Optical Character Recognition (OCR) - Group 14

Samanyu Kamra
samanyu21487@iiitd.ac.in

Jayshil Ketankumar Shah
jayshil23138@iiitd.ac.in

Deep Shekhar
deep20193@iiitd.ac.in

Prem Kamal Jain
prem21483@iiitd.ac.in

Nikhil Kumar Y
nikhil22045@iiitd.ac.in

1 Introduction

In the vast realm of information technology, the ability to convert unstructured data into structured and usable formats is paramount. This capability is especially crucial when dealing with physical documents in an increasingly digital world. The solution lies in a technology known as **Optical Character Recognition (OCR)**, which has dramatically reshaped the way we approach text digitization. Optical Character Recognition, or OCR, is a technological marvel that translates printed or handwritten characters from digital images or scanned documents into machine-readable text. This translation facilitates text searching, editing, and storage, enabling seamless integration of physical documents into digital workflows.

2 Dataset Overview

Description: The dataset, obtained from a reliable source, contained two versions:

- data (new version with greater pixels intensity)
- data2 (older version)

We used data directory for building our model. This directory was later sub divided into Training and Testing sample folders.

Each of the training and testing sample contained 36 sub-directories, which comprised of 26 Alphabets and digits from 0 to 9.

3 Exploratory Data Analysis (EDA)

Understanding and Exploring is the initial step for building any Classical Machine Learning Model. The EDA of OCR dataset involved:

1. **Average Dimension of the Images:** We plotted the histograms for image height and width.

   From the Image Width Distribution (blue histogram), it appears that the majority of image widths are clustered around two primary values, suggesting there might be two common widths among your images. There's a significant peak around 20-30 pixels and a smaller peak around 50-60 pixels. The distribution is bimodal, meaning it has two distinct modes or peaks. This indicates that there may be two groups of images with different standard widths or perhaps two types of characters that tend to have different widths.

   The Image Height Distribution (green histogram) shows a more unimodal distribution, centered around 40-50 pixels in height. This suggests that the images are more consistent in height than in width.

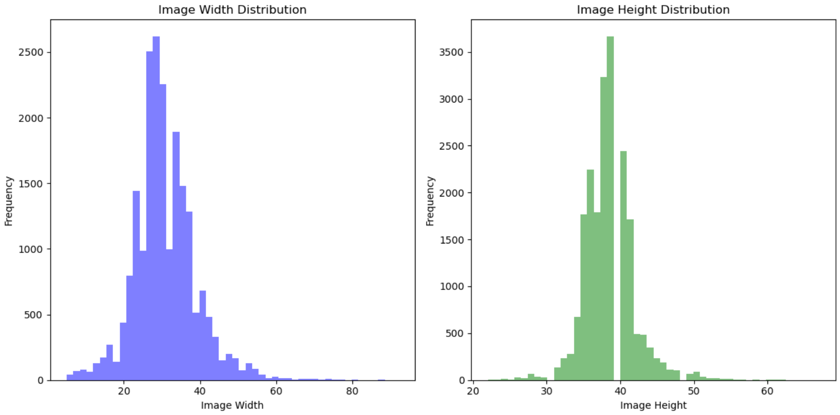

Figure 1: Histograms showing the distribution of image widths and heights.

2. **Distribution of different classes:** On looking at the barplot we found uniform distribution and enough training samples in our dataset.

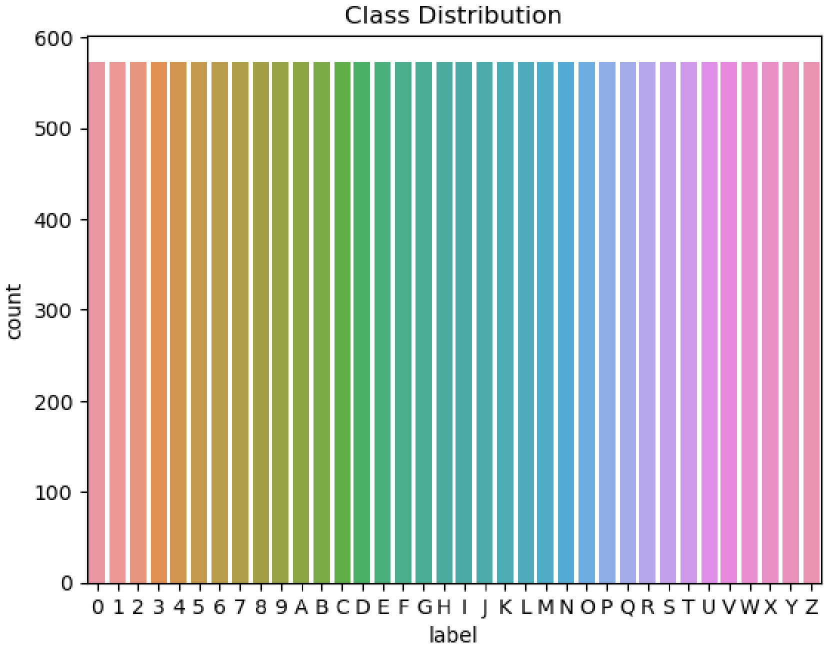

Figure 2: Barplot showing class distribution.

3. **Distribution of Aspect Ratio:** The most common aspect ratios are clustered around 1.0, which suggests that, on average, images in your dataset are as wide as they are tall, indicating a square-like shape.

   Outliers: There are relatively few images with very high or very low aspect ratios (approaching 2 or 0.25), which could be considered outliers. These could represent atypical images that might require special handling or preprocessing.

   Preprocessing Implications: When preprocessing images for OCR, it may be important to consider normalizing the aspect ratios, as images that are too wide or too tall might not be processed optimally by OCR algorithms that expect more standardized input.

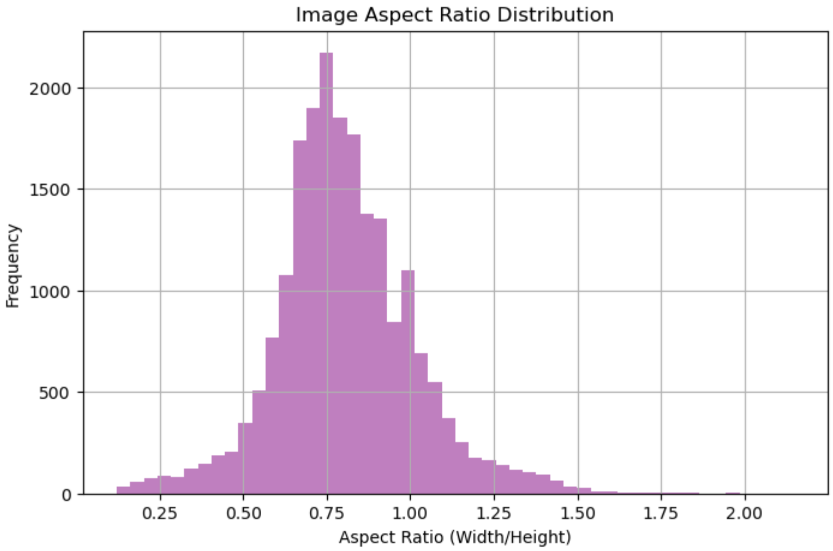

Figure 3: Histogram for aspect ratio of images.

4. **Pixel Intensity Histogram and Outliers:** Pixel Intensity Histogram: This shows the distribution of pixel intensities within a set of images for the character 'Q'. In grayscale images, pixel intensities range from 0 (black) to 255 (white). The histogram has a high frequency of pixels at the higher end of the intensity spectrum, near 255, which suggests that the character 'Q' is likely white or light-colored on a darker background.

   The image of Q is possibly an outlier that is not following this pixel distribution.

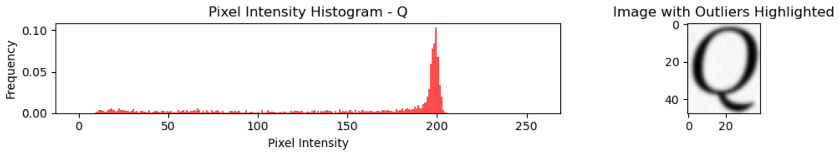

Figure 4: Histogram for pixel intensity and outlier.

4 Data Pre-Processing

Initially the training data was split into training and validation set, ninty percent data was used for training.

Grayscale Conversion: Grayscale conversion is the process of converting an image from color (RGB) to shades of gray, representing the image with a single channel where only intensity information is retained:

1. **Function:** cv2.imread
2. **Role of OpenCV:** It reads an image from a file into an array. OpenCV handles the internal structure of an image file and converts it into a format that can be manipulated in Python.
3. **Input Parameters:** image-path: The file path of the image to be read. cv2.IMREAD-GRAYSCALE: A flag telling OpenCV to read the image as a grayscale image.
4. **Output** A 2-dimensional NumPy array representing the grayscale image. Each element of the array corresponds to a pixel value in the range of 0 to 255, with 0 being black and 255 being white.

Resizing: Resizing an image involves changing its dimensions to a specified width and height, which often requires interpolation to estimate new pixel values.

1. **Function:** cv2.resize
2. **Role of OpenCV:** Resizes the image to a specified size. OpenCV provides several interpolation methods for resizing, which dictate how the pixel values are computed during the scaling process.
3. **Input Parameters:** img: The grayscale image as a 2D array. IMAGE-SIZE: A tuple specifying the desired size (width, height) for the output image.
4. **Output** A 2-dimensional NumPy array of the resized image. The output image has the same number of channels as the input (one, in this case, since it's grayscale), and its dimensions are now equal to IMAGE-SIZE.

Normalization: Normalization in the context of image processing usually refers to scaling pixel values to a standard range, often [0.0, 1.0], to facilitate uniform processing and improve the numerical stability of algorithms.

1. **Role of OpenCV:** This step doesn't explicitly use an OpenCV function, but normalization is a common preprocessing step in image processing pipelines, often done after image data has been loaded and manipulated using libraries like OpenCV.
2. **Input Parameters:** The resized image as a 2D NumPy array with pixel values ranging from 0 to 255.
3. **Operation:** Pixel values are divided by 255.0 to scale them to the range [0.0, 1.0].
4. **Output** A 2-dimensional NumPy array of the same size as the input, where each pixel value is now a floating-point number between 0.0 and 1.0.

Normalization of pixel values to the range [0.0, 1.0] is particularly useful for machine learning models, as it helps to maintain numerical stability and can speed up the learning process by keeping the weights small.

5 Feature Descriptors for OCR

5.1 HOG Features

Definition: The Histogram of Oriented Gradients (HOG) is a feature descriptor that is widely used in computer vision and image processing for object detection, particularly due to its effectiveness in capturing edge and shape information.

Process: HOG features are extracted by:

1. Dividing the image into small spatial regions called cells.
2. Computing a histogram of gradient directions or edge orientations for the pixels within each cell.
3. Normalizing these histograms across larger regions called blocks.
4. Concatenating the normalized histograms to form the feature descriptor.

Input and Output:

- *Input*: A grayscale image.
- *Output*: A feature vector formed by concatenating the normalized histograms of oriented gradients.

Importance: HOG descriptors are robust to changes in illumination and geometric transformations, making them particularly suitable for OCR tasks where character shapes must be recognized consistently.

Mathematics: The gradient computation involves calculating the change in brightness over pixels of the image, and the orientation binning involves accumulating a histogram of gradient orientations.

5.2 Gabor Kernel

Definition: The Gabor kernel, or Gabor filter, is utilized for texture analysis in images – detecting specific frequency content in specific directions.

Process: Gabor features are computed by:

1. Convolution of the image with a Gabor filter, which is a complex sinusoidal wave modulated by a Gaussian envelope.
2. Extracting the filter responses.

Input and Output:

- *Input*: A grayscale image.
- *Output*: The Gabor filter response from which features are derived.

Importance: Gabor filters are sensitive to spatial frequency and orientation, making them effective for OCR by capturing the textural properties that characterize different characters.

Mathematics: A Gabor filter is defined by a sinusoidal plane wave, modulated by a Gaussian envelope, capturing both the frequency and orientation information of the image region it is applied to.

6 Models Trained

In our Optical Character Recognition (OCR) project, we employed both supervised and unsupervised learning approaches for classifying 36 different characters, including 26 alphabets and 10 digits.

6.1 Supervised Learning Models

Support Vector Machine (SVM):

- Classification: Constructs a hyperplane in a high-dimensional space for class separation.
- Parameters: Kernel type, C (regularization), and Gamma. Gamma was set to auto.
- Feature Usage: Uses HOG features to distinguish between character classes.

K-Nearest Neighbors (KNN):

- Classification: Classifies based on the majority class among its 'k' nearest neighbors.
- Parameters: Number of neighbors (K) and distance metric. K was chosen to be 7.
- Feature Usage: Classifies characters based on feature similarity to known examples.

Random Forest:

- Classification: An ensemble of decision trees that give a collective decision.
- Parameters: Number of trees, max depth, and min samples split. Number of trees were chosen as 100.
- Feature Usage: Makes decisions based on subsets of features and samples.

6.2 Unsupervised Learning Model

K-Means Clustering:

- Classification: Groups data into K clusters based on feature similarity.
- Parameters: Number of clusters (K), initialization method, and max iterations. The number of clusters was chosen to be 36, one for each label.
- Feature Usage: Finds natural groupings but does not use label information.

Each model uses the features extracted from the images, like HOG, to classify the different characters. The supervised models learn to distinguish the classes, while K-means clustering groups similar features without class labels.

7 Conclusion and Results

The following table summarizes the performance of the models on both the validation and test sets:

Model Name	**Validation Set Accuracy**	**Test Set Accuracy**
K-Nearest Neighbors (KNN)	95.83%	95.83%
Support Vector Machine (SVM)	97.18%	98.11%
Random Forest	95.78%	98.71%
K-Means Clustering	76.1%	85.32%

Table 1: Accuracy of different models on validation and test sets

7.1 Classification Report Random Forest

The classification report for the OCR model is as follows:

```
classification_report Random Forest:
               precision    recall  f1-score   support

           0       0.75      0.88      0.81        56
           1       0.90      0.97      0.94        39
           2       0.96      1.00      0.98        54
           3       0.94      0.98      0.96        49
           4       0.97      1.00      0.99        67
           5       1.00      1.00      1.00        56
           6       0.95      0.98      0.97        62
           7       0.97      1.00      0.99        73
           8       0.93      1.00      0.97        56
           9       0.95      0.97      0.96        58
           A       0.95      0.97      0.96        63
           B       1.00      0.95      0.98        63
           C       0.97      0.98      0.97        59
           D       0.99      0.99      0.99        69
           E       0.98      0.92      0.95        60
           F       0.95      0.98      0.96        55
           G       0.93      0.90      0.92        63
           H       1.00      0.97      0.98        62
           I       0.94      0.98      0.96        59
           J       0.92      0.97      0.94        58
           K       0.98      0.98      0.98        51
           L       0.95      0.95      0.95        58
           M       0.98      0.97      0.98        62
           N       1.00      0.96      0.98        48
           O       0.84      0.70      0.77        54
           P       0.95      0.95      0.95        65
           Q       1.00      0.93      0.96        55
           R       0.96      0.92      0.94        59
           S       1.00      0.96      0.98        53
           T       1.00      1.00      1.00        67
           U       1.00      0.96      0.98        50
           V       0.95      0.96      0.95        55
           W       0.96      0.94      0.95        54
           X       0.96      0.98      0.97        45
           Y       0.98      0.98      0.98        49
           Z       1.00      0.93      0.96        57

    accuracy                           0.96      2063
   macro avg       0.96      0.96      0.96      2063
weighted avg       0.96      0.96      0.96      2063
```

7.2 Classification Report SVM

The classification report for the OCR model is as follows:

```
Classification Report:
              precision    recall  f1-score   support

           0       0.63      0.81      0.71        42
           1       0.96      0.93      0.95        59
           2       0.98      1.00      0.99        52
           3       1.00      1.00      1.00        52
           4       0.98      1.00      0.99        54
           5       1.00      0.98      0.99        55
           6       1.00      1.00      1.00        60
           7       0.98      1.00      0.99        42
           8       1.00      0.99      0.99        72
           9       0.98      1.00      0.99        58
           A       0.98      0.97      0.98        62
           B       1.00      0.98      0.99        46
           C       0.98      1.00      0.99        45
           D       0.95      1.00      0.97        56
           E       1.00      0.96      0.98        52
           F       0.98      0.97      0.98        65
           G       0.99      0.94      0.96        72
           H       0.98      0.98      0.98        58
           I       0.94      0.97      0.95        64
           J       0.97      1.00      0.98        57
           K       0.98      0.98      0.98        46
           L       1.00      1.00      1.00        66
           M       0.98      1.00      0.99        43
           N       1.00      0.96      0.98        48
           O       0.87      0.74      0.80        72
           P       0.99      1.00      0.99        71
           Q       0.99      0.99      0.99        67
           R       1.00      0.97      0.98        60
           S       0.96      1.00      0.98        69
           T       1.00      1.00      1.00        58
           U       1.00      0.96      0.98        56
           V       1.00      0.98      0.99        60
           W       1.00      0.98      0.99        55
           X       0.98      0.98      0.98        48
           Y       0.96      1.00      0.98        53
           Z       1.00      1.00      1.00        68

    accuracy                           0.97      2063
   macro avg       0.97      0.97      0.97      2063
weighted avg       0.97      0.97      0.97      2063
```

7.3 Classification Report KNN

The classification report for the OCR model is as follows:

```
classification_report KNN:
```

```
              precision    recall  f1-score   support

           0      0.66      0.70      0.68        56
           1      1.00      0.97      0.99        39
           2      0.96      1.00      0.98        54
           3      1.00      1.00      1.00        49
           4      1.00      1.00      1.00        67
           5      1.00      0.98      0.99        56
           6      1.00      1.00      1.00        62
           7      1.00      1.00      1.00        73
           8      0.95      1.00      0.97        56
           9      0.93      0.98      0.96        58
           A      0.98      0.97      0.98        63
           B      0.95      0.97      0.96        63
           C      0.89      0.97      0.93        59
           D      0.97      0.97      0.97        69
           E      0.91      0.87      0.89        60
           F      0.93      0.91      0.92        55
           G      0.98      0.90      0.94        63
           H      0.98      0.95      0.97        62
           I      0.89      0.98      0.94        59
           J      0.98      0.95      0.96        58
           K      0.94      1.00      0.97        51
           L      0.98      0.98      0.98        58
           M      0.98      0.95      0.97        62
           N      0.90      0.96      0.93        48
           O      0.63      0.63      0.63        54
           P      0.94      0.95      0.95        65
           Q      1.00      0.96      0.98        55
           R      0.96      0.88      0.92        59
           S      1.00      0.96      0.98        53
           T      0.98      0.97      0.98        67
           U      0.96      0.96      0.96        50
           V      0.98      1.00      0.99        55
           W      1.00      0.94      0.97        54
           X      0.98      0.98      0.98        45
           Y      0.98      1.00      0.99        49
           Z      1.00      0.98      0.99        57

    accuracy                          0.95      2063
   macro avg      0.95      0.95      0.95      2063
weighted avg      0.95      0.95      0.95      2063
```

7.4 Classification Report K-means

The classification report for the OCR model is as follows:

```
Classification Report:
              precision    recall  f1-score   support

           0      0.33      0.93      0.48        42
           1      0.70      0.81      0.75        59
           2      0.90      1.00      0.95        52
           3      1.00      0.92      0.96        52
           4      1.00      0.96      0.98        54
           5      0.95      0.96      0.95        55
           6      0.97      0.98      0.98        60
           7      0.88      0.90      0.89        42
           8      0.69      0.96      0.80        72
           9       0.96      0.95      0.96        58
          10       0.98      0.92      0.95        62
          11       0.00      0.00      0.00        46
          12       0.40      0.98      0.56        45
          13       1.00      0.89      0.94        56
          14       0.38      0.96      0.54        52
          15       0.00      0.00      0.00        65
          16       0.00      0.00      0.00        72
          17       0.95      0.97      0.96        58
          18       0.83      0.59      0.69        64
          19       0.45      0.96      0.61        57
          20       0.96      0.96      0.96        46
          21       1.00      0.71      0.83        66
          22       0.95      0.93      0.94        43
          23       1.00      0.85      0.92        48
          24       0.00      0.00      0.00        72
          25       0.27      0.20      0.23        71
          26       0.95      0.82      0.88        67
          27       0.40      0.58      0.47        60
          28       0.98      0.94      0.96        69
          29       0.98      0.97      0.97        58
          30       1.00      0.91      0.95        56
          31       0.84      0.95      0.89        60
          32       1.00      0.82      0.90        55
          33       1.00      0.94      0.97        48
          34       0.98      0.91      0.94        53
          35       0.96      0.94      0.95        68

    accuracy                           0.76      2063
   macro avg       0.74      0.78      0.74      2063
weighted avg       0.73      0.76      0.73      2063
```

What we observed from these classification report was that the model had difficulty identifying between 0 and O.

8 Literature Survey(Previous Deadline)

8.1 Paper 1: Statistical Learning for OCR Text Correction

This paper proposes steps for improving the accuracy of Optical Character Recognition by post processing using different features through a learning process.The model uses a regression approach on OCR-specific features that can predict and correct errors in a text translated using Optical Character Recognition.This model recognises errors caused by insertion, deletion, substitution or transposition in words , errors caused by factors such as scanning errors caused by low quality of paper or poor condition of scanning equipment, zoning error caused by complex page layout,errors caused by broken characters, overlapping characters or non-standard fonts and errors caused by difference in characteristics of algorithms.

8.2 Paper 2: Optical Character Recognition Techniques: A Review

This paper discusses taking the image data, menu scripts, handwritten scripts, as input and using the SVM Classifier that attains the accuracy of 92%. For Gujarati scripts, it uses k - NN classifier and gives an accuracy of 86.33%. For Thai, Bangla and Latin languages, in this paper, MNIST dataset is used. For character classification, algorithms such as Support Vector Machine (SVM) and k-nearest neighbor (k-NN) are used. Support Vector Machine gives better results as compared to k-NN. It achieves a highest accuracy of 98.93% on Thai dataset.

8.3 Paper 3: OCR Based Image Text to Speech Conversion using K-Nearest Neighbors and Comparing with Fuzzy K-Means Clustering Algorithm

This paper discusses Optical character recognition using KNN and FKM .The research paper aims to advance the field of assistive technology by focusing on the conversion of visual text into spoken language. Two machine learning algorithms, Novel K-Nearest Neighbours (KNN) and Fuzzy K-Means Clustering (FKM), are evaluated for their efficacy in this transformation process. Rigorous statistical methodologies are employed, utilizing SPSS software and independent sample t-tests for data analysis. In terms of results, the KNN algorithm outpaces FKM by delivering an 89.5% mean accuracy rate, compared to FKM's 81.6%. This quantifiable edge suggests that KNN provides a more accurate and thus, more reliable means for converting text images into spoken language. The paper advocates for the adoption of the KNN algorithm in applications aimed at converting visual text to spoken language. Its higher accuracy rate makes it a more dependable choice, thereby pushing the field of assistive technology one step closer to creating reliable solutions for those in need.

9 References

[1] Jie Mei, Aminul Islam, Yajing Wu, Abidalrahman Moh'd, Evangelos E. Milios, Faculty of Computer Science, Dalhousie University: Statistical Learning for OCR Text Correction

[2] S. Srivastava, A. Verma and S. Sharma, "Optical Character Recognition Techniques: A Review," 2022 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), BHOPAL, India, 2022, pp. 1-6, doi: 10.1109/SCEECS54111.2022.9740911.

[3] M. P. Babu and A. G, "OCR Based Image Text to Speech Conversion using K-Nearest Neighbors and Comparing with Fuzzy K-Means Clustering Algorithm," 2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 2023, pp. 1-5, doi: 10.1109/ACCAI58221.2023.10200864.

[4] S. K. Henge and B. Rama, "Comprative study with analysis of OCR algorithms and invention analysis of character recognition approched methodologies," 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India, 2016, pp. 1-6, doi: 10.1109/ICPEICES.2016.7853643.