Arabic Handwritten Characters Recognition

Machine Learning Nanodegree Capstone Project by Amr Hendy

Domain Background

The automatic recognition of text on scanned images has enabled many applications such as searching for words in large volumes of documents, automatic sorting of postal mail, and convenient editing of previously printed documents. The domain of handwriting in the Arabic script presents unique technical challenges and has been addressed more recently than other domains. Many different methods have been proposed and applied to various types of images.
Here we will focus on the recognition part of handwritten Arabic letters and digits recognition that face several challenges, including the unlimited variation in human handwriting and the large public databases.
There are many related papers that introduced some solutions for that problem such as :

- **HMM Based Approach for Handwritten Arabic Word Recognition**
- **An Arabic handwriting synthesis system**
- **Normalization-Cooperated Gradient Feature Extraction for Handwritten Character Recognition**

Problem Statement

In this project we want to build a model which can classify a new image to an arabic letter or digit. We can use machine learning classification algorithms or deep learning algorithms like CNN to build a strong model able to recognize the images which high accuracy.
There are some related kernels which discuss this problem on kaggle **here**
To Conclude our input of the model will be an image and the output will be one of the arabic letters or digits which almost represents the image

.

Datasets and Inputs

We will use these datasets Arabic Digits and Arabic Letters for arabic letters and arabic digits respectively.
All the datasets are csv files representing the image pixels values and their corresponding label.
Here are some more details about the datasets:

- Arabic Digits Dataset represents **MADBase** (modified Arabic handwritten digits database) which contains **60,000 training images**, and **10,000 test images**. MADBase was **written by 700 writers**. Each writer wrote each digit (from 0 -9) ten times. To ensure including different writing styles, the database was gathered from different institutions: Colleges of Engineering and Law, School of Medicine, the Open University (whose students span a wide range of ages), a high school, and a governmental institution.
  MADBase is available for free and can be downloaded from here**.**
- Arabic Letters Dataset is composed of **16,800 characters written by 60 participants**, the age range is between 19 to 40 years, and 90% of participants are right-hand. Each participant wrote each character (from ’alef’ to ’yeh’) ten times. The images were scanned at the resolution of 300 dpi. Each block is segmented automatically using Matlab 2016a to determining the coordinates for each block. The database is partitioned into two sets: a **training set (13,440 characters to 480 images per class)** and a **test set (3,360 characters to 120 images per class)**. **Writers of training set and test set are exclusive**. Ordering of including writers to test set are randomized to make sure that writers of test set are not from a single institution to ensure variability of the test set.

Solution Statement

I will build a CNN model which is more appropriate for the images classification problems. Then I will tune the parameters using grid search to find the best parameter values to be used in the model and compare the results obtained from the previous models.

Benchmark Model

We can compare our solution with a simple MLP model with small number of layers, machine learning models such as (KNN, Decision Trees, etc).
I prefer to compare the final solution with MLP models or CNN with low number of layers as they will provide some good results too.

Evaluation Metrics

I will use accuracy, precision, recall and F1 score metrics to compare the results obtained from the different models.

Project Design

1. Data Exploration including
    - Reading dataset files.
    - Visualization of some images.
2. Data Preprocessing including
    - Image values normalization.
    - One Hot Encoding for labels.
    - Splitting dataset into training, validation and testing.
3. Design the Model Architecture
4. Augmenting Images (if applicable)
5. Parameters Tuning
6. Training the model
7. Testing the model
8. Evaluating the model using the specified metrics.
9. Comparing the obtained results with the benchmark models.