ITM Web of Conferences **82**, 01004 (2026) https://doi.org/10.1051/itmconf/20268201004
ICNEXTS'25

Dynamic Letter Recognition Utilizing Motion Analysis and Voice Based Error Feedback

Sudarsan P[1*], *Sanjay* Kumar M[2], *Vibin* S[3] and V.*Sathiesh* Kumar[4]

[1]*Student, Department of Electronics Engineering*, *Madras Institute of Technology Campus, Anna University* Chennai, Tamil Nadu, India.
[2]*Student, Department of Electronics Engineering*, *Madras Institute of Technology Campus, Anna University* Chennai, Tamil Nadu, India.
[3]*Student, Department of Electronics Engineering*, *Madras Institute of Technology Campus, Anna University* Chennai, Tamil Nadu, India.
[4]*Assistant Professor(Sl. Gr.), Department of Electronics Engineering*, *Madras Institute of Technology Campus, Anna University* Chennai, Tamil Nadu, India.
*Corresponding author: sudarshanprabha2004@gmail.com

Abstract. This paper presents a dynamic letter recognition system that utilizes motion based analysis and real time auditory feedback to enhance handwriting recognition accuracy, accessibility, and learner engagement. The suggested framework uses a combination of a hybrid Bidirectional Long Short Term Memory (BiLSTM) and Multi Layer Perceptron (MLP) model to find alphanumeric characters (A-Z, 0-9) from motion based stroke patterns that the Intel RealSense D405 depth camera picks up. The handwritten motion dataset is in JSON format and has both temporal stroke sequences and geometric data. It was gathered with a custom Streamlit interface and marked up with the Computer Vision Annotation Tool (CVAT). Since the method examines both the spatial and temporal aspects of handwriting dynamics instead of fixed image, it has been successfully applied across various writing forms.The TTS based modules providing real time audio feedback to students during interactive and corrective processes increase the students ability to correct their identification mistakes rapidly through real time reminders. Experimental results demonstrated a 96 percent classification rate, demonstrating that this system is reliable and capable of adapting to a wide variety of applications, which are suitable as an affordable and flexible solution for assistive education and communication for individuals with disabilities.

1 Introduction

Handwriting recognition supports educational and assistive technology allowing more natural interaction with computers. Standard handwriting recognition systems usually look at static images of written text. These perform well in controlled conditions, but often fail when presented with irregular, incomplete, or fluctuating handwriting styles.These challenges are especially important for users with fine motor disabilities for whom irregularities in stroke motion and pressure change recognition accuracy. The motion based dynamic letter recognition system captures the motion of a user's hand as they write and analyzes the geometric properties and temporal relationships of each individual stroke of the letter. We utilized an Intel RealSense D405 camera to obtain video data of the user as they wrote. We extracted information about the motion of the user's hand from the video data by processing the video data using a combination of a Bidirectional Long Short Term Memory (BiLSTM) and Multi Layer Perceptron (MLP). The MLP was trained to learn the normalized geometric characteristics of each letter and the BiLSTM was trained to predict the successive changes in stroke flow of each letter. By utilizing the output of the MLP and BiLSTM, we were able to create a unified representation of each letter that was robust enough to correctly classify letters regardless of whether the user's handwriting was uneven, partially formed, or had varying strokes between users.

One of the primary features of the system is the voice based feedback mechanism. This system consists of a TTS (text to speech) module that provides audio instructions in real-time. Users get feedback that helps them detect, fix and learn the error when writing. This keeps them involved better and helps in developing the neuro motor skills. The model, which was trained and obtained 96% accuracy in recognition, is implemented on the Jetson nano board which has TTS for real-time error correction and feedback creation. The research presented bridges the gap between assistive learning and AI based handwriting recognition that offers a smart, flexible, and easy solution for educational and rehabilitation purposes. Combining motion analysis, deep learning, and instantaneous aural feedback demonstrated the effectiveness of spatiotemporal modeling in improving handwriting recognition capabilities and making learning more inclusive.

2 Literature Survey

[1] A writing assessment tool was developed by William Villegas Ch et al. (2024) using real-time writing feedback via a standard RGB camera and Convolutional Neural Network (CNN) where children's performance could be assessed in real time for writing skill development. The CNN was effective at feature extraction from captured images of handwriting but was ineffective at capturing the dynamics of stroke motion and the inference of temporal relationships, the latter being required for writing process assessments.

[2] A comparative study of deep learning methods for handwritten digit recognition by J. Deepika et al. (2024) used CNN and RNN. CNN effectiveness at feature extraction and classification was noted. However, the performance degradation for different handwriting styles and free writing environments suggests a lack of generalisation.

[3] An Extreme Learning Machine (ELM) was used for handwritten alphabet recognition by Junlei Song et al. (2018) which showed that an ELM for static character image classification produced high accuracy; however, there was no ability to assess stroke flow from one character to the next and thus the motion and dynamics of handwriting were not appreciated.

[4] A trajectory based air writing recognition device by Fang Jin et al. (2020) was established through an LSTM network with depth sensor data which reports 99.32% accuracy on the Real Time Digit (RTD) dataset suggesting that where motion is observed a greater level of accuracy exists as compared to static assessments of handwriting.

[5] An AI based handwriting evaluation system by Yoshida R. et al. (2022) assessed pen pressure movements of children's finger movements in developmental disabilities through an optical sensor relative to grip quality and stroke flow. This was successful but reliant on special hardware and not sensitive enough with respect to adaptability for various motor patterns.

[6] An intelligent rehabilitation glove by Huang T. et al. (2021) relied on IMUs to detect upper limb exercise movements with Cerebral Palsy (CP) patients with data sent to a mobile application for feedback. However, practical implementation where cost effectiveness and a limitation for which exercises could be suggested reduced the effectiveness of this approach for writing on a daily basis.

[7] Mouelhi A. et al. (2023) designed a computer vision and deep learning based AI assisted writing rehabilitation framework that analyzed writing trajectories and pressure for providing progress reports to therapists but without an interactive and real time correction module for user feedback.

[8] Kumar S. and Reddy V. (2021) developed a webcam based handwriting motion capture system using MediaPipe hand tracking API. While low cost, the system was prone to failure under a poor lighting environment and required manual data labeling for training.

[9] An LSTM based model for writing gestures in the air using leap motion controller by Chen X. et al. (2022) however is similar to other models mentioned before because it's an external sensor and not a practical reality for daily or classroom use.

[10] Lastly, this was tested using voice feedback gesture recognition integrated rehabilitation system gesture feedback for children with movement disabilities that increased their motivation response by Gupta R. et al. (2023) However, this feedback is neither scalable nor adjustable enough to be worthwhile for handwriting correction criteria for intervention.

Finally, based on literature, there's a clear shift from handwriting and motion recognition models based on pictures and a more static moment to one that embraces moment and flow over time and space. For example, the studies prior to movement and flow over time and space used CNN models (Villegas Ch et al., 2024; Deepika et al., 2024) to determine applicable visual features and extensive new data for classification levels, determining visual precision index results determining classification levels. This model worked well but failed for attention to dynamics of writing and time variations for each letter. Preliminary results of Song et al. (2018); Jin et al. (2020) show strong recognition accuracy for different CNN and ELM frameworks tested and although the researchers found success, these findings were limited by a lack of focused attention to the writing order and flow. Subsequently, research shifted to LSTM based models to determine motion modeling (Fang Jin et al., 2020; Chen X. et al., 2022) and based on hand gestures and in air writing, but similar to the first tested models, these required depth and motion sensors, not easily accessible for practical use in real world settings. Finally, explorations by Kumar S. & Reddy V. (2021); Gupta R. et al. (2023) use gesture recognition through MediaPipe for affordable voice based actions and gesture recognition with voice feedback to gesture recognition however these model still suffer from lighting sensitivity, manual annotation, lack of complication with other types of intervention and adjustability and other similar issues. Thus, there is a clear research gap for Real time interactive handwriting recognition that an Intel RealSense D405 will present through an LSTM MLP framework.

3 Proposed Methodology

The proposed dynamic handwritten character recognition system combines temporal stroke order prediction and geometric feature encoding. The entire system operates as an end to end model with four submodules - Data Collection, Data Labeling, Feature Encoding and Data Augmentation - and hybrid LSTM

MLP network training for model training to achieve accurate recognition of A-Z and 0-9 from recorded strokes of dynamic handwritten motion.

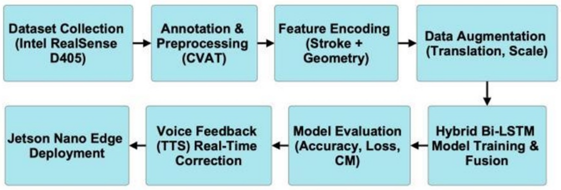

Fig. 1. Block diagram of the proposed handwriting recognition approach from data collection to preprocessing to feature vectorization and enrichment, training and validation of the BiLSTM and implementation on Jetson Nano.

3.1 Data Collection

Data collection was performed through a custom interactive data collection application in Streamlit, connected to the Intel RealSense D405 depth camera. Thus, users were able to compile data from the dynamic writing motion of each letter in the alphabet, and similarly, numbers. The RealSense D405 connected through multiple streams, allowing simultaneous RGB and depth stream capture which facilitated proper motion detection of handwriting to follow along with the proper motion direction, position, and speed.

Each data capture was compiled with metadata time stamps denoting user identification, handedness (left or right), light type (natural, artificial, low light), and camera positioning (top, front, side). Data was compiled in .AVI format for video recordings where applicable, and metadata was compiled in CSV. All data was segmented for standardization and dataset replicability.

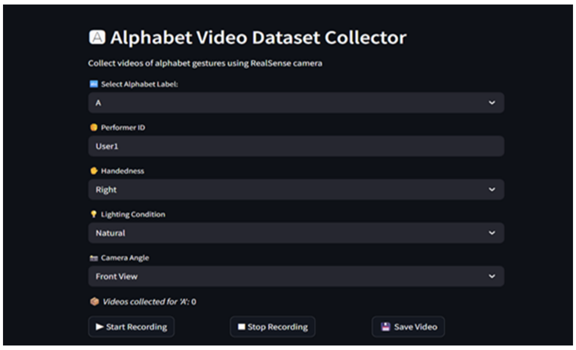

Fig. 2. Data collector interface for gesture videos of letters and numbers recorded through the Intel RealSense camera.

3.2 Annotation and Data Conversion

The videos were annotated via the Computer Vision Annotation Tool (CVAT). Stroke level annotations and motion event annotations per frame for each video included start_frame, end frame, pen down, pen lift, and keypoints. Annotations also included vertical stroke, horizontal stroke, diagonal stroke, curve, oval and circle for direction of strokes and curvature.

In addition, inclination (upward/ downward/ leftward/ rightward), intensity (small/large), and rotation (clockwise/anticlockwise) were assessed for every stroke. The annotations were exported to XML and converted to JSON for model preprocessing. Each JSON entry contained:

Stroke Order: in temporal order with appropriately marked frames. ·

Quantifiable Attributes: height of stroke, ratio of curvature, and placement in space relative to each other.

This structured representation captured both the time and space information, which would be a basis for the recognition module based on LSTMs.

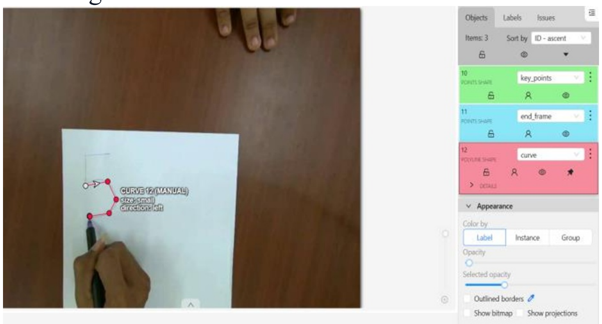

Fig. 3. Overlay of Handwritten Digit "5" on Acquisition with directionality of the points represented by detected keypoints and curvature.

3.3 Feature Encoding

This part of the process digitized this hand labeled data to tokenizing when applicable to deep learning methods. The tokenization of the stroke class and relevant features was conducted using a common lexicon and integerized to standardized arrays of 40 tokens.

In addition, six geometric features were determined relative to each character class sample: Vertical stem height, Primary horizontal stroke position, Secondary horizontal stroke stroke position, Curve extreme point ratio, Curve lowest point ratio, Curve end point ratio. Min max scaling was applied to digitize these features on a [0,1] scale to accommodate reduced value discrepancies. The input parameters established for usability were:

Sequential Character Field ($N \times 40$): tokenized sequential strokes over time.

Geometric Character Field ($N \times 6$): normalized geometric features

This level of distinction is needed since characters that are visually similar (like F/I and L/T) are characterized based on geometrics where recognizable strokes are minor but where acquired geometric differentials provide significant benefit to recognition.

3.4 Data Augmentation

All subsequent stroke level data augmentation methods were then applied to the dataset to facilitate model generalization and prevent overfitting. These natural variations of handwriting were applied to the data set while preserving the integrity of the labels of the characters. These include:

Translation and Scaling - replicates the distance a pen would lift and move across a page and the size in which handwriting occurs.

Rotation (±5°) - prevents character error by giving an angular degree to the mean of slanted writing.
Coordinate Jittering - noise applied to the coordinates to represent hand shaking.
Nonlinear Warping - sinusoidal adjustments to represent warped, nonlinear movements.
Stroke Dropout - randomly dropping small strokes in a manner consistent with a partially finished writing style.

The augmented dataset was 1.5 times larger than the original as each JSON entry had the potential to sound like it compounded many more instances. This is the Augmentation Pipeline which makes for better and transferable research models on works that are familiar, not on new, untested handwriting.

3.5 Deep Learning Architecture

The BiLSTM and MLP proposed system can understand and learn the time varying development of writing and the spatial orientation and geometrical form of the characters, enabling reliable recognition and classification of alphanumeric characters (A-Z, 0-9) as intended even if the writing is faded, dirty or not performed in a straight manner.

3.5.1 Bi-LSTM Branch

The BiLSTM component is responsible for learning temporal relations for each sequence. Each character is represented by a series of encoded stroke tokens corresponding to events related to the character's writing, which include the writing activities of pendown, penlift, and the stroke directions. Initially, We pass these tokens through an embedding layer that turns the tokens which are represented by the discrete indices, to a continuous dense vector, while retaining the semantic connections to the different types of strokes which are represented by the tokens.

Afterward, two stacked Bidirectional LSTM layers, each with 64 hidden units, process the embedded sequences. Unlike standard LSTMs that only process sequences in one direction, the BiLSTM model interprets the interactions between earlier and later strokes, crediting the model with the ability to comprehend how strokes interact in the construction of a letter. This bidirectional processing allows the network to understand the successive strokes and learn the temporal flow of how a vertical stem is followed by a horizontal or curved stroke. To prevent overfitting and ensure stable convergence, we implement dropout regularization (rate = 0.3) and batch normalization after each of the BiLSTM layers.

3.5.2 MLP Branch

The BiLSTM branch focuses on sequential relationships, instead the MLP branch out with spatial and geometric relationships from numerical descriptors. We represent each handwriting sample in a 6D geometric feature vector which includes elements like vertical stem height, horizontal stroke start position, secondary horizontal stroke placement, curve extreme point ratio, curve lowest point ratio, and curve end point ratio. Also we normalize these parameters from 0 to 1 which in turn removes user specific scaling issues and we achieve a uniformity between samples.

Dense layers are used to gradually understand non linear correlations between these geometric proportions after the normalized geometric vector has been passed through. The first dense layer consists of 64 neurons with ReLU activation, followed by a dropout layer with a rate of 0.1 to prevent overfitting. The second dense layer, which contains 128 neurons and uses ReLU activation, refines spatial representations and captures complicated geometric connections between stroke components. This branch gives the network the ability to differentiate between letters that have different structural shapes but identical motion sequences. For example, 'F' and 'E' share vertical stems but differ in horizontal positioning, whereas 'P' and 'R' differ by the existence of a diagonal leg. The MLP thus learns geometric cues that are not directly expressed in temporal data.

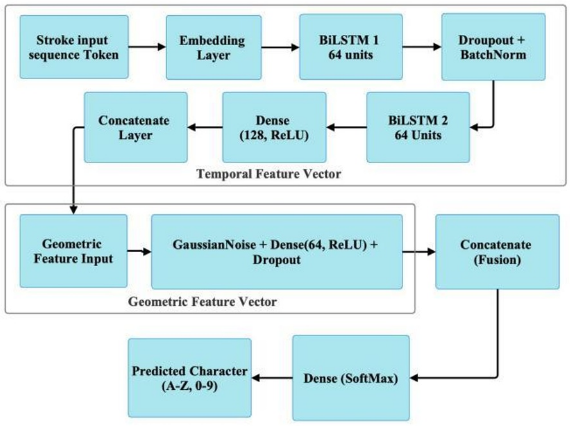

Fig 4. illustrates the architecture of hybrid deep learning systems based on temporal and geometric feature extraction for handwriting recognition.

3.5.3 Feature Fusion and Classification

The output feature vectors from the BiLSTM and MLP branches are concatenated to create a single spatiotemporal representation following independent learning in those branches. This fused feature vector combines temporal dependence obtained from stroke flow and spatial consistency recorded by geometric features.

The concatenated feature vector is then passed via fully connected layers with ReLU activation to refine higher level abstractions, followed by a softmax output layer with 36 neurons that correspond to the 26 alphabets and 10 digits.The network is trained using sparse categorical cross entropy loss and is optimized using an Adam optimizer which guarantees good convergence and classification accuracy, even within the boundaries of a small dataset.

3.6 Training Configuration

Adam optimizer with a learning rate of 0.001 and batch size of 4 are used. The loss function is a sparsified categorical cross entropy, which allows early stopping to prevent overfitting and capture the best generalization.

Eighty percent of the data is used as a training set and 20 % of the remaining data is kept for validation. This will enable you to during training. The performance of each training process is checked by accuracy, precision, confusion matrix analysis etc. These are then used to assess how each class performed and which mistakes were most frequent. To simulate many possible handwriting scenarios and to be able to handle different users, the training set is modified with controlled distortions, such as translations, rotations, and non linear warpings. The final trained model was capable of classifying the user's writing style accurately, and showed strong convergence at a 96 percent recognition accuracy level.

3.7 Voice Feedback Integration

Following each estimation, there is a comparison between the estimated character and the predicted label. If a match does not occur, the system will use a Text to Speech (TTS) Engine for real time audio feedback to guide the user on how to adjust stroke parameters such as "increase the top curve," or "verify the vertical line." It runs as a single unit on the Jetson Nano Board. The trained recognition model runs at the same time that the built in TTS module does. Since the Jetson Nano has an extremely low latency (less than 100ms/frame) and GPU accelerated inference to perform the audio prompts, this allows immediate responses based on stroke detection. This method could be used in assistive or adaptive learning environments, and is effective due to a continuous real time feedback loop providing engagement, as well as, reinforcing learning and motor skills training for users who need them.

4 Results and Discussion

4.1 Model Training and Evaluation

The Hybrid Bidirectional LSTM MLP architecture developed and trained under the specifications of the Kaggle GPU environment. Since the project had considerable computational requirements, it was the most appropriate configured to support them. As previously mentioned, 242 annotated samples of handwriting were collected, 182 alphabetical (A-Z) and 60 numerical (0-9). Each was saved as a JSON file with temporal and six geometric features.

Rotation, translation, scaling, nonlinear warping and stroke dropout as data augmentation techniques were applied to triple the data set due to limited data available. The model was trained for 50 epochs (batch size = 4) with the Adam optimizer (learning rate = 0.001) and the early stopping limit to prevent overfitting.

The data set was divided into 80% training and 20% validation subsets. The loss and accuracy plots converged well, very little difference between training and validation indicates acceptable generalization.

The resulting recognition on the final test corresponds to 96.33% which indicates that the training was successfully completed for the time and space features of handwriting.

Metric	Description	Observed Value
Training Accuracy	*Post convergence accuracy*	*99.8%*
Validation Accuracy	*Accuracy on unseen validation data*	*97.25%*
Test Accuracy	*Final accuracy on test dataset*	*96.33%*

Table above contains performance results from the proposed hybrid model which boasts high accuracy on training (99.8%), validation (97.25%) and test (96.33%) datasets.

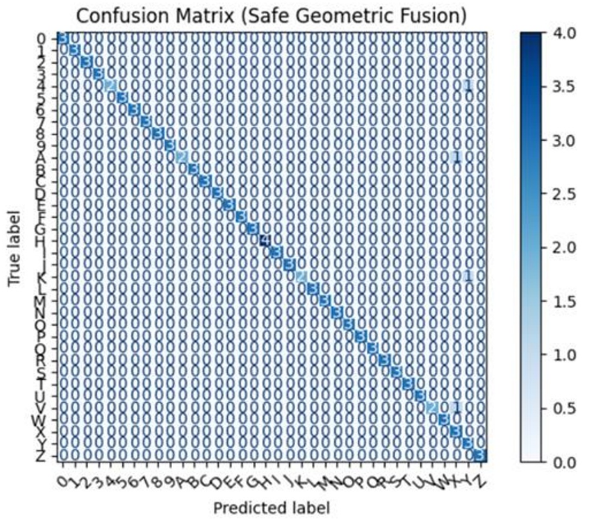

Fig. 5 presents the confusion matrix of the Safe Geometric Fusion model where since most samples lie on the diagonal this suggests extensive confidence in predictions and limited classification mistakes.

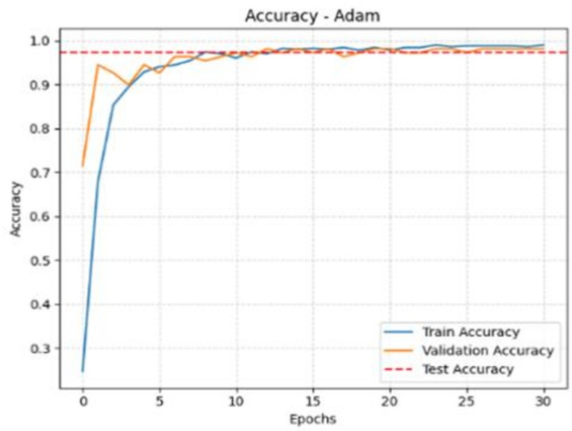

Fig. 6 presents the confidence of the model during training, validation and testing through epochs with Adam optimizer on stable convergence showing stable performance and minor overfitting.

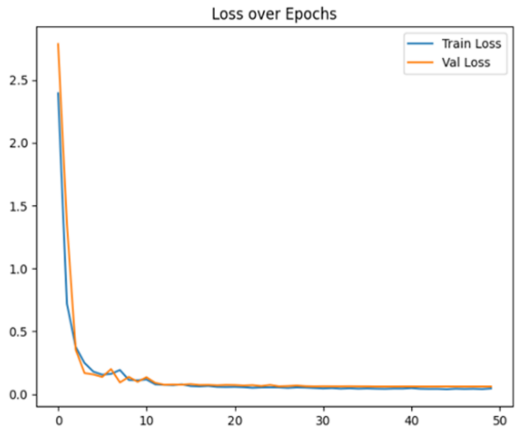

Fig. 7 presents the training and validation loss through epochs where quick convergence is clear and slight distance between the two curves suggests no overfitting was present.

4.2 Real-Time Deployment and Voice Feedback

The optimized model was deployed on the NVIDIA Jetson Nano for deployment where inference latency was reduced to approximately 0.35 seconds/frame after TensorRT conversion, achieving realtime requirements. TTS feedback was deployed via pyttsx3 and gITS for an immediate audio feedback loop which enhanced user motivation and focus, making the system suitable for interactive writing sessions and other neuro motor rehabilitation solutions for example, children with Cerebral Palsy.

4.3 Discussion

The application of the bidirectional LSTM layers with geometric feature fusion performed well on the dynamic handwriting recognition problem since it could capture both the temporal dependencies of strokes and the spatial properties of shapes, improving generalization from various writing styles.

This framework was surpassed by traditional CNN based models when tested on handwritten images that were static; however, this framework is best suited for a moving environment. With little latency, low powered hardware and a responsive feedback loop, this framework is an inexpensive, mobile option for assistive education and rehabilitation

4.4 Conclusion

The hybrid deep learning system proposed in this paper was extremely accurate (96.33% accuracy), made decisions in under one second, and provided continuous, real time feedback to the user. As such, the proposed system will provide accurate recognition and immediate feedback through the integration of motion dynamics and geometric structure analysis. The data collected demonstrate that the model has potential as an effective, artificial intelligence based rehabilitation tool to aid children with various forms of physical disabilities in developing their handwriting and fine motor skills capabilities.

Future work could include collecting a larger, more diverse set of data sets which would allow the generalizability of the model to be improved across multiple types of handwriting styles, age groups and levels of impairment.In addition, more sensors of movement or pressure (flex sensors, accelerometers, etc.) would enable better motor analysis and gait development since kinematic readings would be more nuanced. Also, the inclusion of reinforcement learning would help make the system more user friendly with personalized rehabilitation feedback relative to one's learning speed, difficulty, and success.

4.5 ACKNOWLEDGEMENT

This research work is funded by Directorate of Technical Education (DoTE), Government of Tamil Nadu under Chief Minister Research Grant (CMRG 2023 - 24), Project ID : CMRG2023AEC02008

REFERENCES

1. W. Villegas, J. García-Ortiz, and S. Sánchez-Viteri, Optimizing Writing Skills in Children Using a Real-Time Feedback System Based on Machine Learning. IEEE Access, (2025), doi: 10.1109/ACCESS.2024.3492974.
2. J. Deepika, Abirami Ravi, K. Chitra, T. Senthil, A study of deep learning techniques for handwritten digit recognition and classification, doi: 10.32629/jai.v7i5.1585.
3. J. Song, Q. Liu, S. Tian, Y. Wei, F. Jin, W. Mo, Research on Handwritten Alphabet Recognition System Based on Extreme Learning Machine, 2018 37th Chinese Control Conference (CCC), Wuhan, China, 4959–4964 (2018), doi: 10.23919/ChiCC.2018.8482669.
4. M.S. Alam, K.C. Kwon, M.A. Alam, M.Y. Abbass, S.M. Imtiaz, N. Kim, Trajectory-Based Air-Writing Recognition Using Deep Neural Network and Depth Sensor, Sensors, **20**(2), 376 (2020), doi: 10.3390/s20020376.
5. Xin Liu, Baotian Hu, Qingcai Chen, Xiangping Wu, Jinghan You, Stroke Sequence-Dependent Deep Convolutional Neural Network for Online Handwritten Chinese Character Recognition, doi: 10.1109/TNNLS.2019.2956965
6. C.L. Liu, F. Yin, D.H. Wang, Q.F. Wang, Online and offline handwritten Chinese character recognition: Benchmarking on new databases, Pattern Recognit., 46, 1, 155–162 (2013).
7. S.A. Husain, A. Sajjad, F. Anwar, Online urdu character recognition system, in Proc. IAPR Conf. Mach. Vis. Appl., 98–101 (2007).
8. Goswami, Mukesh, Suman Mitra, Structural Feature Based Classification of Printed Gujarati Characters. Pattern Recognition and Machine Intelligence. Springer Berlin Heidelberg, 82-87 (2013).
9. Multi-Language Online Handwriting Recognition Daniel Keysers, Thomas Deselaers, Henry A. Rowley, Li-Lun Wang, Victor Carbune Digital Object Identifier no. 10.1109/TPAMI.2016.2572693

10. A.H. Toselli, M. Pastor, E. Vidal, On-Line Handwriting Recognition System for Tamil Handwritten Characters, in Pattern Recognition and Image Analysis. Springer, 370–377 (2007).

.