	0.7765
	Support	500	471	480	481	499
K	Precision	0.9934	0.9772	0.9771	0.9198	0.8271
	Recall	0.9978	0.9874	0.9812	0.9008	0.8753
	F1-score	0.9956	0.9823	0.9791	0.9102	0.8505
	Support	453	477	478	484	481
L	Precision	0.9935	0.9725	0.9688	0.8147	0.8022
	Recall	0.9872	0.9808	0.9802	0.8905	0.8326
	F1-score	0.9903	0.9766	0.9745	0.8509	0.8171
	Support	467	469	506	484	448
Accuracy		0.9879	0.9634	0.9616	0.8400	0.8028
Support		5280	5280	5280	5280	5280

https://doi.org/10.53964/mit.2025001

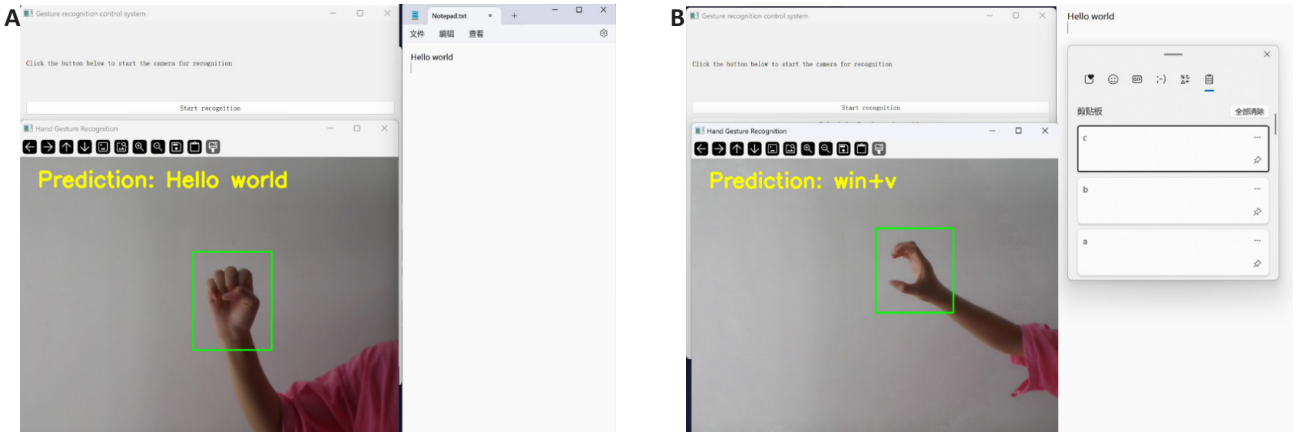

Figure 5. Gesture Classification Test Effect. (A) Map to Text Content; (B) Map to Shortcut Key.

We also observed that GestureTransformer achieves these gains without significantly increasing model complexity. The overall parameter count remains relatively small due to the shallow CNN backbone and lightweight Transformer encoder. This makes it feasible for deployment in real-time systems or resource-constrained environments such as mobile or embedded devices.

3.3 Real-world Demonstration

To validate the practicality of the proposed gesture recognition system in real-world applications, we implement a gesture-to-keyboard mapping mechanism and conduct demonstrations under two representative scenarios: text content mapping and shortcut key mapping, as shown in Figure 5.

Map to text content: When a gesture corresponding to a specific text string (e.g., "Hello world") is recognized, the system automatically focuses on the Notepad window and inputs the predefined text content. This verifies the system's ability to support natural language input through gestures.

Map to shortcut keys: When the recognized gesture is associated with a system shortcut (e.g., using win+v to open the clipboard), the corresponding key combination is triggered automatically. This demonstrates the system's capability to facilitate hands-free control of operating system functions or applications.

These examples illustrate that the developed software not only achieves accurate real-time gesture recognition but also supports seamless interaction with external applications. The gesture-to-keyboard mapping enables intuitive, non-contact control, showcasing its potential for smart HCI environments such as distance education, assistive technologies, and touchless control interfaces.

4 DISCUSSION

The experimental findings validate the effectiveness of the proposed GestureTransformer in enabling accurate and robust hand gesture recognition. The model's hybrid architecture—combining CNNs for fine-grained spatial feature extraction with Transformer modules for capturing global semantic context—effectively addresses the limitations of conventional CNN-based methods. Achieving a classification accuracy of 98.79%, which is critical for gesture-based interaction in intelligent educational environments.

Beyond classification metrics, the model exhibits stable convergence and resilience to overfitting, as evidenced by the consistent training curves and high validation accuracy. The confusion matrix analysis shows that most gesture classes were accurately identified, although occasional misclassifications occurred between visually similar categories. These errors highlight the inherent difficulty in distinguishing gestures with overlapping hand shapes or occluded fingers.

The proposed system also demonstrates strong real-world applicability through its gesture-to-keyboard mapping module, which enables real-time, contactless control on standard Windows platforms. This functional integration provides a complete pipeline from recognition to action, enhancing usability in classrooms where voice input may be disruptive or undesirable.

However, the model still has some limitations. For example, it is currently limited to static, one-handed gestures and does not support continuous gesture flow or two-handed interactions. Additionally, the dataset was collected in relatively controlled indoor conditions, which may restrict its application to different lighting, backgrounds, or hand shapes. While the system performs well in real time on PC-based platforms, its performance on low-power edge devices has not yet been evaluated.

These observations indicate both the effectiveness and boundaries of the current design, motivating future work to enhance generalization, extend gesture types, and improve system deployment flexibility.

5 CONCLUSION

This paper presents GestureTransformer, a lightweight hybrid architecture that combines CNNs and Transformers

for static hand gesture recognition. The model achieved a classification accuracy of 98.79% on a custom dataset, outperforming several baseline architectures including standalone CNNs and lightweight convolutional models. These results validate the effectiveness of integrating local feature extraction with global semantic modeling for fine-grained visual recognition tasks.

From an educational technology perspective, the proposed system provides a practical and intuitive interface for contactless interaction, contributing to more inclusive and interactive learning environments. By mapping recognized gestures to real-time keyboard inputs, the system enables seamless integration into existing educational software without the need for additional hardware or intrusive voice commands.

Nevertheless, the current study is constrained by its focus on static, single-hand gestures and the absence of real-world classroom deployment trials. To address these limitations, future research will explore dynamic gesture recognition using temporal sequence modeling, incorporate 3D hand pose estimation, and investigate sensor fusion approaches such as combining visual input with inertial or depth data. In addition, testing in authentic classroom settings and on embedded platforms will be conducted to evaluate robustness, usability, and latency in real-time educational scenarios.

These directions aim to expand the applicability of gesture-based interaction systems in smart learning environments, special education contexts, and remote or hybrid teaching platforms—advancing the vision of accessible, intelligent, and human-centered educational technologies.

Acknowledgements

This work is supported by the Project for the Development of Guangdong Province Philosophy and Social Science (No. GD25CTY14), the Educational Science Planning Project of Guangdong Province (No. 2023GXJK125), and the Guangdong Provincial Special Program in Key Areas for Higher Education Institutions (New Generation Electronic Information (Semiconductors)) (No. 2024ZDZX1040).

Conflicts of Interest

The authors declared no conflict of interest.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Author Contribution

Yang J wrote the manuscript and proposed the methodology. Li W conducted the data processing and analysis. Zheng M designed and performed the experimental validation. All authors participated in revising the manuscript, reviewed the final version, and approved its submission.

Abbreviation List

CNNs, Convolutional neural networks
HCI, Human-Computer Interaction
ViT, Vision Transformer
VGG, Visual Geometry Group
ResNet, Residual Neural Network

References

[1] Guo L, Lu Z, Yao L. Human-machine interaction sensing technology based on hand gesture recognition: A review. *IEEE T Hum-Mach Syst*, 2021; 51: 300-309.[DOI]

[2] Zholshiyeva L, Manbetova Z, Kaibassova D et al. Human-machine interactions based on hand gesture recognition using deep learning methods. *Int J Electr Comput*, 2024; 14: 741-748.[DOI]

[3] Zhang H, Wang L, Pei J et al. RF-sign: Position-independent sign language recognition using passive RFID tags. *IEEE Internet Things*, 2023; 11: 9056-9071.[DOI]

[4] LeCun Y, Bottou L, Bengio Y et al. Gradient-based learning applied to document recognition. *P IEEE*, 1998; 86: 2278-2324.[DOI]

[5] Linck I, Gómez AT, Alaghband G. SVG-CNN: A shallow CNN based on VGGNet applied to intra prediction partition block in HEVC. *Multimed Tools Appl*, 2024; 83: 73983-74001.[DOI]

[6] Zhang J. Research on Speech Information to Sign Language Translation Based on 1D-GoogLeNet and LSTM. Proceedings of the 2024 9th International Conference on Cyber Security and Information Engineering, Kuala Lumpur, Malaysia. 2024: 707-712.[DOI]

[7] Gong S. Traffic Image Representation-Based ResNet-BiGRU Intrusion Detection.Proceedings of the 7th International Conference on Information Technologies and Electrical Engineering, Guangzhou, China. 2024: 325-330.[DOI]

[8] Tan YS, Lim KM, Lee CP. Hand gesture recognition via enhanced densely connected convolutional neural network. *Expert Syst Appl*, 2021; 175: 114797.[DOI]

[9] Mujahid A, Awan MJ, Yasin A et al. Real-time hand gesture recognition based on deep learning YOLOv3 model. *Appl Sci*, 2021; 11: 4164.[DOI]

[10] Chen Q, Tao Q, Zhang X et al. Gesture accuracy recognition based on grayscale image of surface electromyogram signal and multi-view convolutional neural network. *Journal of Biomedical Engineering*, 2024; 41: 1153-1160.[DOI]

[11] Hong D, Han Z, Yao J et al. SpectralFormer: Rethinking hyperspectral image classification with transformers. *IEEE T Geosci Remote*, 2021; 60: 1-15.[DOI]

[12] Vaswani A, Shazeer N, Parmar N et al. Attention is all you need. Advances in neural information processing systems, 2017; 30: 1-15.[DOI]

[13] Huang X, Ma G. Cross-Modality Object Detection Based on DETR. *IEEE Access*, 2025; 13: 51220-51230.[DOI]

[14] Liu Z, Lin Y, Cao Y et al. Swin transformer: Hierarchical vision transformer using shifted windows. *P IEEE*/CVF international conference on computer vision. 2021: 10012-10022.[DOI]

[15] Garg M, Ghosh D, Pradhan P M. Gestformer: Multiscale wavelet pooling transformer network for dynamic hand gesture recognition. *P IEEE*/CVF Conference on Computer Vision and Pattern Recognition. 2024: 2473-2483.[DOI]

[16] Besrour S, Surapaneni Y, Mubibya GS et al. A Transformer-Based Approach for Better Hand Gesture Recognition. 2024 International Wireless Communications and Mobile Computing (IWCMC). *IEEE*, 2024: 1135-1140.[DOI]

[17] Liu X, Hu L, Tie L et al. Integration of Convolutional Neural Network and Vision Transformer for gesture recognition using sEMG. *Biomed Signal Proces*, 2024; 98: 106686.[DOI]

[18] Haq MA, Ridlwan M, Naila I et al. Leveraging Self-Attention Mechanism for Deep Learning in Hand-Gesture Recognition System. E3S Web of Conferences, 2024; 500: 01009.[DOI]