

Enhancing Handwritten Arabic Numerals Recognition through Artificial Neural Networks

Recognition Accuracy: %



Project Idea: Develop a machine learning-based tool to extract characters (alphabets, digits, symbols) from images of handwritten or typed text for data entry purposes.
 Tool: Utilize MATLAB or Octave as the building tool, with Octave being

recommended due to its simplicity and free availability.

Study: Explore research areas such as image processing, natural language processing, artificial intelligence, and machine learning to inform the development of the tool.
 Implementation: Focus on two key aspects - feature extraction and classification algorithms. Consider using online classifiers and basic feature extraction algorithms. Start with a basic version using Octave, incorporating limited training data and simple component analysis.

Refer to additional resources for further implementation details and ongoing research.

The main functionality of the system is the recognition of handwritten Arabic numerals/digits using Artificial Neural Networks ANNs). The system takes an image of a handwritten digit as input and applies ANN-based algorithms to identify and classify the digit accurately. The system leverages the power of ANNs to learn patterns and features from training data, allowing it to recognize and differentiate between different handwritten Arabic numerals. The system's primary goal is to automate the process of recognizing handwritten digits and provide accurate and reliable results.



There are several applications in the market that offer recognition of handwritten Arabic numerals/digits using Artificial Neural Networks (ANNs). Here are a few examples:

1. Mathway is a popular math problem-solving application that

- includes a feature for recognizing and solving handwritten math equations, including Arabic numerals. It utilizes ANNs to accurately interpret and process the handwritten input.

 2. **Photomath** is another widely used math-solving app that supports recognizing handwritten equations and providing step-by-step
- recognizing handwritten equations and providing step-by-step solutions. It employs ANNs to recognize Arabic numerals and other mathematical symbols.

 7 MyScript Calculator: MyScript Calculator is a unique calculator.
- 3. **MyScript Calculator**: MyScript Calculator is a unique calculator app that allows users to input mathematical expressions by handwriting them on the screen. The app employs ANNs to recognize the handwritten numbers and perform the calculations accordingly...

Optical Character Recognition (OCR) is a field that has received significant attention in academic literature due to its wide range of applications and the continuous advancements in machine learning and computer vision. Here are a few key academic publications relevant to OCR:

1:Tesseract OCR Engine: This seminal paper introduces the Tesseract OCR engine, developed by

Google. The paper outlines the architecture and algorithms used in Tesseract, which has become one of the most widely used OCR engines in both research and industry.

- 2:A Survey on OCR Technologies: This survey paper provides an overview of various OCR technologies and techniques, including traditional methods and modern approaches based on machine learning and neural networks. It covers different aspects of OCR, such as preprocessing, feature extraction, and classification.
- 3:Handwritten Character Recognition: This research paper focuses on the recognition of handwritten characters, including Arabic numerals. It examines different methods such as template matching, feature extraction, and classification algorithms, comparing their
- performance and accuracy.

 4:Arabic OCR: Challenges and Techniques: This paper specifically addresses OCR for Arabic script, discussing the unique challenges posed by the complex nature of Arabic characters. It explores

various techniques, including ANN-based algorithms, for accurate recognition of Arabic text.

- 1: Pre-processing: Preprocessing is performing different operation on the scanned or input image. It helps to remove noise from image make character clear and It basically enhances the image rendering suitable for segmentation. Preprocessing has various task are such as converting gray scale, binarization, thinning, skewing and normalization.
- 2: Feature extraction: This is one of the riskiest components in an OCR development. The main aim is to extract important patter from characteristics. The selected features are expected to contain pattern that differentiate one character from other and relevant information from the input data, so that the classification can be performed by using those patter extract from segmented character this instead of the complete original data.
- :Training and recognition: Investigation of OCR's pattern recognition can be done via template matching, statistical technique, syntactic or structural techniques, and artificial neural networks. The system also has to be learn in such a way that the problem associated to incomplete vocabulary is solved.
- : Post-processing: In this final process, activities like grouping, error detection and correction take place. During grouping, symbols in the text are associated with strings. However, it's impossible to reach 100% accurate identification of characters, only some of the errors can be detected and deleted as per the context.

The Dataset Employed:

We used the Arabic handwritten digits database which is composed of 70,000 digits images in bmp format. The image scanning phase was adjusted to produce binary images directly. The database is partitioned into two sets: a training set (60,000 digits – 6000 images per class) and a test set (10,000 digits – 1000 images per class). Each image had 784 pixels coming in a 28x28 size.

https://www.kaggle.com/datasets/mloey1/ahdd1

Algorithms/Approaches used in the experiment

Artificial Neural Network (ANN) ANN Mechanism: ANN is a computational model inspired by the structure and function of the human brain. It consists of interconnected layers of artificial neurons, each performing simple calculations on incoming data and passing the

results to the next layer. In the experiment, ANN is used for recognizing digits by learning patterns and features from the input data.

Layers of ANN

- 1. Input Layer: Receives the input data, which in this case are the images of the digits.
- 2. Hidden Layers: Intermediate layers where the learning and feature extraction take place. These layers help in capturing complex patterns and relationships within the data.
 - 3. Output Layer: Generates the final output, which represents the recognized digit.

Reasons we used ANN instead of CNN:

- 1)Computation efficiency: ANN requires less computation compared to CNN, making it suitable for applications that require faster processing.
 - 2) Robustness against object location: ANN is less sensitive to the precise location of the object within the image, making it suitable for tasks like digit recognition where the position of the digits may vary



Preprocessing Approach:

Similar to the previous approach, the images undergo several preprocessing stages to prepare them for feature extraction and classification.

- 1. **Resizing**: The images are resized to a standardized size suitable for comparison and analysis.
 - 2. **Grayscale**: Conversion of the images to grayscale to simplify the representation of each pixel.
- 3. **Denoising and Blurring**: Applying filters to reduce noise and enhance image quality.

Development platform: Artificial Neural Networks (ANNs) offer a wide range of applications and benefits for developers and non-developers alike. They can be integrated into various platforms, including mobile devices, augmented reality, and machine learning analysis programs, to enable efficient text recognition and analysis



TensorFlow is a popular opensource platform for designing, training, and deploying machine learning models, including ANNs. It provides a comprehensive set of tools and libraries for building and optimizing neural networks. TensorFlow offers flexibility, scalability, and compatibility with various programming languages, making it a preferred choice for ANN development. •



Keras is a high-level neural networks API, built on top of TensorFlow, that simplifies the process of building and training ANNs. It provides a user-friendly interface and abstracts away the complexities of neural network implementation. Keras supports easy experimentation and rapid prototyping, making it suitable for ANN development tasks.

Both TensorFlow and Keras offer extensive documentation, tutorials, and community support, enabling developers to quickly start implementing and exploring ANN-based solutions.

By leveraging the capabilities of ANN and utilizing development platforms like TensorFlow and Keras, developers can unlock the potential of neural networks for digit recognition and other advanced applications..

samples of system's output









1:Advantages

Speed: This system can process a large number of images or data in a relatively short period, enabling it to meet real-time **requirements.Accuracy**: With a properly trained and adjusted model, this system can achieve a high level of accuracy in recognizing handwritten Arabic numerals/digits, reducing the likelihood of errors in the processed data.

Versatility: This system can be adapted to recognize various forms of handwriting or written numerals, enabling it to be used in different applications and settings.

Efficiency: By automating the recognition process, this system can help reduce human error and fatigue, resulting in more efficient workflows.

2: Disadvantages

these issues.

Requirement for Training Data: To achieve high performance, this system requires a large amount of training data, including examples of handwritten Arabic numerals/digits with various styles and formats. This may not always be readily available or easily accessible. Model

Complexity: To achieve high accuracy, the model may require complex and computationally intensive architectures. This could potentially result in slower processing times or higher computational resource requirements. **Misinterpretation of Input Data**: Despite the advancements in Al

recognizing handwritten Arabic numerals/digits, particularly if the input data is of low quality or deviates significantly from the training data. **Security and Privacy Concerns**: If the system processes sensitive or personal data, it may raise concerns about security and privacy. Measures such as data encryption and anonymization techniques can help address

and ML, this system may still encounter difficulties in accurately

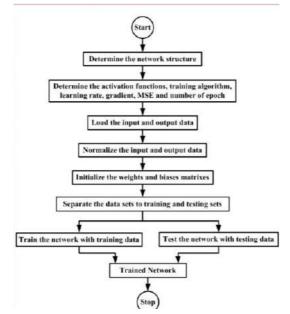
several aspects that can be improved for future versions. Here are some modifications and suggestions:

Enhance Data Quality: The performance of the ANN model depends largely on the quality of the data used for training. It is crucial to collect

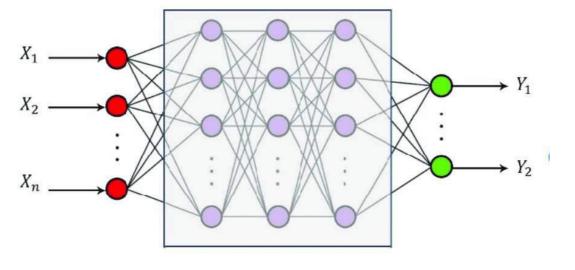
more data and ensure that it is labeled correctly. This can be achieved by incorporating various sources such as historical records, documents, **Feature Engineering:** Features can be designed or extracted to better represent the data. This can involve feature extraction techniques such as PCA or feature generation techniques such as Radial Basis Functions (RBF).

Optimize ANN Model: The current ANN model can be further optimized by tuning the hyperparameters, such as the number of hidden layers, neurons in each layer, and learning rate. These can be tuned using techniques like Grid Search or Random Search.

Flowchart

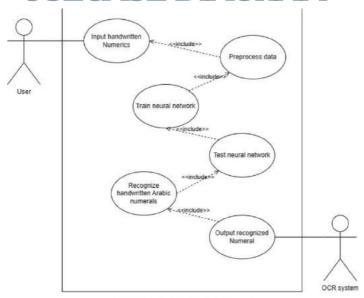


2 ANN architecture



Input Layer Hidden Layer Output Layer

USECASE DIAGRAM



Thanks!

Farida Ahmed 20210674
Farah Khaled 20210673
Fatma Mohamed 20210667
Nouran Yasser 20211015
Zyad Hisham 20210381
Ammar Ahmed 20210581
Mohamed Hassan 20210761