International Journal of Research Publication and Reviews, Vol (6), Issue (6), June (2025), Page – 6114-6119

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Neural Decipher: An Intelligent Framework for Handwritten Character Recognition

Pooja[1], Meenakshi Arora[2]

P.G. Student, Department of CSE, Sat Kabir Institute of Technology and Management, Haryana, India[1]
Assistant Professor, of CSE, Sat Kabir Institute of Technology and Management, Haryana, India[2]

ABSTRACT:

A fundamental problem in computer vision and pattern recognition is the recognition of handwritten characters, particularly when noise artifacts and a variety of writing styles are present. In order to effectively recognize handwritten letters using neural networks, this study suggests Neural Decipher, an intelligent and flexible framework. The approach is based on an ordered pipeline: in order to minimize computing complexity, the system first converts the RGB input image to grayscale before acquiring the image. The image clarity is improved by further noise reduction techniques, which are then converted to a binary image to enable accurate bounding box retrieval of character zones. After being segmented to separate individual characters, these recovered areas are fed into a trained neural network model for categorization. Ultimately, the identified characters are found and shown as output text. This entire procedure obtains high recognition accuracy and shows resilience in managing a range of input situations. The system has potential uses in automated form processing, digital archiving, and multilingual handwriting analysis.

KEYWORDS: Image processing, handwritten character recognition, Neural Cipher

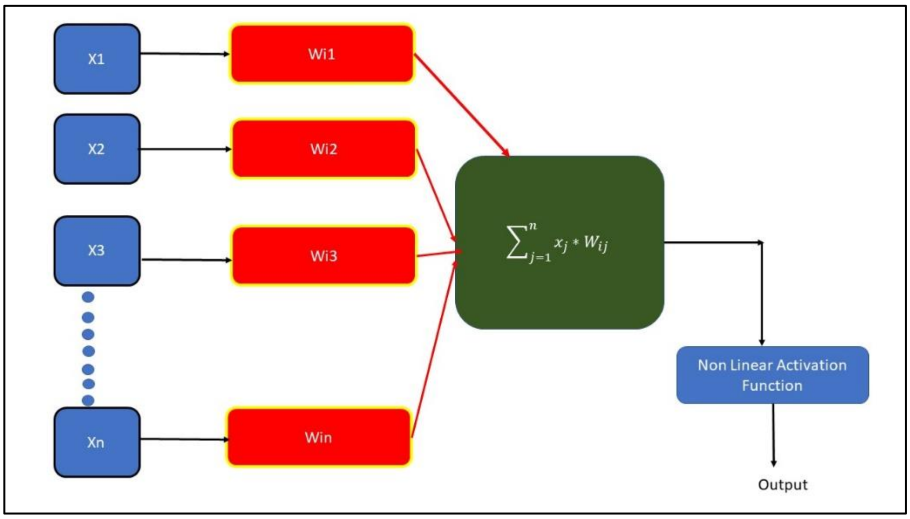
INTRODUCTION:

Handwriting is the most typical and systematic way of recording facts and information. The handwriting of an individual is idiosyncratic and unique to individual people. The capability of software or a device to recognize and analyze human handwriting in any language is called a handwritten character recognition (HCR) system. Recognition can be performed from both online and offline handwriting. In recent years, applications of handwriting recognition are thriving, widely used in reading postal addresses, language translation, bank forms and check amounts, digital libraries, keyword spotting, and traffic sign detection.

In image categorization, neural networks have advanced significantly. Neural networks can be effectively operated on direct pictures using convolutional neural networks (CNN). These days, handwritten character recognition (HCR) is a very effective tool for a variety of tasks, including document information extraction, language translation, and traffic signal detection. Despite the industry's usage of handwritten character recognition technologies, accuracy is currently subpar, which affects both usability and performance [1-2]. Thus, the character recognition technologies in use are still not very reliable and need further improvement to be extensively deployed for serious and reliable tasks.

Neural networks (NNs), a specialized subset of machine learning algorithms, are designed to mimic the information-processing mechanisms of the human brain. In a neural network, the depth—reflected by the number of layers—defines whether it qualifies as deep learning. Neurons, serving as the fundamental information-processing units, form the backbone of the network and are inspired by biological neural systems. Key components of a neural network include inputs, weights, biases, and outputs. The strength of the connections between neurons is represented by weights, while biases help adjust the output independently of the input. Each processing node in the network is referred to as a perceptron [3].

Figure 1: A Basic Neural Network



Research Background

The human brain served as the inspiration for the development of convolutional neural networks, or CNNs. A child can guess an object they have never seen before since people can recognize items from their childhood because they have seen hundreds of photos of those items. CNNs operate similarly. CNNs are a type of fully connected MLP deep neural network that is utilized for visual image analysis. When a layer's neurons are totally connected to every other layer's neurons, it is said to be fully connected. Since CNN models don't require prior knowledge of designer features, they can produce outstanding recognition results. CNNs, on the other hand, are independent of input image rotation.

Utilizing the MNIST dataset, a CNN model has been substantially set for the HCR system. Such studies have been conducted for a number of years. According to some researches, handwritten digits can be recognized with a reliability of up to 99% [4]. A mixture of many CNN models was used in an experiment for MNIST digits, and the accuracy was 99.73% [5]. Later, when this 7-net committee experiment was expanded to a 35-net committee, the detection accuracy increased to 99.77% for the identical MNIST dataset [6]. By incorporating the SVM for MNIST digit recognition, Niu and Suen were able to reduce the structural risk and achieve an impressive 99.81% accuracy rate [7]. A CNN was used to study the recognition of Chinese handwritten characters [8]. Alvear-Sandoval et al. recently developed deep neural networks (DNN) for MNIST, achieving an error rate of 0.19% [9]. However, following a thorough examination, it was shown that the MNIST dataset's maximum recognition accuracy may be achieved with only ensemble approaches, as these help to improve the precision of classification. There are trade-offs, nevertheless, such as higher testing complexity and greater computing cost [10].

Although the industry has been using HCR technology for a long time and research on it has been ongoing, the accuracy of the technology is low, which affects its overall performance and usability. The current state of character recognition technology is still lacking in reliability and requires further advancement before it can be widely used for reliable applications. Therefore, the study [1] proposes a customized CNN model using two distinct datasets of handwritten images to accomplish digit identification and English alphabet character recognition. Authors in [11] recently presented a novel method for scene text recognition called the conventional recurrent neural network (CRNN), which combines the deep CNN (DCNN) and recurrent neural network (RNN). They claimed that this method outperformed more standard approaches for character identification. Badrinarayanan et al. suggested a deep convolutional network design for semantic segmentation, utilizing the max-pooling layer to achieve good performance. The authors also conducted a comparison between their model and existing methods. A pixel-wise classification layer, an encoder network, and a decoder network make up the segmentation architecture called SegNet [12]. CNN has demonstrated exceptional performance in offline HCR for a variety of regional and foreign languages. Chinese handwritten text recognition [13], Arabic [14], Urdu text analysis [15] and Tamil character detection [16], Telugu character detection [46], and Indic scripts [47] have all been the subject of research initiatives.

PROPOSED METHODOLOGY

1. **Image Acquisition:** In any HCR system, this is the initial stage. It entails taking a picture of handwritten text with a tablet, scanner, or digital camera. The overall accuracy of identification is influenced by the quality of acquisition. Usually, the end product is a grayscale or colorful image with handwritten characters.
2. **RGB to Grayscale Conversion:** Most images are captured in RGB (Red, Green, Blue) format. However, color information is not necessary for character recognition. Therefore, the RGB image is converted into a grayscale image by calculating a weighted average of the RGB channels. This simplifies processing and reduces computation.
3. **Noise Elimination:** Noise such as smudges, dust particles, or scanner artifacts can reduce the clarity of the characters. We have used morphological operations (erosion/dilation) to clean the image. This improves segmentation and recognition accuracy.
4. **Grayscale to Binary Image:** The grayscale image is then converted into a binary image, where pixels are either black (foreground/text) or white (background). This is typically done using thresholding techniques (Otsu's method), which automatically determines the optimal threshold value.
5. **Bounding Box Extraction:** A bounding box is drawn around each character or word to isolate it. This involves finding the connected components or regions in the binary image and enclosing them in rectangles. Each box ideally contains a single character.
6. **Image Segmentation:** Segmentation involves dividing the text image into individual characters. This can be done line-wise, word-wise, and then character-wise. Common techniques include projection profiling (horizontal/vertical histograms) and contour detection. We have used contour detection method for image segmentation.
7. **Feed to Neural Network:** Once characters are segmented and resized to a standard size (e.g., 28×28 pixels), they are flattened into feature vectors and fed into a neural network, such as a Convolutional Neural Network (CNN). The network is trained to classify the input into corresponding character classes.
8. **Detecting and Displaying Text (Characters):** The output of the neural network is a label for each character. These are combined sequentially to reconstruct the original text string. The recognized text is then displayed on the screen or stored for further use. Figure 2 shows a working flow of the proposed method.

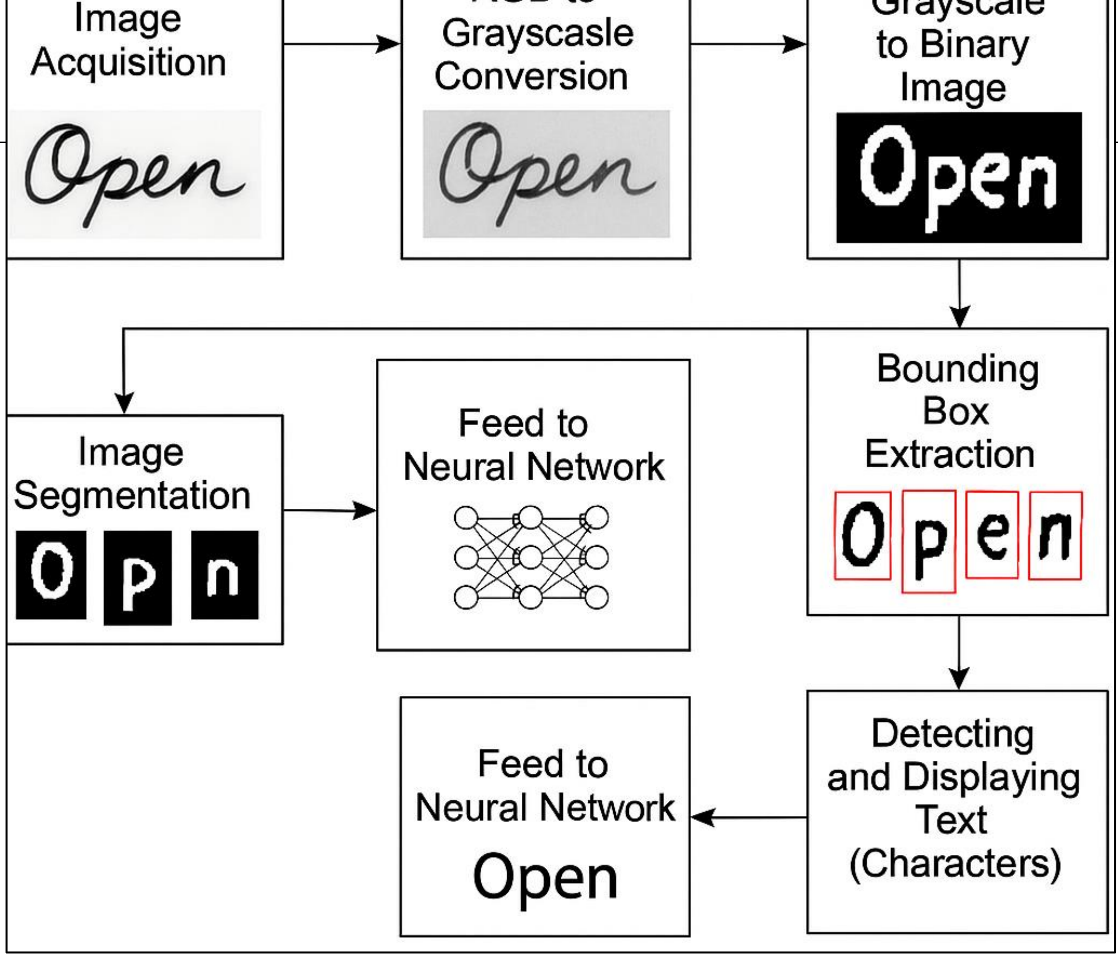


Figure 2: Working flow of the proposed methodology

SIMULATION ENVIRNEMNT AND OUTCOMES:

We have created handwritten character images by paint brush application [17]. We have implemented the proposed method in MATLAB 2024a [18]. We have conducted following tests. We have achieved 99 % accuracy.

Test 1: Input Characters: TK

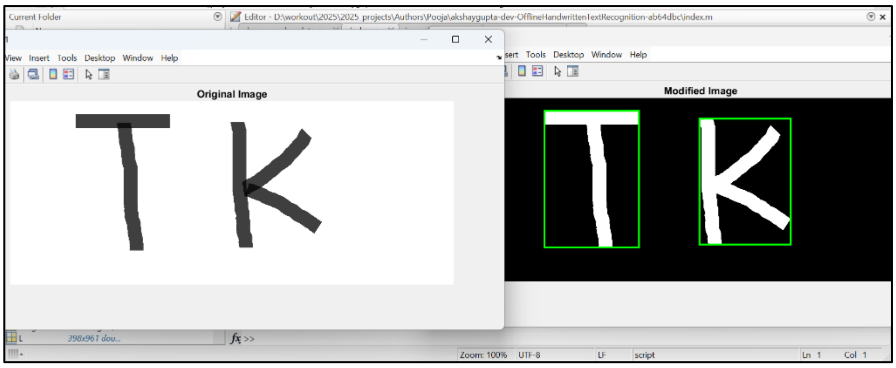


Figure 3a: Input Text (TK)

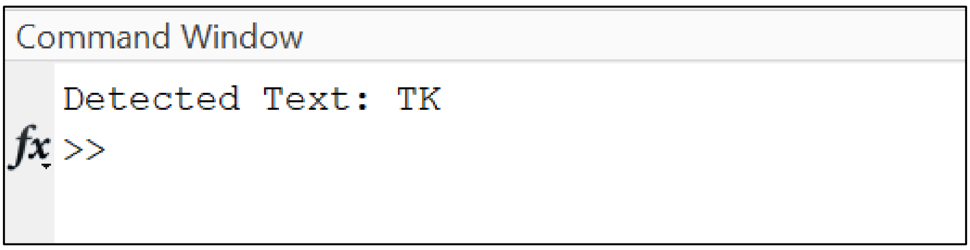


Figure 3b: Character Identified Successfully

Test 2: Input Character RP

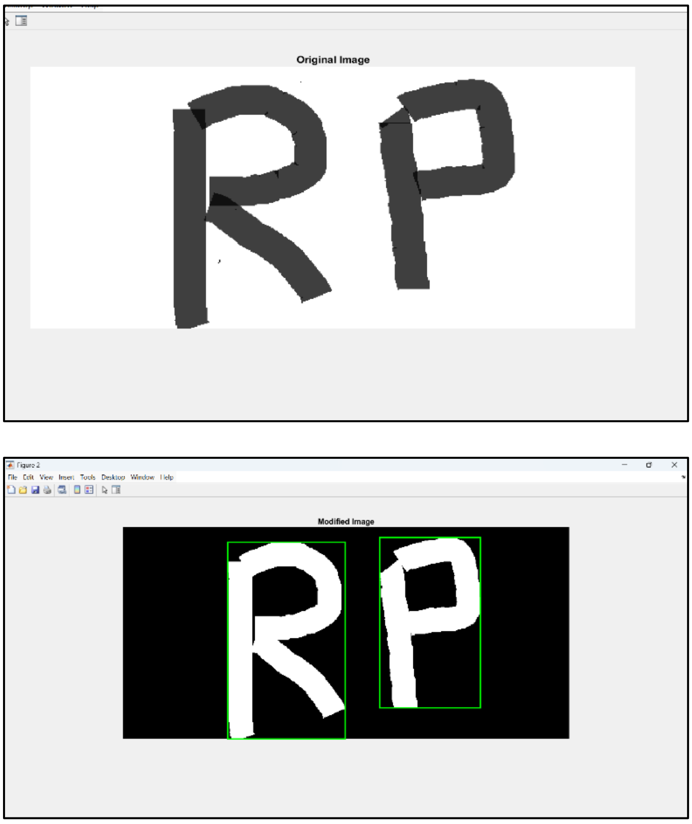


Figure 4a: Input Character RP

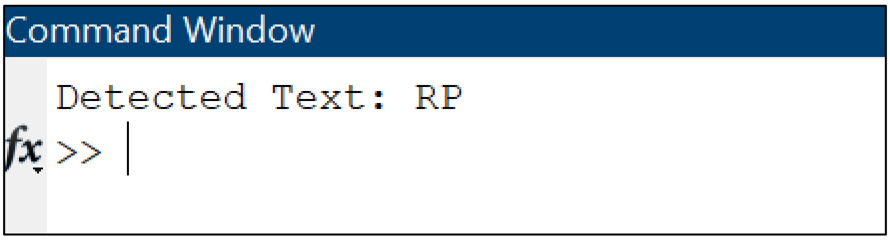


Figure 4b: Characters Detected Successfully

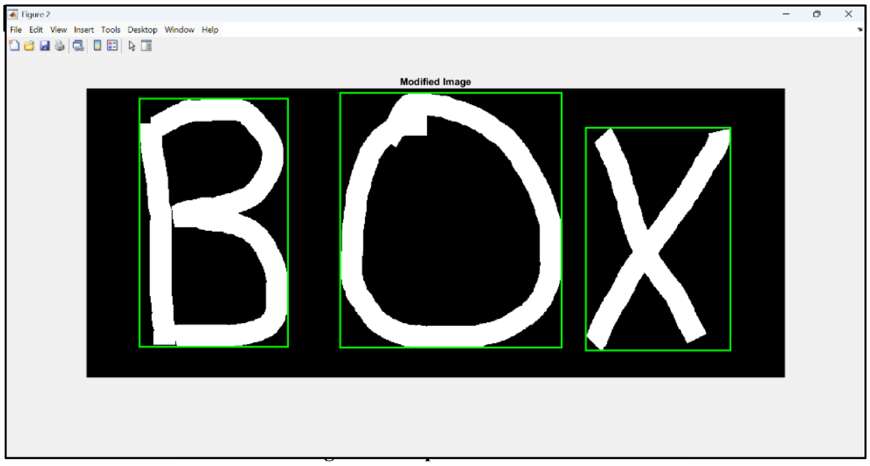
Test 3: Input word: BOX

ANALYSIS

Table 1: Proposed Method vs. Other HCR Techniques

Feature	**Proposed Method (Traditional + Neural Network)**	**CNN-Based Method**	**HOG + SVM-Based Method**	**End-to-End Deep Learning (RNN/Transformer)**
Pipeline Modularity	Modular (clear stages: preprocessing, segmentation, classification)	Integrated (feature extraction + classification)	Modular	Fully integrated (image-to-text)
Preprocessing Required	Yes (grayscale, noise removal, binarization)	Minimal (resizing, normalization)	Yes (HOG feature extraction)	Very minimal
Feature Extraction	Implicit (via bounding box & segmentation)	Automatic (via convolution layers)	Explicit (HOG or similar descriptors)	Automatic

Segmentation	Required	Not required	Required	Not required
Classifier	Feed-forward Neural Network	CNN (deep)	SVM	RNN, BiLSTM, Transformer
Accuracy (on standard datasets like IAM, MNIST)	Moderate (80–90%)	High (95–99%)	Moderate (85–92%)	Very high (97–99%)
Complexity	Low to Moderate	High	Moderate	Very high
Hardware Requirements	Low (runs on CPU)	GPU recommended	CPU	High-performance GPU/TPU
Training Time	Short to Moderate	Long	Moderate	Very Long
Flexibility Across Languages	Needs adjustment	High (with training data)	Needs retraining	High (can handle cursive, complex scripts)
Real-time Capability	Yes (lightweight)	Possible with optimization	Yes	May require optimization



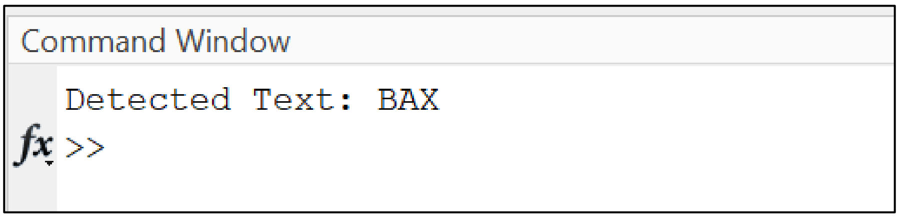


Figure 5b: Characters B and X identified successfully

A simple neural network handles character categorization in the suggested HCR methodology, which uses a conventional, modular pipeline with preprocessing stages including grayscale conversion, noise removal, binarization, and bounding box-based segmentation. For datasets with a moderate level of complexity, this method provides a decent compromise between accuracy and simplicity. It works especially effectively in controlled settings with constant handwriting and image resolution quality. On the other hand, a large portion of the manual preprocessing and segmentation is eliminated by CNN-based techniques. Convolutional neural networks are very good at identifying characters in noisy or diverse datasets because they automatically extract hierarchical characteristics from raw or little edited images. Because of their processing needs, these algorithms typically benefit from GPU acceleration and require a significant amount of labeled data. They are far more accurate and resilient than conventional techniques, although requiring more resources.

CONCLUSION

While the proposed method (image preprocessing + neural network) offers a balance between interpretability, control, and accuracy, it is less flexible than fully automated end-to-end deep learning models. However, for controlled environments with limited data, it remains a powerful, interpretable, and efficient approach. The most sophisticated methods make use of end-to-end deep learning models like Transformers, BiLSTMs, and RNNs. Without explicit segmentation, these models can process entire lines or paragraphs of text. They are perfect for identifying cursive scripts or mixed-language inputs since they can process sequential and contextual information. Although these systems provide cutting-edge accuracy, they require a large amount of training data and substantial processing power. Compared to modular approaches, their intricacy also makes them less transparent and more difficult to debug. In conclusion, the suggested conventional neural network-based approach is perfect for academic or lightweight applications where modular architecture and interpretability are crucial. HOG+SVM is a compromise option for easier deployments, whereas CNNs offer a balance between automation and performance. Although they are more expensive and sophisticated, end-to-end deep learning models are still the best option for high accuracy requirements across difficult or varied datasets.

REFERENCES

1. Saqib, N.; Haque, K.F.; Yanambaka, V.P.; Abdelgawad, A. Convolutional-Neural-Network-Based Handwritten Character Recognition: An Approach with Massive Multisource Data. Algorithms 2022, 15, 129.
2. Pooja, Meenakshi Arora, Rohini Sharma, "From Pixels to Patterns: A Review of Handwritten Character Recognition Methods", IJIRE-V6I3-119-123.
3. Vinh, T.Q.; Duy, L.H.; Nhan, N.T. Vietnamese handwritten character recognition using convolutional neural network. IAES Int. J. Artif. Intell. 2020, 9, 276–283.
4. Jarrett, K.; Kavukcuoglu, K.; Ranzato, M.; LeCun, Y. What is the best multi-stage architecture for object recognition? In Proceedings of the IEEE 12th International Conference on Computer Vision (ICCV), Kyoto, Japan, 29 September–2 October 2009; pp. 2146–2153.
5. Cireşan, D.C.; Meier, U.; Masci, J.; Gambardella, L.M.; Schmidhuber, J. High-Performance Neural Networks for Visual Object Classification. arXiv 2011, arXiv:1102.0183v1.
6. Ciresan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 3642–3649.
7. Niu, X.X.; Suen, C.Y. A novel hybrid CNN-SVM classifier for recognizing handwritten digits. Pattern Recognit. 2012, 45, 1318–1325.
8. Qu, X.; Wang, W.; Lu, K.; Zhou, J. Data augmentation and directional feature maps extraction for in-air handwritten Chinese character recognition based on convolutional neural network. Pattern Recognit. Lett. 2018, 111, 9–15.
9. Alvear-Sandoval, R.F.; Figueiras-Vidal, A.R. On building ensembles of stacked denoising auto-encoding classifiers and their further improvement. Inf. Fusion 2018, 39, 41–52.
10. Demir, C.; Alpaydin, E. Cost-conscious classifier ensembles. Pattern Recognit. Lett. 2005, 26, 2206–2214.
11. Shi, B.; Bai, X.; Yao, C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2017, 39, 2298–2304.
12. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 2018, 40, 834–848.
13. Wu, Y.C.; Yin, F.; Liu, C.L. Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models. Pattern Recognit. 2017, 65, 251–264.
14. Boufenar, C.; Kerboua, A.; Batouche, M. Investigation on deep learning for off-line handwritten Arabic character recognition. Cogn. Syst. Res. 2018, 50, 180–195.
15. Husnain, M.; Missen, M.M.S.; Mumtaz, S.; Jhanidr, M.Z.; Coustaty, M.; Luqman, M.M.; Ogier, J.M.; Choi, G.S. Recognition of urdu handwritten characters using convolutional neural network. Appl. Sci. 2019, 9, 2758.
16. Kavitha, B.R.; Srimathi, C. Benchmarking on offline Handwritten Tamil Character Recognition using convolutional neural networks. J. King Saud Univ.-Comput. Inf. Sci. 2019, 34, 1183–1190.
17. https://paintbrush.sourceforge.io/
18. https://in.mathworks.com/products/new_products/release2024a.html.