Grenze International Journal of Engineering and Technology, January Issue

Predicting the Stages of Banana Ripeness by CNN

Ganeshan R[1], Raju T[2], Thiyagu T[3], Tamilselvan T[4], Pragadeeswaran Sundaresan[5] and Anitha.P[6]
[1,3-6]School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology
Avadi, Chennai, India.
[2]Department of CSE (Cyber Security), CMR College of Engineering & Technology, Kandlakoya, Medchal, Hyderabad, India
Email: ganeshramasamy111@gmail.com, t.raju@cmrcet.ac.in, thiyagu.57@gmail.com, tamilselvan.akash@gmail.com, pragadeesresearch@gmail.com, anitapmca@gmail.com

***Abstract*— Bananas are one of the most popular fruits in the world. Bananas are a terrific source of energy; they fill you up. In any sport, bananas provide you more energy. They are a nutritious source of potassium, vitamin B6, vitamin C, and fiber. There are many different kinds and sizes, and their colors span from green to yellow. One of the cheapest fresh fruits is this one. The ripeness of banana fruit usually determines how good it tastes and how much it costs on the market. This research aims to forecast the ripening stages of bananas, including unripe, ripe, and overripe classifications. The classification of the banana would not have been achievable with perfect accuracy and precision without the invention of the convolutional neural network.**

***Index Terms*— Energy, Booster, Vitamin, Fiber, Types, Quality, Ripening, Dataset.**

I. INTRODUCTION

Eating fruit is good for your health. People who consume a balanced diet that includes more fruits and vegetables are less likely to get sick. But it's not enough to just eat fruits; the fruits we eat directly or indirectly should also be fresh. Freshness is one of the most important signs of good meals. So, we really need a way to predict fruit.

II. RELATED WORKS

Fruit recognition can be difficult since the stacked fruits on a scale are both complicated and similar. This paper, "Fruit Classification and Detection Using Deep CNN," proposes a methodology for recognizing fruit using CNN. The shape and shading of each photo tell it apart. A three-stage model of image recognition is proposed: preparation, extraction, and arrangement [1]. This research, "Classification of Ripening of Banana Fruit Using Convolutional Neural Networks," use convolutional neural networks to ascertain the ripening phases of bananas, utilizing historical relational data to develop a model that elucidates the determination of these stages. We also learned about data-driven methods [2]. This article, "Monitoring The Change Process Of Banana Freshness By GoogleNet," uses the GoogLeNet network to figure out how fresh bananas are. By rotating and translating the data, the model's capacity to generalize is improved and it is less likely to over-fit. This paper has improved our understanding of how to extract and separate features [3]. The paper "Banana Ripeness Detection Using CNN and Flask" talks about the requirement to prepare data in a short way. It is very important to preprocess the data before using it since Machine Learning and Deep Neural Networks have very specific input needs. You can add to data in a number of ways, such as by cropping, padding, flipping, zooming, and rescaling [4].

Grenze ID: 01.GIJET.12.1.531_1

This research, "Ripeness Classification of Banana Using An Artificial Neural Network," proposes an automated computer vision system to recognize the ripening phases of bananas. To start, a four-class handmade database is built. Second, an artificial neural network-based framework that utilizes color, the emergence of brown spots, and Tamura statistical texture features is applied to identify and assess the ripening stage of banana fruit. We compare the results and performance of the proposed system to other methods like the SVM, the naive Bayes, and the KNN. [5].This study, “Deep Indicator For Fine Grained Classification of Banana Ripening Stage’s,” presents CNN models for the initial time in the context of banana ripening prediction. It puts up a new CNN architecture that has been altered to work with fine-grained feature representations for a more precise classification of banana ripening phases. Most of the common ways to anticipate when bananas will ripen are only good for rough classification of maturity. [6].

The aim of this research was to develop a grading algorithm utilizing a fusion of image processing and machine learning methodologies, specifically neural networks (ANN) and support vector machines (SVM), to classify peeled pistachio kernels (PPKs) into five distinct categories: green, yellowish green, yellow, mixed color, and waste materials [7].

It has done a lot of research on bananas to find out what their different biological traits are. To properly classify fruit, you need to know a lot about it. The most important things to look at are color and shape. This work "Prediction of Banana Quality Attributes and Ripeness Classification Using Artificial anana [8].

II. Materials and Methods

A. Dataset

Every neural network only works on categorization depending on the input. Input is the information we use to teach the network. Once the model is trained, we send it just one input to predict the output again. The input goes to the neural network's input layer for processing.This paper is on how to tell the difference between ripe and unripe bananas, so we needed to collect photographs of bananas at different stages of ripeness. This model needs images as its main input. We could only guess what stage the banana was at after the neural network processed each image. The total data set contains of 54 images in which 30 percent, that is 16 images are test images and 38 images are Train images. So our neural network is trained with the 38 images.This report presents data obtained at various stages of ripening. We have divided the ripening stages of bananas into three groups: green (not ripe), ripe, and overripe.

So, as was said before, the three types of ripening photos were gathered. These three groups of pictures made up the DATASET. We don't train the neural network on the whole dataset since we want it to be accurate and precise. We need to split our dataset into training and testing data in a way that makes sense. To start, we train the model with the training data. Then, we test or validate the model by giving it the test data as input. This lets us accurately forecast how well the model will work. This is Fig. 2.1, which shows how the directory is set up.

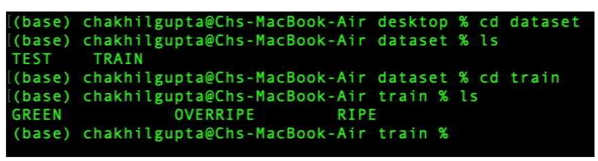

Figure 1. Dataset Directory

The dataset directory is split into two parts: "TEST" and "TRAIN," as seen in the picture above. The TRAIN is once more split into the "GREEN, OVERRIPE, AND RIPE" subdirectory. All the photographs of the ripening stage are in the TEST folder. We have split the data into groups of 7 and 3.

This Fig.2.2 shows data consisting of sixteen different images of bananas of different ripening stages.

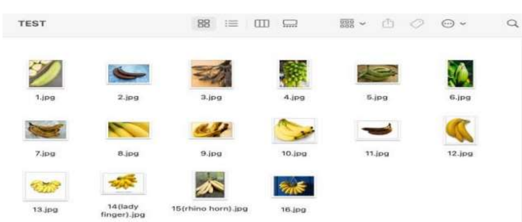

Figure 2 .TEST Images This Fig.3 shows the thirteen images of the bananas taken when they were unripe

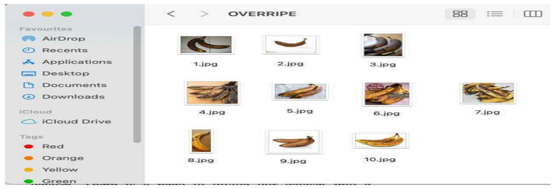

Figure 4. OverRipe training examples

This Fig.2.4 shows the ten overripe bananas collected as images for training data.
This Fig.2.5 shows the fifteen ripe sample bananas collected as training data for the neural network.

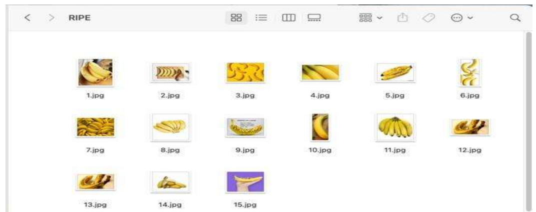

Figure 5. Ripe training examples

The model gets better at predicting the ripeness of any test image given to it as it learns from these diverse training examples.

B. Sequential Model

When you need to stack layers on top of each other, a Sequential Model is the best choice. We may say that each layer only has one input and one output. This model is made up of Keras, which is an open-source software framework that gives Python an interface for artificial neural networks and an interface for the TensorFlow library. The main use of this model is that in most neural networks, like one of the convolutional neural network layers, the layers are arranged in a sequential order and the data flows from each layer in that order until it reaches the output layer, which is also called the SoftMax layer and has the activation function[9].The sequential approach is straightforward to use, which is its best feature. This Fig.2.6 shows how the neural network is put together.

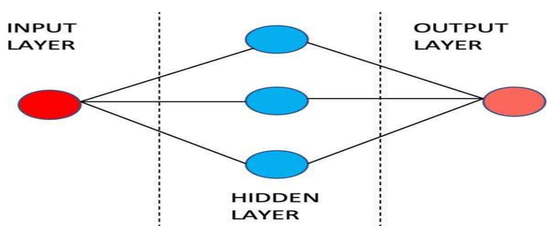

Figure 6. A Sequential Model

Neural networks are algorithms that are made to work like the human brain, which has many layers. Every layer has its own job. This helps to break down a hard problem into smaller steps. The Input layers, Hidden Layers, and Output Layers make up the Neural Network. The Input layer is in charge of getting the data ready for processing. The height, breadth, and pixels of the image should all be the same. This layer is where rotation, translation, and rescaling happen.
The hidden layer changes the input. The hidden layers are made up of the layers that are combined, like convolutional layers, pooling layers, fully connected layers, and normalizing layers. There is always at least one hidden layer in a neural network for processing data. The more hidden layers there are, the longer it takes for the network to give the output.
The output layer takes the results from the hidden layer, changes the values it has collected, and presents the correct answer in a way that makes sense. The output of this layer is the last output. Any neural network must have at least one output layer. This layer gets the input from the layers before it, processes it with the neurons, and then combines the output[10].A sigmoid function helps you get the output between 0 and 1. The activation function is this sigmoid function. The hidden layer has two pieces inside the neuron: the weighted sum of the

input and the activation function. If there were no activation function, the output would just be a linear equation that was a weighted sum of the input feature. Linear equations can't tackle the hard problems. To find patterns in this universe, you need a nonlinear equation, and the activation function helps you build the equation.

To gain a greater accuracy, we have split the training data set into seven parts and three parts. The CNN gets this information. There are different layers in Convolutional Neural Networks, like the convolutional layer, inception layer, pooling layer, fully connected layer, linear layer, and softmax layer. Every layer has its own vital part in the criterion for classification[11].

It is always a Convolutional Layer that makes up the initial layer of a Convolutional Neural Network. Convolutional layers do a convolution operation on the input and send the result to the next layer. A convolution takes all the pixels in its receptive area and turns them into one value. If you apply a convolution to an image, for instance, you will make the image smaller and combine all the information in the field into one pixel. A vector is the last thing that the convolutional layer gives out. We can employ different sorts of convolutions depending on the problem we need to solve and the features we want to learn.

The 2D convolution layer, which is generally called conv2D, is the most frequent type of convolution. In a conv2D layer, a filter or kernel "slides" over the 2D input data and multiplies each element by itself. It will then add together the findings into one output pixel. The kernel will do the same thing at each place it glides over, turning a 2D matrix of features into a separate 2D matrix of features. This is the experimental flow chart in Fig. 2.7.

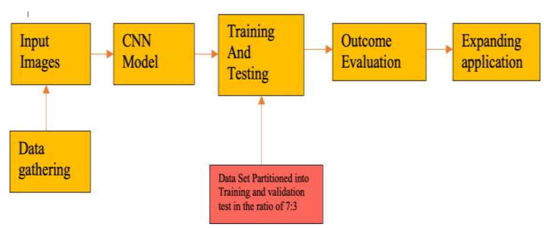

Figure 7. Experimental Flow Chart.

A convolutional layer has a group of filters whose settings need to be learned. The filters are smaller in height and weight than the input volume. To make an activation map made of neurons, each filter is convolved with the input volume[12].

Activation maps are just a way to show the activation values at different levels of the network. We know that the system sees images as pixels. The pixel value might be anywhere from 0 to 255. We can lower these pixels to either zero or one by using a min max scalar . We may find the features with the aid of filters during the convolution procedure[13]. The filters could be used to find any form of edge. This Fig.2.8 shows how the convolution stage works.

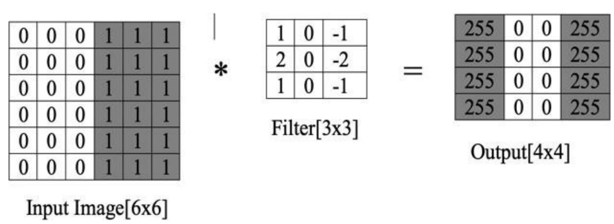

Figure 8: Convolution step by a vertical edge filter

The gap layer is sometimes called global average pooling. We send a Feature Map with the dimensions H, W, and D to the input layer of the gap operation. The operation then finds the average value of each feature map and sends the result to the output node. This helps cut down on the number of dimensions. It also helps to cut down on overfitting. This Fig. 2.9 displays the gap.

Figure 9. Global Average Pooling

Then comes the inception layer, which is one of the most crucial layers of the CNN. The inception layer is made up of the convolution and max pool layers. There were a lot of convolutional layers in the inception module. There are many versions of inception modules, like v1, v2, and v3. The inception v3 contains a 7x7 convolution that is broken down into parts. And it also contains a regularization part to stop overfitting.

A big Convolutional neural network with a lot of depth has a lot of weights and parameters that are biased. The neural network tends to overfit when there are a lot of parameters. Overfitting happens when the network has a lot more data than it needs. So when this happens, the network starts to pick up noise and wrong data in the sample. This leads to inefficiency and inaccuracy in an indirect way.

To use dropout, we choose a dropout ratio that can be any number between 0 and 1. We just choose a small number of the activation functions or hidden layers. And a group of neurons in the input and hidden layers. So, after completely shutting down the undesired neurons, the burden is lower and the overfitting is less.

The SoftMax layer is the last layer of the convolutional neural network. The SoftMax activation function is used by this layer. This layer produces the last output. This layer finds the chance that the output belongs to each class. This is the equation that SoftMax employs to figure out the chance of a class. The exponential of a certain class divided by the total of the exponentials of all classes. This is the SoftMax activation function, which is shown in Equation 2.1.

$$\square(y_i) = \frac{\square^{y_i}}{\sum_k \square^{y_k}}$$

Equation.2.1 Softmax Activation Function.

III. EXPERIMENTAL RESULT

Compared to AlexNet, the Convolutional Neural Networks is far more accurate and precise. We didn't just look at the ripeness of one variety of banana in our experiment; we also looked at other types, like the rhino horned banana and the lady finger banana.

It was very necessary to train the neural network with the training data in order to keep the accuracy high. Training and. Testing is a key idea for getting a greater rate of categorization. An epoch is one full pass of the training dataset through the neural network. The total number of times the model has been trained is called an epoch. All training data is used only once. An iteration is the total number of batches needed to finish one epoch. By splitting the training data into batches and training the model with those batches across N iterations and N epochs, the model becomes fully trained with the data, which improves accuracy and precision. This Fig.3.1 illustrates how many times the epoch has happened.

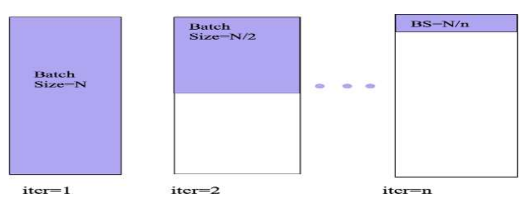

Figure 10. Epoch Iterations

After this era, we were able to get the accuracy up to 82 percent. The model has done a good job of predicting when the banana would mature. The accuracy of the classification has also been kept up. Let's look at how to classify a green banana that isn't ripe yet. This Fig.3.2 shows the model output.

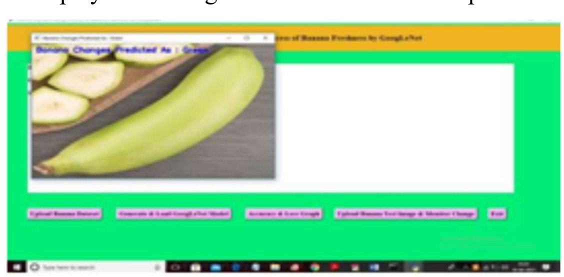

Fig.11: Banana Prediction Output.

V. Conclusion

This project lets both the manufacturers (the users) and the everyday people who buy bananas sort their goods. This fixes a lot of problems, such figuring out how much fruit to store and how much demand there is based on the quantity you have. Our model does a great job at figuring out how fresh bananas are, with a high level of accuracy and a wide range of time periods. Our technique also worked well for sorting a lot of different types of bananas. We want to make this approach more general in the future by including a wider range of natural gifts, such different kinds of fruits and vegetables.

References

[1] Electronics.4u, "Fruit Classification And Quality Detection Using Deep CNN", Global Research and Development Journal,[2021].

[2] Vinay Kumar, "Classification of Ripening of Banana Fruit Using Convolutional Neural Network's", IAET Journal, [2020].

[3] Jiangong, "Monitoring The Change Process Of Banana Freshness By GoogleNet", IEEE, [2020]. [4]. Abhay, "Banana Ripeness Detection Using CNN and Flask",[2020].

[4] Fatma, Ahmed, "Ripeness Classification of Banana Using An Artificial Neural Network", Journal For Science and Engineering, [2019].

[5] Yan Zahng, "Deep Indicator For Fine Grained Classification of Banana Ripening Stage's", EURASIP Journal On Image and Video Processing, [2018].

[6] Mahmoud, "Classification of peeled pistachio kernels using computer vision and colour feature's", Journal: Engineering In Agriculture and Food, [2017].

[7] Adebayo, "Prediction Of Banana Quality Attributes And Ripeness Classification Using Artificial Neural Network", Journal:ICAFE,[2017].

[8] A Parameswari, R Ganeshan, V Ragavi, M Shereesha ," Hybrid rat swarm hunter prey optimization trained deep learning for network intrusion detection using CNN features", Computers & Security, Elsevier Advanced Technology,[2024].

[9] G Nirmala Priya, B Kishore, R Ganeshan, R Cristin ,"Video forgery detection using competitive swarm sun flower optimization algorithm based deep learning", Wireless Networks, Springer US,[2024].

[10] Vadamodula Prasad, R Ganeshan, R Rajeswari ,"Artificial gannet optimization enabled deep convolutional neural network for autism spectrum disorders classification using MRI image". Multimedia Tools and Applications, Springer US,[2024].

[11] R Ganeshan, Satish Muppidi, DR Thirupurasundari, B Santhosh Kumar." Autoregressive-elephant herding optimization based generative adversarial network for copy-move forgery detection with interval type-2 fuzzy clustering", Signal Processing: Image Communication, Elsevier [2022].

[12] Pooja Lalwani, R Ganeshan," A Novel CNN-BiLSTM-GRU Hybrid Deep Learning Model for Human Activity Recognition" ,International Journal of Computational Intelligence Systems, Springer Netherlands,[2024].

[13] Pooja Lalwani, Ganeshan Ramasamy," Human activity recognition using a multi-branched CNN-BiLSTM-BiGRU model", Applied Soft Computing, Elsevier,[2024].