Fresh and Rotten Fruit Classifiers

Jason Yu

Abstract

An important quality assurance process in the agriculture industry involves the classification of fresh and rotten produce. This paper explores use of an SVC classifier, pre-trained models, and a custom CNN model to solve the binary classification problem of fresh and rotten fruit. Specifically, the fruits focused on this paper are apples, oranges, and bananas. Experiments show ultimately the custom CNN model performed the best.

1. Introduction

A common application of pattern recognition and image processing is produce (fruits and vegetables) classification. A similar yet equally important problem is the classification between fresh and rotten produce. This classification is vital for stakeholders across the supply chain from suppliers to consumers at the grocery store. The challenge of this problem can be attributed to the variability of the decomposition and its visual impact on produce. Some produce may show distinct decomposition features such as discoloration, texture change, and mold growth, while others may be less detectable, especially through only an image. Often, separating fresh from rotten produce can be a manual process with quality assurance workers making this classification on a conveyer belt line. The goal of this project is to explore classifiers and CNN models for a certain group of produce (apples, oranges, and bananas) to classify images as fresh or rotten

2. Dataset

The dataset used to train and test the models in this project is sourced from Kaggle (Kalluri, 2018). The dataset consists of six classes: a fresh and rotten class for apples, bananas, and oranges. As shown in in Figure 1, some of the images in the dataset are transformed by various augmentations such as rotations and cropping. Specifications of the images include 24-bit depth, 3 channels (RGB), and various sizes with the smallest size being (144, 122) in the training dataset and (144, 176) in the test dataset. The original dataset included separate train and test images which were used accordingly in this project. The train and test dataset consisted of 10,901 and 2,698 images, respectively, across the six classes.

Figure 1. Sample images of fresh and rotten classes in dataset

3. Modelling the Data

3.1. Approach 1

To determine an appropriate classification technique, an approach to visualize and plot the data was required. Two main challenges included the different sizing in images and the dimensionality reduction problem. The different image sizing was resolved with a reshape transformation to (128, 128). As a result, all images were of equal size and square, which will be beneficial for data manipulation such as convolution. The downside of a simple reshape is that images become distorted and causes loss of information in terms of shape of the fruit. After reshaping, the shape of the images is as follows (128, 128, 3). If the goal is to plot each image on a grid, each image would need to be reduced to two components (two-dimensional plot) or three components for (three-dimensional plot). A common technique for dimensionality reduction is principal component analysis (PCA) which approximates the data by projecting the data on k-main axes for k-components. Intuitively, each image could be flattened to one dimension for a new shape

. **AUTHORERR: Missing \icmlcorrespondingauthor.**

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

of (49152,) and normalized. PCA could then be applied to reduce the shape to two components for a new shape of (2,).

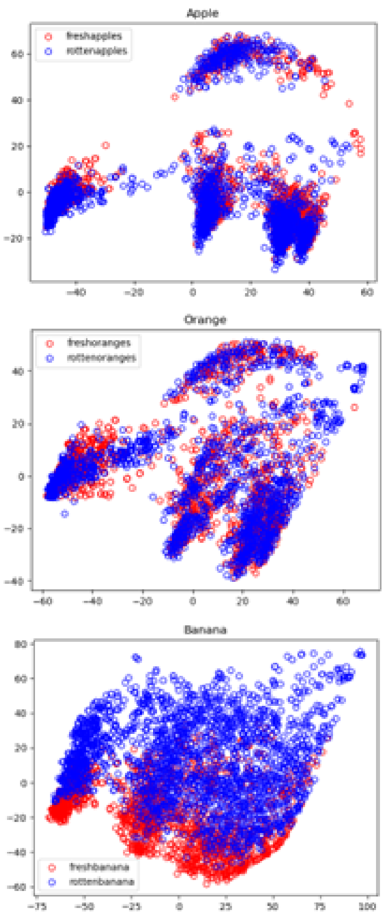


Figure 2. Train dataset plotted on 2-d plot using flatten and PCA technique

As shown in Figure 2, it is clear this technique yields poor distribution between fresh and rotten images across all fruits. Most of the data points overlap each other and it is difficult to make a distinction between fresh and rotten points. Applying any distance-based or probabilistic classifier would be ineffective due to the lack of separation between fresh and rotten classes. An attempt to project the image data on a three-dimensional grid using PCA was also made, as shown in Figure 3. However, the results were similarly ineffective for the same overlapping reasons as the two-dimensional plot. Nonetheless, an interesting observation can be made comparing Figure 2 and Figure 3, in which the 2-d plot is top-view of the 3-d plot, thus demonstrating PCA's projection onto a third dimension.

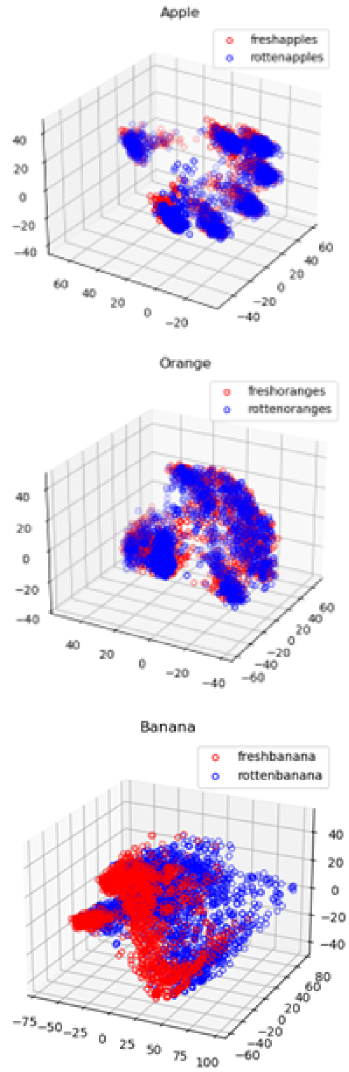


Figure 3. Train dataset plotted on 3-d plot using flatten and PCA technique

The overlapping nature between fresh and rotten fruits is likely attributed to the oversimplified transformations by flattening all the data into a single one-dimension array and applying global normalization. As a result, key spatial and color information that would differentiate the two images is completely lost.

3.2. Approach 2

A second approach to modelling the data consisted of using the three channels (RGB) as the basis of the plot. As shown in Figure 4, each RGB image can be separated into its channel components for the red, blue, and green spectrum.

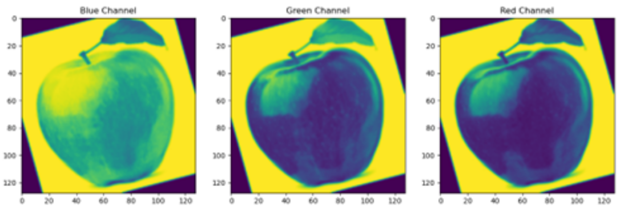


Figure 4. Sample image of a fresh apple split into its RGB channels

The reshaped image of (128, 128, 3) shape can be separated into three matrices of size (128, 128). Similar to the first approach, each channel matrix can be flattened and fed through PCA to reduce its dimensionality to one component. As a result, each image consists of three components, one for each RGB channel. The result of this process is shown in Figure 5.

For apple and orange images, the number of clusters in which the data was distributed reduced to three. However, within each cluster, fresh and rotten images were still significantly overlapping, This poses as a difficult problem for most classifiers that rely on the two classes to be separated. On the other hand, bananas yielded a distinct separation between fresh and rotten classes. Based on the separation behavior, it can be concluded that the red and blue channel data of fresh and rotten bananas are distinctively different. Since this plot is three-dimensional, a decision boundary in the form of a hyper plane can be determined.

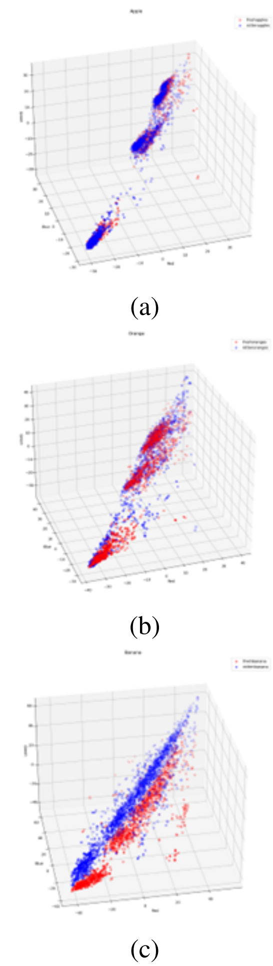
To define this hyperplane that separates fresh and rotten data points, a support vector classification (SVC) classifier will be deployed. Equation 1 outlines the objective function of a support vector machine (SVM) classifier (generalized SVC).

$$min_w \lambda ||w||^2 + \sum_{n=1}^{n} (1 - y_i < x_i, w >)_+ \tag{1}$$

The SVM algorithm aims to separate the two classes using the best fit which maximizes the distance between the closest data points to the decision boundary of each class. These distances are known as support vectors. Applying an SVC classifier with a linear kernel yields the decision boundary equation in Equation 2 and is visualized in Figure 6.

$$0.1529 - 0.3586x - 0.1745y + 0.553z = 0 \tag{2}$$

Testing this classifier on the test dataset yielded an accuracy of 92.87% with 846 errors.



(a)

(b)

(c)

Figure 5. Train dataset plotted on 3-d plot using RGB channel as the basis (a) apple (b) orange (c) banana

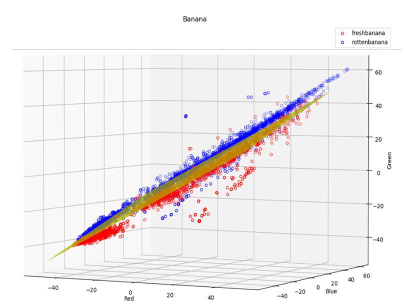


Figure 6. Banana training dataset decision boundary using SVC

4. CNN

4.1. Pre-trained Models

Although the SVC classifier is proven to be effective, it is limited to only the banana dataset. To build a more robust solution, a convolution neural network (CNN) was explored next. Rather than approaching the problem as a multi-label classification problem with six classes, the proposed CNN will solve the binary classification problem between fresh and rotten classes. Preliminary testing was performed using known models such as Resnet50, EfficientNetB0, and MobileNetV2 with pre-trained weights. Given the pre-trained weights, transfer learning was applied in which all layers after removing the top layer were frozen. A fully connected (FC) layer was implemented as the new top layer to satisfy the binary classification. The results of training each model for 10 epochs are shown in Table 1.

Table 1. Training and test accuracies of training known models

Model	Layers	Training Accuracy (%)	Test Accuracy (%)
Resnet-50	50	98.92	73.68
EfficientNetB0	237	98.28	74.05
MobileNetV2	53	87.45	58.19

Using pre-trained models demonstrated overfitting with mediocre test accuracy despite high training accuracy. This can be attributed to each model consisting of a high number of convolutions layers. For a relatively simple binary classification problem, too many layers cause an abundance of features to be extracted which can easily lead to overfitting of the training dataset.

4.2. Custom CNN

As an alternative to pre-trained models, a custom CNN was designed with fewer convolution layers to match the complexity of the binary classification problem. The architecture of this custom CNN consists of four convolution layers, a fully connected layer, and a sigmoid classifier as shown in Figure 7.

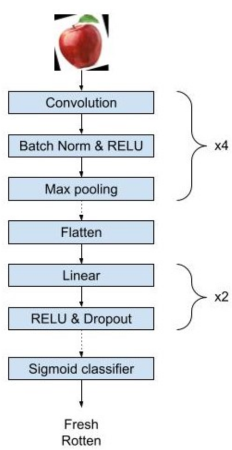
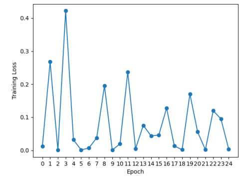
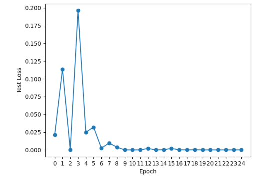


Figure 7. Custom CNN architecture

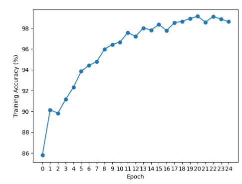


(a) Training Loss

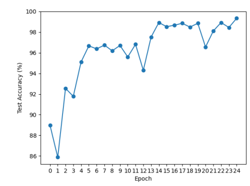


(b) Test Loss

Figure 8. Training and test loss of custom CNN model



(a) Training Accuracy



(b) Test Accuracy

Figure 9. Training and test accuracies of custom CNN model

Training this model over 25 epochs demonstrated exceptional results, yielding training and test loss shown in Figure 8 and training and test accuracy shown in Figure 9.

5. Application

With a trained model, we can now apply it to a practical use case. An application dataset was built and consisted of several images documenting a timelapse for each fruit as it decayed from fresh to rotten [3]. As expected, the model predicted earlier images as fresh, while later images as rotten. Figure 10 displays the images of each fruit that was first classified as rotten by the model.

(a)

(b)

(c)

Figure 10. Images of timelapse that were first classified as rotten by the model (a) apple at 224 days (b) orange at 18 days (c) banana at 13 days

6. Conclusion

Overall, we explored the usage of SVC classifier, pre-trained CNN models, and a custom CNN model to solve the binary classification problem of fresh and rotten fruits. We were unable to use traditional distance-based or probabilistic classifier due to the poor distribution of data once plotted on a grid. However, one technique of processing the data allowed us to use an SVC classifier on the banana dataset which yielded a test accuracy of 92.87%. When exploring pre-trained models, it was discovered that the excess number of convolution layers caused overfitting and poor test accuracy. Finally, a custom CNN was built which proved to be robust for our use case yielding a test accuracy of 98.56%.

References

Kalluri, S. R. Fruits fresh and rotten for classification. https://www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification, 2018. Version 1. Accessed: 2022-11-28.