Journal of Engineering Sciences
Vol 11, Issue 6,June/2020
ISSN NO:0377-9254

Color Recognition Algorithm using a Neural Network Model in Determining the Ripeness of a Banana

C. Hari Priyanka[1], V. Shikha Rachel[2], B. Harshith[3] and R. Moulisha[4]

[1]*Assistant professor, CSE, BE, Matrusri engineering college, Saidabad, Telangana State, India.*

[2]*Student, CSE, BE, Matrusri engineering college, Saidabad, TS, India.*

[3]*Student, CSE, BE, Matrusri engineering college, Saidabad, TS, India.*

[4]*Student, CSE, BE, Matrusri engineering college, Saidabad, TS, India*

***Abstract*— The quality of fresh banana fruit is a concern for consumers and fruit industrial companies. The effectiveness and fast classification of banana's maturity stage are the most decisive factors in determining its quality. It is necessary to design and implement image processing tools for correct ripening stage classification of fresh incoming banana bunches. This paper presents a simple color recognition algorithm using a Convolutional Neural Network model and applied to determine the ripeness of a banana based on the colour of the banana in the picture and determine the ripeness namely :raw, ripe and overripe.**

***Keywords*— color recognition, banana, CNN, overripe, ripe, under ripe, classification, ripeness, Neural Network.**

1. INTRODUCTION

The normal human eyes have three types of sensors and the signal of these three sensors determine the color response of the observer. The response of this system produces the three-dimensional phenomenon of three dimensional spaces. From this theory, the color recognition model has been applied widely in industrial sectors, commercial fields as well as in social responsibilities. Normally, the weighing system is used for cashiers to determine the prices of the fruits or vegetables. However this color recognition system can substitute the weighing system and it is capable to recognize the type of fruits or vegetables based on the color of the scanned image. Therefore, in this research, a simple algorithm method using the CNN model is proposed to determine the ripeness color of a fruit and the sample of the fruit used is banana.

2. LITERATURE SURVEY

The old method to detect the ripeness of banana is using chemicals in order to obtain the characteristic of the fruit. This method will harm the fruit and also affects its quality. There are also other techniques used after the growth in digital world. The very previous method used to recognize the banana is done using MatLab. A simple graphical user interface system is developed in MATLAB that classifies the ripeness of the banana. This model has an accuracy of 76%. Lastly, it is a big disadvantage for the customer who has a lack of knowledge of the banana to buy one. This is because the customer is not able to pick the right fruit based on their desires. There are some of the fruit condition which needs to wait long until it can be eaten, and some fruit can be eaten raw by cooking it into any delicacies.

In order to avoid this scenario, the proposed system is to determine the ripeness of banana using neural network techniques coded in python.

3. SYSTEM ANALYSIS

3.1 **TensorFlow**

The most famous deep learning library in the world is Google's TensorFlow. Google product uses machine learning in all of its products to improve the search engine, translation, image captioning or recommendations. They build a framework called **Tensor flow** to let researchers and developers work together on an AI model.

Tensorflow architecture works in three parts:

- Preprocessing the data
- Build the model
- Train and predict the model
- It is called Tensorflow because

It takes input as a multi-dimensional array, also known as **tensors**. You can construct a sort of **flowchart** of operations (called a Graph) that you want to perform on that input. The input goes in at one end, and then it flows through this system of multiple operations and comes out the other end as output.

3.2 **Keras**

KERAS is an Open Source Neural Network library written in Python that runs on top of Theano or Tensorflow. It is designed to be modular, fast and easy to use. Keras is high-level API wrapper for the low-level API, capable of running on top of TensorFlow, CNTK, or Theano. Keras also compiles our model with loss and optimizer functions, training process with fit function.

3.3 **Tkinter**

Tkinter is actually an inbuilt **Python** module used to create simple **GUI** apps. It is the most commonly used module for **GUI** apps in the **Python**.

We don't need to worry about installation of the **Tkinter** module as it comes with **Python** default. We can create our own GUIs.

3.4 **System Architecture**

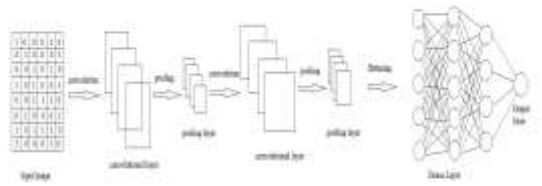

Fig 3.1

4. REQUIREMENTS SPECIFICATION

Python

what is Python?

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

It supports functional and structured programming methods as well as OOP.

It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking.

IT supports automatic garbage collection.

What Is A Graphical User Interface (GUI)?

Graphical User Interface (GUI) is nothing but a desktop application which helps you to interact with the computers. They are used to perform different tasks in the desktops, laptops and other electronic devices.

GUI apps like **Text-Editors** are used to create, read, update and delete different types of files.

GUI apps like **Google Chrome, Firefox and Microsoft Edge** are used to browse through the **Internet**.

4. IMPLEMENTATION

Deep Learning is becoming a very popular subset of machine learning due to its high level of performance across many types of data. A great way to use deep learning to classify images is to build a convolutional neural network (CNN). The Keras library in Python makes it pretty simple to build a CNN. Computers see images using pixels. Pixels in images are usually related. For example, a certain group of pixels may signify an edge in an image or some other pattern. Convolutions use this to help identify images. In simple word what CNN does is, it extract the feature of image and convert it into lower dimension without losing its characteristics.

A convolution multiplies a matrix of pixels with a filter matrix or 'kernel' and sums up the multiplication values. Then the convolution slides over to the next pixel and repeats the same process until all the image pixels have been covered.

The ImagedataGenerator dataset is conveniently provided to us as part of the Keras library, so we can easily load the dataset. We created 2 types of datasets used for traning and testing. One is to recognize the ripeness of banana which is called

banana dataset and the other is to check whether the given item is banana or not which is non banana dataset.

Next, we need to reshape our dataset inputs (train_X and test_X) to the shape that our model expects when we train the model. The first number is the number of images (200 for train_X and 100 for test_X). Then comes the shape of each image (28x28). The last number is 1, which signifies that the images are greyscale.

The model type that we will be using is Sequential. Sequential is the easiest way to build a model in Keras. It allows you to build a model layer by layer. We use the 'add()' function to add layers to our model. Our first 2 layers are Conv2D layers. These are convolution layers that will deal with our input images, which are seen as 2-dimensional matrices.

Activation is the activation function for the layer. The activation function we will be using for our first 2 layers is the ReLU, or Rectified Linear Activation. This activation function has been proven to work well in neural networks.

In between the Conv2D layers and the dense layer, there is a 'Flatten' layer. Flatten serves as a connection between the convolution and dense layers.

'Dense' is the layer type we will use in for our output layer. We will have 10 nodes in our output layer, one for each possible outcome (0–9).

The activation is 'softmax'. Softmax makes the output sum up to 1 so the output can be interpreted as probabilities. The model will then make its prediction based on which option has the highest probability.

Next, we need to compile our model. Compiling the model takes three parameters: optimizer, loss and metrics.

The optimizer controls the learning rate. We will be using 'adam' as our optmizer. We will use 'categorical_crossentropy' for our loss function. To make things even easier to interpret, we will use the 'accuracy' metric to see the accuracy score on the training set when we train the model.

Now we will train our model. To train, we will use the 'fit()' function on our model with the following parameters: training data, validation data, and the number of epochs. The number of epochs is the number of times the model will cycle through the data. The more epochs we run, the more the model will improve. For our model, we will set the number of epochs to 3.

5. EXPERIMENTAL RESUTS

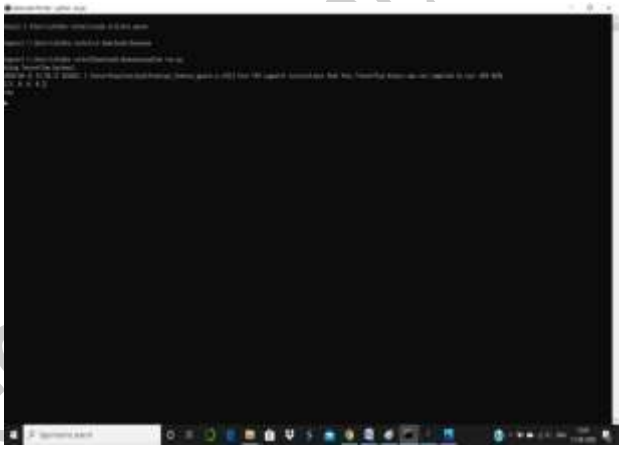

Fig 5.0

Fig 5.1 Imput Image

Fig 5.2 Output of given image

Fig 5.3 Input Image

Fig 5.4 Output of given image

6. CONCLUSION

We developed a system for the purpose of recognizing the ripeness of banana. The system has three main stages ; preprocessing, training and ripeness classification. The work was accomplished by training a set of input data. A Convolutional Neural Network model is developed for color recognition. After preprocessing , the model is trained with the dataset. The images are resized to the required size and undergo through two layers of convolution and max pooling. The model is trained with the dataset passing through several layers and by using backtracking to ensure the model is correctly trained until atmost accuracy is obtained . From the experimental results, the simulations show that the ripeness recognition rate is more than 93%.

In future, this method can be applied in increasing the effectiveness for cashiers and customers in counters when determining the prices of fruits or vegetables instead of the conventional weighing method as this method is simpler and will save time CNN approach. There is a possibility for enhancements for this project for detecting the morphed images to the original images using CNN

7. REFERENCES

1. J. Hongjiao, "Application of Advanced BP Neural Network in Image Recognition," 2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES), Wuhan, China, 2019.
2. Paulraj M P, Hema C R, Pranesh Krishnan, School of mechatronics engineering, University Malaysia Perlis, Electronics and Electrical Engineering Communication, University of Sheiffeld, England.
3. S. J. Lee and S. W. Kim, "Recognition of Slab Identification Numbers Using a Deep Convolutional Neural Network," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, 2016.
4. D. M. Filatov, K. V. Ignatiev and E. V. Serykh, "Neural network system of traffic signs recognition," 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), St. Petersburg, 2017.
5. S. Jung, U. Lee, J. Jung and D. H. Shim, "Real-time Traffic Sign Recognition system with deep convolutional neural network," 2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Xi'an, 2016.
6. B. Jiang, X. Li, L. Yin, W. Yue and S. Wang, "Object Recognition in Remote Sensing Images Using Combined Deep Features," 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019.
7. J. Bai, Z. Chen, B. Feng and B. Xu, "Image character recognition using deep convolutional neural network learned from different languages," 2014 IEEE International Conference on Image Processing (ICIP), Paris, 2014.
8. A. Kausar, M. Sharif, J. Park and D. R. Shin, "Pure-CNN: A Framework for Fruit Images Classification," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018.

Journal of Engineering Sciences

9. G. Zeng, "Fruit and vegetables classification system using image saliency and convolutional neural network," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, 2017.
10. Z. M. Khaing, Y. Naung and P. H. Htut, "Development of control system for fruit classification based on convolutional neural network,"

Journal of Engineering Sciences