ENGR 418 PROJECT REPORT

School of Engineering
Faculty of Applied Sciences
University of British Columbia

Project Title: Sorting using Raw Images
Group No.: 20
Members: Jesse Alele (82807728), Zach Kelly (41637836)
Date: November 13th, 2022

Introduction

The problem considers sorting of lego blocks using raw image data. The process of sorting involves feature differentiation and object classification. The images of each lego block are differentiated based on the features (pixels) and classified by assigning a label to the corresponding image. How well the machine sorts the lego blocks depends on the classification model and dataset used. In this report, a logistic regression model is implemented on a dataset containing circular, rectangular, and square lego blocks of known labels, as shown below in figure 1. The objective is to describe the algorithm used and model performance on the training and testing datasets.

Figure 1: class labels of the dataset.

Theory

To correctly solve the problem at hand, a classification algorithm involving a logistic regression model that learns the relationship between inputs and their outputs from data was developed to sort multiple images. The images were sorted according to classes rectangle, circle or square depending on the arrangement of pixels within them. A classification algorithm worked well for this problem since the data used contained discrete labels.

Developing the classification algorithm involved obtaining the x_train and y_train datasets from a training folder through a "get_data" function. This function returned x and y values. Within the "get_data" function, a for loop nested with conditional statements was used to assign classes 0, 1 or 2 to the image files depending on their filenames. Numerical classes were chosen because they could easily be used within certain functions. This process enabled us to obtain a column vector of classes that would be returned as y_train. Additionally, the images were processed through conversion to grayscale and resizing to the same length and width. These were done in an attempt to localize the pixels needed to differentiate the images.

Pixels within the images were then stacked as a single column and later transposed through a reshape function to obtain features in each row of the matrix that would similarly be returned as x_train. Once the x_train and y_train datasets were obtained, a logistic regression model was trained using the training data. The model's performance on the training data was also obtained. A process similar to the one within the "get_data" function was then followed when defining the "test_function" function. This function was then similarly used to obtain x_test and y_test datasets from the testing folder and return the model's performance on the testing data.

Algorithm

The algorithm consists of extracting the features and classes from both training and testing datasets, training a classification model and testing the model's performance on both training and testing datasets.

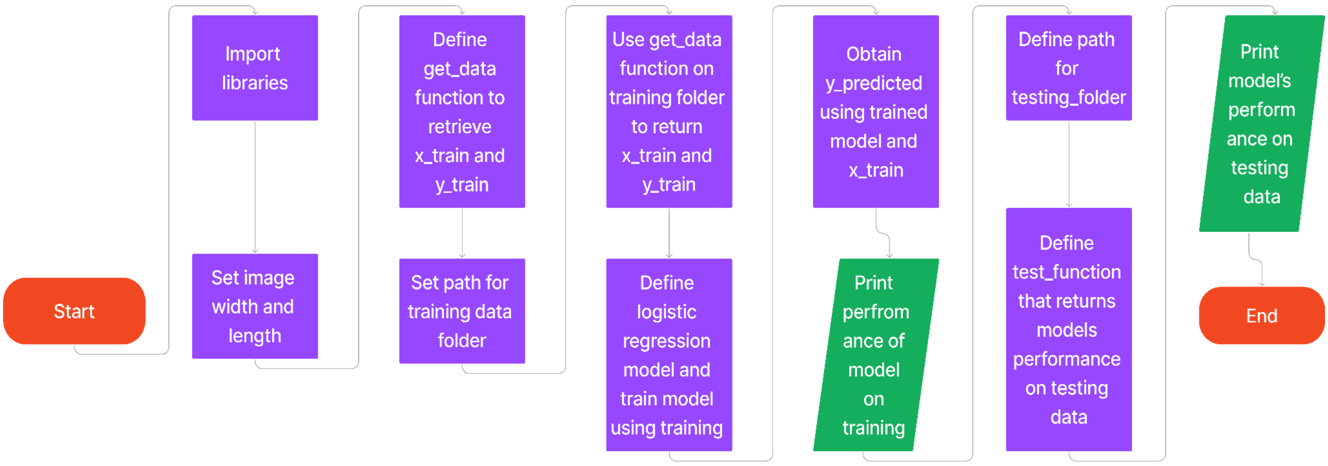

Figure 2: Flowchart describing the classification algorithm.

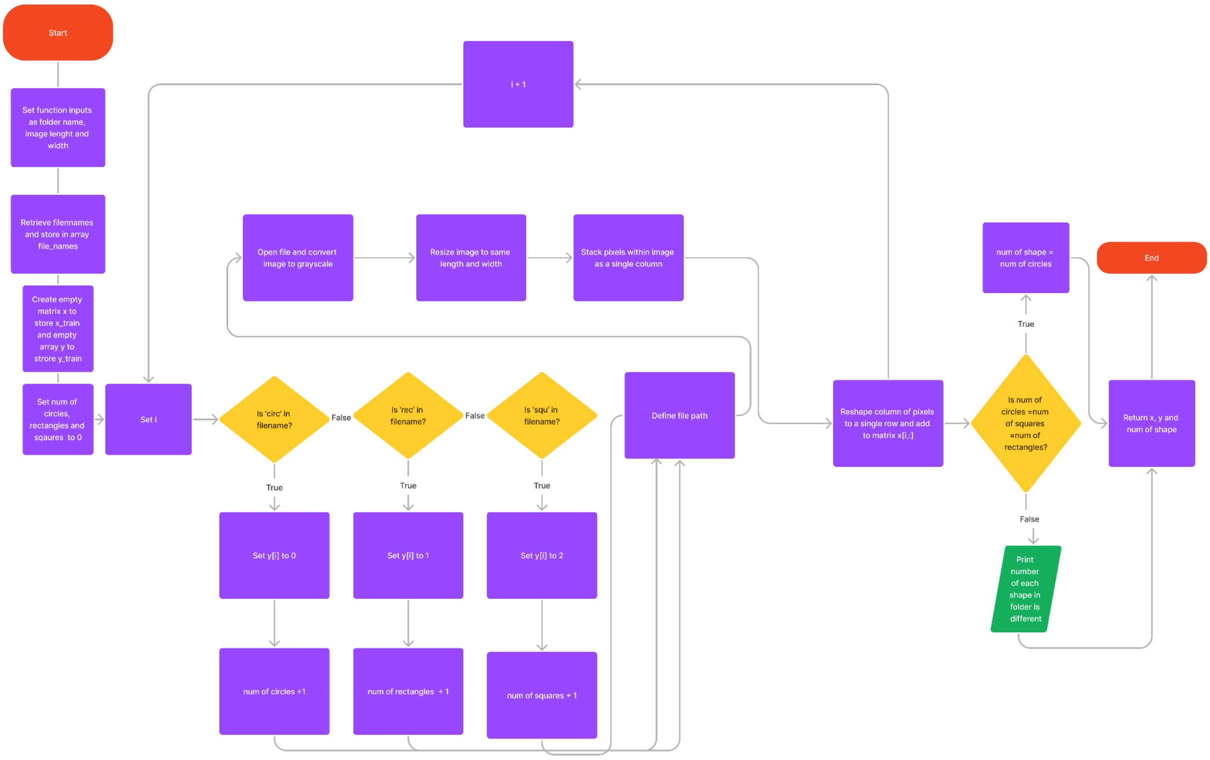

Figure 3: Flowchart describing the get_data function algorithm.

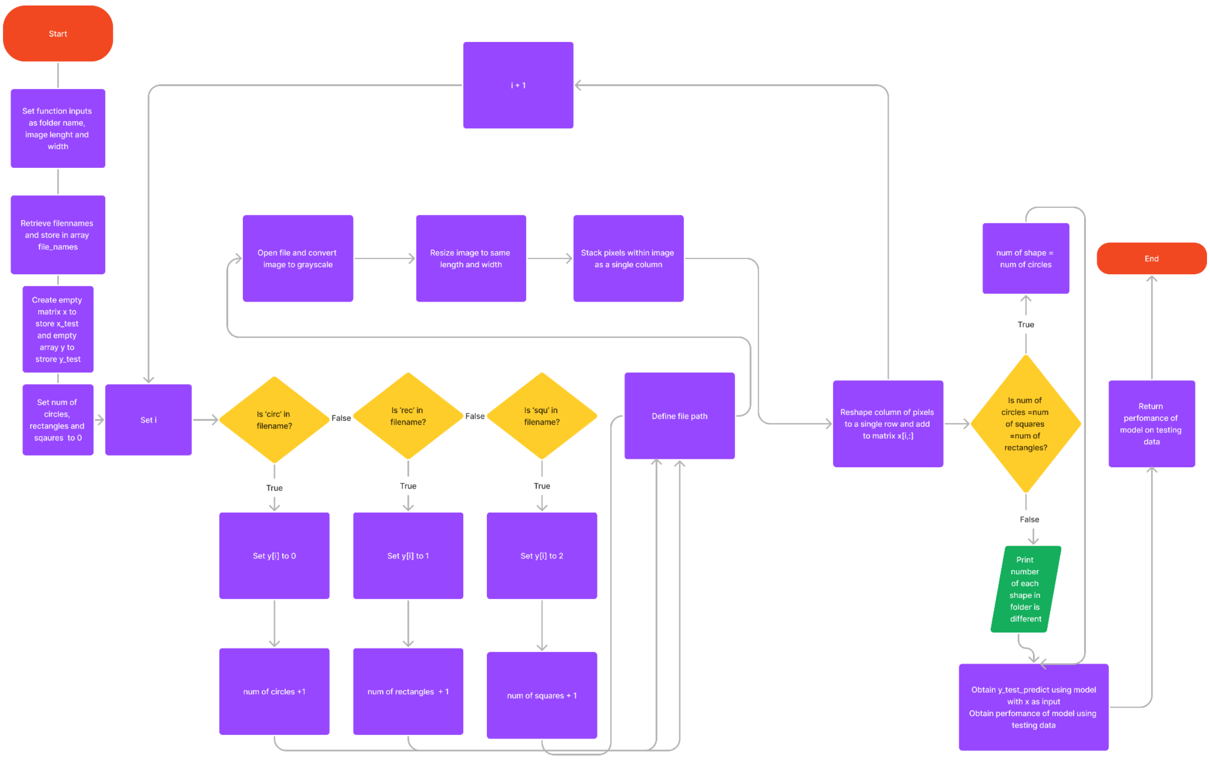

Figure 4: Flowchart describing the test_function function algorithm.

Note: the test_function and the get_data functions make the assumption that the circular, rectangular and square shapes are not mixed with respect to order and that there exists an equal count of circles, rectangles and squares in the selected folder.

Results and Discussion

The confusion matrix of the model predicted classes versus the known classes of the training data is shown below in figure 5.

```
[[36  0  0]
 [ 0 36  0]
 [ 0  0 36]]
```

Figure 5: confusion matrix of classification model on training data (image size: 63x63 pixels).

Figure 5 suggests that all 108 images were classified correctly and there are a total of 36 circles, rectangles and squares in the training dataset. In other words, the classification model has 100% accuracy. This agrees with our expectations, hence the model was trained on the training dataset.

The confusion matrix of the model predicted classes versus the known classes of the testing data is shown below in figure 6.

```
[[17  0  1]
 [ 0 17  1]
 [ 1  0 17]]
```

Figure 6: confusion matrix of classification model on testing data (image size: 63x63 pixels).

Figure 6 suggests that there were 3 misclassified shapes in the testing dataset. In fact, one misclassification for each shape. This agrees with the accuracy scores calculated below in figure 7.

```
The accuracy score on the circles is: 94.44%
The accuracy score on the rectangles is: 94.44%
The accuracy score on the squares is: 94.44%
The accuracy score on the entire testing dataset is: 94.44%
```

Figure 7: summary of the accuracy scores.

According to figure 7, the classification model has an equal success rate in predicting circle, rectangle and square class outcomes in the testing dataset at 63x63 pixels.

The image size was varied within the range of 0 to 4067 to further investigate the model performance. The results suggested that the model performed best when the image size was between 625 and 1600 pixels. The following performance characteristics were obtained in figure 8 for an image size of 900 pixels.

```
The confusion matrix is:
 [[17  0  1]
 [ 0 18  0]
 [ 1  0 17]]
The accuracy score on the circles is: 94.44%
The accuracy score on the rectangles is: 100.00%
The accuracy score on the squares is: 94.44%
The accuracy score on the entire testing dataset is: 96.30%
```

Figure 8: performance characteristics of the classification model (image size: 30x30).

Based on varying the image size parameter and computing the accuracy scores, the results suggested that the model was most accurate in classifying the rectangles and least accurate in classifying the circles.

Further investigation of the raw image dataset revealed shadowing produced by the lego blocks. This attributes to the errors in classification because the darker shades represent a different grayscale value than the background colour. The grayscale value of the shadow, in a sense, smears the boundaries of the lego block making the shape harder to discern. If the value of the pixels defining the shadow is similar to the value of the boundary pixels, then the pixels may be misunderstood by the model as pixels defining the lego block.

Conclusion

A classification model using logistic regression was trained on raw image data to classify circular, rectangular, and square-shaped lego blocks. The model was trained on labelled data. To assess the model performance, a testing set of 54 images was used and the image size was varied between 0 and 4067 pixels. At an image size of 900 pixels, the model successfully classified 94.44% of the circular lego, 100.00% of the rectangular lego and 94.44% of the square lego. The total accuracy of classifying the lego blocks was 96.30%. This model's accuracy meets an acceptable level, however, improvements may be made via processing of the raw images to reduce the influence of the shadowing.