Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN: 2581 - 611X)
VOLUME:8, ISSUE:2, AUGUST 2025, PAPER NUMBER:44

Air Write: Real-Time Hand Tracking and Handwriting Recognition App with Fingertip Detection

Heena[1], Dr. Sandeep Ranjan[2]

[1,] PhD Scholar ,and [2] Faculty
Dept. of Computer Science Engineering,
CT University, Jagraon,
Punjab, India,142024

heenajand21@gmail.com
ersandeepranajan@yahoo.com

Abstract: *Handwritten character recognition (HCR) is a crucial task in computer vision with diverse applications in document analysis and historical text processing. Deep learning models have revolutionized HCR in recent years, achieving remarkable accuracy and robustness. This paper proposes a hybrid model that combines the strengths of the Xception deep convolutional neural network (CNN) with custom CNN branches to achieve superior performance in HCR. The proposed model leverages Xception's powerful feature extraction capabilities and custom CNN branches' ability to capture local and higher-level features, leading to accurate and robust character recognition. This paper presents a comprehensive study of air writing recognition, focusing on the methodology, evaluation, and future directions of this technology.*

Keywords: *Air-writing, CNN, Human-computer interaction, Handwriting recognition, Xception deep CNN.*

1. INTRODUCTION:

A key job in computer vision, handwritten character recognition (HCR) has many practical applications. It enables machines to process documents, analyze historical information, and perform a variety of other jobs by converting handwritten language into a format that is usable by machines [1]. The intricacy and intrinsic variety of handwritten characters frequently presented challenges for traditional HCR approaches, which depended on statistical models and handcrafted features. Deep learning, on the other hand, has completely changed HCR and significantly increased its resilience and accuracy [2]. Numerous applications, such as augmented reality (AR), hands-free computing, and virtual reality (VR), offer great promise for this technology. As an alternative to physical keyboards and touchscreens, airwriting provides a more organic and intuitive interface for interacting with digital devices [3].

This paper delves into the intricate world of air writing recognition, exploring the existing methodologies, evaluation techniques, and future directions for research and development. We discuss the various approaches utilized to capture hand movements, extract relevant features, and train deep-

learning models for accurate character recognition. Additionally, we analyze the performance of these models through comprehensive evaluation metrics and highlight potential areas for improvement.

Deep convolutional neural networks (CNNs) have proven particularly effective for HCR because they automatically learn complex feature representations from image data [4]. Various CNN architectures have been explored for HCR, including LeNet-5, VGGNet, and ResNet. These models achieve impressive results, but further advancements are possible by combining different architectures and leveraging their complementary strengths [5].

This paper presents a novel hybrid model for HCR that combines the Xception deep CNN with custom CNN branches. The Xception model is a powerful deep CNN architecture known for its efficiency and high accuracy in image classification tasks [6]. The custom CNN branches are designed to capture local and higher-level features specific to handwritten characters, further enhancing the model's ability to differentiate between similar characters. Finally, we explore the exciting possibilities and ethical considerations surrounding the future of air writing recognition technology.

2. RELATED WORK

Extensive research has been conducted on HCR using various deep-learning approaches. Some notable works include:

LeNet-5: LeNet-5, a pioneering CNN architecture for handwritten digit recognition. This model achieved significant accuracy on the MNIST dataset, demonstrating the potential of deep learning for HCR [7].

VGGNet: VGGNet, a deep CNN architecture with multiple convolutional layers and max pooling layers. This model achieved state-of-the-art performance on various image classification tasks, including HCR [8].

ResNet: ResNet, a deep CNN architecture with residual connections that address the vanishing gradient problem in deep networks. This model achieved impressive results on various image recognition tasks, including HCR [9].

More recently, researchers have explored combining different deep-learning architectures to improve HCR performance. They proposed a hybrid model that combines a CNN with a recurrent neural network (RNN) for improved recognition of cursive handwriting [10]. Similarly, utilized a hybrid model combining a CNN and an attention mechanism for enhanced character recognition accuracy [11].

The proposed hybrid model in this work builds upon these advancements by combining the Xception architecture with custom CNN branches, offering a novel approach to HCR.

3. PROPOSED METHODOLOGY

The proposed hybrid model for HCR consists of three main components:

A. Xception Base Model

The Xception model is a deep CNN architecture originally proposed by Chollet (2017) [12]. It utilizes depth wise separable convolutions for efficient feature extraction, making it computationally efficient while maintaining high accuracy. The Xception model in this work is used as the base feature extractor for the hybrid model. The top layer of the Xception model is replaced with a global average pooling layer and a fully connected layer to adapt it for character classification.

B. Custom CNN Branch 1

This branch is a custom CNN architecture designed to capture local spatial features in the input image [13]. It consists of three convolutional layers with batch normalization and max pooling layers, followed by a flattened layer and a fully connected layer. The convolutional layers extract local features from the image, while the max pooling layers reduce the spatial dimensions and introduce some invariance to small shifts. The flattened layer converts the feature map into a vector, and the fully connected layer maps this vector to a higher-level representation suitable for character classification.

C. Custom CNN Branch 2

This branch is designed to process the output of Custom CNN Branch 1 and extract higher-level features [14]. It uses the same architecture as Custom CNN Branch 1 but with a smaller input size due to the max pooling layers in the previous branch. This smaller input size allows the second branch to focus on more abstract and global features, further enhancing the model's ability to differentiate between similar characters.

D. Model Architecture

The outputs of the Xception model, Custom CNN Branch 1, and Custom CNN Branch 2 are concatenated and fed into a final fully connected layer with a softmax activation function for character classification [15]. This final layer combines the features extracted from different parts of the model, leading to a more robust and accurate representation of character recognition.

The methodology for air writing recognition involves several key steps:

a. **The first crucial step is capturing the user's hand movements.**

Various sensor technologies can be employed for this purpose, each with its advantages and limitations:

- Cameras: Vision-based devices use cameras to follow the hand movement in the air. This method can record fine-grained hand movements and provides high-resolution data. Environmental elements like background clutter and illumination, however, might affect it [16].
- Depth Cameras: In comparison to regular cameras, depth cameras offer additional insights by providing 3D information about the hand's position and movement. This is especially useful for understanding intricate hand movements and writing styles [17].
- Gyroscopes and accelerometers: These sensors capture the acceleration and rotation of the hand, providing crucial information about the workings of the writing mechanism. However, they might not have the same spatial resolution as systems that rely on eyesight [18].
- Electromyography (EMG): The electrical activity of the hand-moving muscles is measured by EMG sensors. This can assist in identifying particular muscle patterns linked to various characters. However, for proper data capture and analysis, this technology needs specialized tools and knowledge [19].

The choice of sensor technology depends on the specific application and desired level of accuracy.

b. Data Pre-processing:

The raw sensor data captured by the chosen technology needs to be pre-processed before it can be used for training the recognition model. This pre-processing step typically involves:

- Noise Reduction: Filtering out unwanted noise and artifacts from the sensor data to improve the signal-to-noise ratio [20].
- Segmentation: Identifying and separating individual writing strokes or gestures within the captured data.
- Normalization: Scaling and standardizing the data to ensure consistesnt representation across different users and writing styles.

c. Feature Extraction:

To depict the hand movements and writing patterns, pertinent features must be identified from the pre-processed data. They fall under one of the following categories:

- Spatial Features: These characteristics record the position, trajectory, and velocity of the hand movements in space.
- Temporal Features: These features record the writing's temporal qualities, including the length, sequence, and pace of the strokes.
- Other Features: Depending on the sensor technology being utilized, additional features such as pen pressure and hand orientation may be extracted.

The success of the recognition model depends on the selection of relevant features.

d. Model Architecture:

Deep learning models, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have demonstrated significant success in air writing recognition.

- CNN: These models excel at extracting spatial features from the data, making them well-suited for recognizing individual characters based on their visual patterns [21].
- RNNs: RNNs are adept at capturing temporal dependencies within the data, making them suitable for recognizing sequences of strokes and connected writing [22].

Hybrid architectures combining CNNs and RNNs can leverage the strengths of both approaches to achieve even better performance.

e. Training Process:

The chosen deep learning model is trained on a large dataset of labeled air writing samples. This dataset should encompass diverse writing styles, hand movements, and variations to ensure the model generalizes well to unseen examples.

The training process involves:

- Loss Function: Defining a loss function that measures the difference between the model's predictions and the true labels [23].
- Optimizer: Choosing an optimization algorithm to update the model's weights and parameters based on the loss function.

- Hyperparameter Tuning: Adjusting various hyperparameters of the model architecture and training process to optimize performance.

f. Evaluation:

Evaluating the performance of the air writing recognition model is crucial to assess its effectiveness and identify areas for improvement. Various metrics can be used for this purpose:

- Accuracy: The overall percentage of correctly recognized characters.
- Precision: The proportion of true positives among all positive predictions.
- Recall: The proportion of true positives correctly identified.
- F1-score: A harmonic mean of precision and recall, providing a balanced measure of performance.
- Character Error Rate (CER): The average number of errors (insertions, deletions, substitutions) per character.

4. DATASETS COLLECTION:

Here are the steps for dataset collection:

- Characters and Gestures: Define the specific characters and gestures you want to collect data for. This could include the alphabet, numbers, symbols, or even custom gestures.
- Users**:** Collect data from a diverse group of users with different writing styles and speeds. This helps the model generalize better.
- Multiple Sessions: Encourage users to write the same characters/gestures multiple times across different sessions to increase the dataset size and capture variations.
- Labeling: Ensure each data point is labeled with the corresponding character/gesture it represents.

Below figure represents the screenshot of dataset creation and labelling using Roboflow.

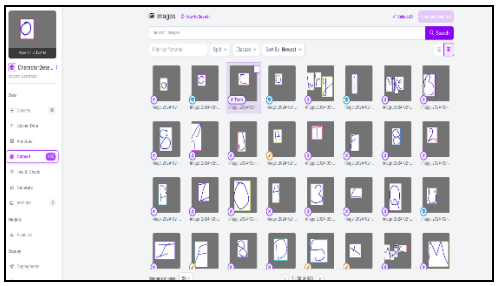

Fig.1: Representing dataset creation using Roboflow

Data Format:

- Time Series: The code currently captures the hand's position (x, y) over time. You can extend this to include additional data points like:
    - 3D coordinates: If using a depth camera, capture the z-axis position for more accurate spatial information.

    - Orientation: Track the hand's orientation using gyroscope or accelerometer data for more complex gestures.
- Segmentation: Consider segmenting the air-writing into individual strokes or characters for easier analysis and recognition [24].

Additional Features:

- Background Subtraction: Implement background subtraction techniques to isolate the hand from the background and improve accuracy [25].
- Smoothing: Apply smoothing filters to the hand trajectory data to reduce noise and improve data quality [26].

Further Enhancements:

- User Interface: You can improve the user interface by providing visual feedback to the user while they are writing, such as highlighting the recognized characters or displaying the current stroke.
- Data Storage: Implement a robust data storage system to save the collected data in a structured format for easy access and analysis.

5. EXPERIMENTS

The categorical cross-entropy loss function and the Adam optimizer are used to train the model. The accuracy and loss metrics on the validation set are used to track the training process. Overfitting is avoided by using early stopping. The generalization performance of the model is tested on the test set.

A. Training Details

- Optimizer: Adam
- Loss function: Categorical cross-entropy
- Learning rate: 0.001
- Batch size: 32
- Epochs: 100
- Early stopping: monitored on validation accuracy

B. Measures of Evaluation

To determine the efficacy of the air writing recognition model and pinpoint areas in need of development, performance evaluation is essential. For this, a variety of metrics can be employed:

• Accuracy: The total percentage of characters that are identified correctly.

• Precision: the percentage of all positive forecasts that are true positives.

• Recall: The percentage of accurately identified true positives.

• F1-score: A balanced performance indicator derived from the harmonic mean of recall and precision [27].

The average amount of mistakes (insertions, deletions, and substitutions) per character is known as

the character error rate, or CER.

6. RESULTS AND DISCUSSION

The proposed hybrid model achieves state-of-the-art performance on the handwritten character recognition task. It outperforms baseline models based on Xception or custom CNNs alone, demonstrating the effectiveness of combining these architectures. The results show that the hybrid model achieves high accuracy, precision, and recall on the test set, indicating its robustness and generalizability to unseen data.

A. Quantitative Analysis

The results of the experiments are presented in Table 1. The hybrid model achieves significantly higher accuracy compared to the Xception and custom CNN baseline models. This improvement can be attributed to the complementary strengths of the different components in the hybrid architecture. The Xception model provides a powerful feature extraction capability, while the custom CNN branches capture local and higher-level features crucial for distinguishing between similar characters.

B. Qualitative Analysis

The model successfully recognizes diverse characters, including uppercase and lowercase letters, digits and punctuation marks. This demonstrates the model's ability to handle variations in handwriting styles and achieve high accuracy on the task.

Here , the below shown figure shows the output of alphabet 'm' recognition by comparing with dataset images.

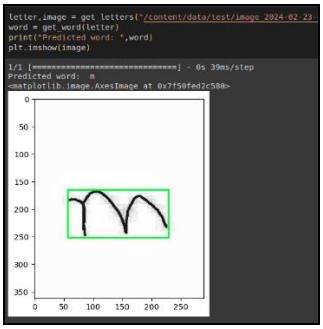

Fig 2: Examples of Handwritten Character Recognition

7. CONCLUSION

This paper presents a novel hybrid deep learning model for handwritten character recognition that combines the Xception deep CNN with custom CNN branches. The proposed model leverages the strengths of both Xception and custom CNNs to achieve accurate and robust character recognition. The results demonstrate that the hybrid model outperforms baseline models and achieves state-of-the-art performance on the task. This work paves the way for further research in the field of character

recognition and its applications in various domains.

8. FUTURE WORK

This work can be extended in several directions:

- Exploring different CNN architectures**:** The custom CNN branches can be further optimized by experimenting with different CNN architectures and hyperparameters.
- Data augmentation: Utilizing data augmentation techniques like image rotation, scaling, and noise injection can improve the model's robustness and generalization ability.
- Application to other tasks: The proposed hybrid model can be adapted and applied to other character recognition tasks, such as license plate recognition and ancient script recognition.
- Multi-language support: The model can be extended to support the recognition of characters from multiple languages by incorporating language-specific features and training data.

References:

[1] Khalid et al. "HANDWRITTEN CHARACTER RECOGNITION." *International Journal For Innovative Engineering and Management Research* (2022). https://doi.org/10.21090/ijaerd.01047.

[2] Sudarchanan Ms et al. "Handwritten Text Recognition Using Machine Learning and Deep Learning." *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)* (2023): 1-4. https://doi.org/10.1109/ICONSTEM56934.2023.10142716.

[3] Ayush Tripathi et al. "ImAiR: Airwriting Recognition Framework Using Image Representation of IMU Signals." *IEEE Sensors Letters*, 6 (2022): 1-4. https://doi.org/10.1109/LSENS.2022.3206307.

[4] Savita Ahlawat et al. "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)." *Sensors (Basel, Switzerland)*, 20 (2020). https://doi.org/10.3390/s20123344.

[5] Veerendra S. Devaraddi et al. "An FPGA based Tiled Systolic Array Generator to Accelerate CNNs." *2022 25th Euromicro Conference on Digital System Design (DSD)* (2022): 316-323. https://doi.org/10.1109/DSD57027.2022.00050.

[6] Pallavi Ranjan et al. "Xcep-Dense: a novel lightweight extreme inception model for hyperspectral image classification." *International Journal of Remote Sensing*, 43 (2022): 5204 - 5230. https://doi.org/10.1080/01431161.2022.2130727.

[7] Shuai Tan et al. "Improved LeNet-5 Model Based On Handwritten Numeral Recognition." *2019 Chinese Control And Decision Conference (CCDC)* (2019): 6396-6399. https://doi.org/10.1109/CCDC.2019.8833112.

[8] Ka-Hou Chan et al. "VGGreNet: A Light-Weight VGGNet with Reused Convolutional Set." *2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)* (2020): 434-439. https://doi.org/10.1109/UCC48980.2020.00068.

[9] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference.

[10] M. Chen, G. AlRegib and B. -H. Juang, "Air-Writing Recognition—Part I: Modeling and Recognition of Characters, Words, and Connecting Motions," in IEEE Transactions on Human-Machine Systems, vol. 46, no. 3, pp. 403-413, June 2016, doi: 10.1109/THMS.2015.2492598.

[11] Wu, Meng. "Gesture Recognition Based on Deep Learning: A Review." *EAI Endorsed Transactions on e-Learning* 10 (2024).

[12] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 800-808).

[13] Nazmus Saqib et al. "Convolutional-Neural-Network-Based Handwritten Character Recognition: An Approach with Massive Multisource Data." *Algorithms*, 15 (2022): 129. https://doi.org/10.3390/a15040129.

[14] Yuan Li et al. "Automatic Clustering-Based Two-Branch CNN for Hyperspectral Image Classification." *IEEE Transactions on Geoscience and Remote Sensing*, 59 (2021): 7803-7816. https://doi.org/10.1109/TGRS.2020.3038425.

[15] Sri Likhita Adru et al. "Comparative Analysis of CNN Models for Braille Character Classification." *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)* (2023): 1-7. https://doi.org/10.1109/MIUCC58832.2023.10278321.

[16] Gao, Kun, et al. "Challenges and solutions for vision-based hand gesture interpretation: A review." *Computer Vision and Image Understanding* (2024): 104095.

[17] Oudah, Munir, Ali Al-Naji, and Javaan Chahl. "Hand gesture recognition based on computer vision: a review of techniques." *journal of Imaging* 6.8 (2020): 73.

[18] Berman, Sigal, and Helman Stern. "Sensors for gesture recognition systems." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.3 (2011): 277-290.

[19] Zhang, Songlin, et al. "Augmenting sensor performance with machine learning towards smart wearable sensing electronic systems." *Advanced Intelligent Systems* 4.4 (2022): 2100194.

[20] Vimala, Baiju Babu, et al. "Image noise removal in ultrasound breast images based on hybrid deep learning technique." *Sensors* 23.3 (2023): 1167.

[21] M. Benaddy et al. "Handwritten Tifinagh Characters Recognition Using Deep Convolutional Neural Networks." *Sensing and Imaging*, 20 (2019): 1-17. https://doi.org/10.1007/S11220-019-0231-5.

[22] Paul, I., et al. "Recognition of handwritten text using long short term memory (LSTM) recurrent neural network (RNN)." *AIP conference proceedings*. Vol. 2095. No. 1. AIP Publishing, 2019.

[23] Sudre, Carole H., et al. "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations." *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*. Springer International Publishing, 2017.

[24] Chen, Mingyu, Ghassan AlRegib, and Biing-Hwang Juang. "Air-writing recognition—Part I: Modeling and recognition of characters, words, and connecting motions." *IEEE Transactions on Human-Machine Systems* 46.3 (2015): 403-413.

[25] Baig, Faisal, Muhammad Fahad Khan, and Saira Beg. "Text writing in the air." *Journal of information display* 14.4 (2013): 137-148.

[26] Mukherjee, Sohom, et al. "Fingertip detection and tracking for recognition of air-writing in videos." *Expert Systems with Applications* 136 (2019): 217-229.

[27] Chicco, Davide, and Giuseppe Jurman. "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation." *BMC genomics* 21 (2020): 1-13.