

****Paper Title:****

Arabic handwriting recognition system using convolutional neural network

****Paper Link:****

<https://link.springer.com/article/10.1007/s00521-020-05070-8>

****1 Summary****

****1.1 Motivation****

The paper aims to explore and address issues related to automatic Arabic Handwriting detection using machine learning, providing insights into the challenges associated with this task.

****1.2 Contribution****

The primary contribution of the paper lies in its identification of biases within AI models employed for handwriting detection. It emphasizes the inadequacy of existing models in solving bias and highlights the need for further improvements in the handwriting detection areas

1.3 Methodology*

The study utilized a set of handwritten characters for evaluating Arabic Handwriting detection models. The characters were collected through charity events. The responses to these prompts were meticulously observed and characterized.

****1.4 Conclusion****

In conclusion, the development and implementation of an Arabic handwriting recognition system using Convolutional Neural Networks

(CNNs) mark a significant advancement in the field of character recognition. The CNN model has demonstrated remarkable accuracy and efficiency in deciphering the intricate nuances of Arabic script, a complex and cursive writing system. Through extensive training on diverse datasets, the system has exhibited robust generalization capabilities, making it adaptable to various handwriting styles and ensuring its practical utility across a broad spectrum of users.

****2 Limitations****

1. **Data Diversity:** The system's performance heavily relies on the diversity and representativeness of the training data. Limited access to diverse and comprehensive datasets may result in reduced accuracy, especially when faced with unusual or less common handwriting styles.
2. **Resource Intensiveness:** CNNs, being complex models, may demand significant computational resources for training and deployment. This can pose challenges for implementation on resource-constrained devices or in environments with limited computing power.