University of Missouri-Kansas City

CS5590/490 - Python and Deep Learning Programming Course

Project: Vegetable Image Classification
Team: NARS

Author(s):	*Role:*
Nikhil Reddy Dumpala	Team Member
Sarath Lella	Team Member
Rahul Kundaram	Team Member
Adishwar Suresh	Team Member

Faculty Advisers: Yugyung Lee , Ahmed Albishri , Saeed Al-Qarni

May 12, 2022

Contents

Abstract

An attempt is made to classify vegetable images accurately. This categorization is based on a dataset of 21,000 photos divided into 15 categories. The most efficient tool in the machine learning field for classification issues is the convolutional neural network, a deep learning technique. However, CNN requires big datasets to perform effectively in natural picture classification issues. Here, we design a CNN model from the ground up to test the efficacy of CNN for vegetable picture categorization.

In addition, the accuracy of multiple pre-trained CNN designs utilizing transfer learning is better compared to that of a standard CNN. A comparison of constructed CNN models and pre-trained CNN architectures is also conducted. We also observed that the transfer learning technique can obtain higher classification results than classic CNN with a small dataset by exploiting past information collected from relevant large-scale work.

1 Introduction

Vegetables are one of the most widely consumed foods in the world. Vegetables are grown by people all around the world. According to a report , there are nearly hundreds of thousands of vegetable species on our planet. Picking and sorting veggies, for example. And distinguishing a vegetable in the market is difficult for the client because different veggies have similar characteristics. To address this problem, vegetable picking, sorting, and labeling must be automated using a vegetable image classifier to save time and money. Fundamental research activity in the field of agriculture nowadays is categorization and detection. There are many different types of veggies, and many people are unfamiliar with them. As a result, the design of a vegetable classifier will also give meaning to people's lives.

In addition, vegetable sorting is done by hand in supermarkets and distribution sites. As a result, this study is being done to address these issues. The goal of this work is to use CNN and pre-trained DCNN to more accurately classify vegetable photos with transfer learning. Convolutional neural networks are commonly utilized in classification, segmentation, and image recognition nowadays. Deep network design is CNN's most powerful feature, allowing it to learn mid-to high-level concepts from fresh data automatically. For vegetable picture classification, we provided a CNN model as well as four, fine-tuned state of the art CNN architectures InceptionV3. A comparison of the performance of CNN and its architectures is also carried out.

2 Problem statement

In daily life, we see a lot of vegetables that belong to different countries and regions; Vegetables will have the same color, shape, and texture, which is challenging to recognize for a customer in the market, so we need to find out a way to solve this problem, so vegetable image classification came into the picture, where the model has recognized different type of vegetables using transfer learning by InceptionV3.

3 Related work

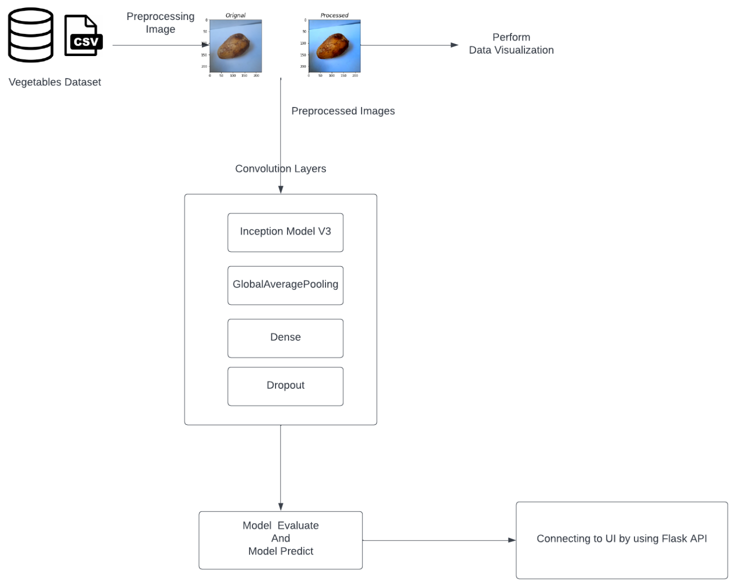
In the market, there are many vegetable image recognition, but they have used small datasets; the accuracy was also not that great with CNN models; we have used transfer learning(InceptionV3), which gave better accuracy. By this method, we can attain better accuracy, and recognition of vegetable images has improved.

4 Datasets

We aim to determine which method of vegetable picture classification has the highest accuracy. The first experiment uses 15 different types of common veggies worldwide. Bean, bitter gourd, bottle gourd, brinjal, broccoli, cabbage, capsicum, carrot, cauliflower, cucumber, papaya, potato, pumpkin, radish, and tomato are among the vegetables chosen for the experiment. A total of 21000 photos from 15 classes are used, with 1400 images of size 224*224 in .jpg format in each category. Dataset was split into 70 percent for training, 15 percent for validation, and 15 percent for testing..

5 Problem Solution

To classify challenging vegetable images for a customer in the market, we can use the CNN and CNN models with the transfer learning technique. Training a model from scratch is difficult, so transfer learning refers to leaving some layer weights and biases unfrozen and using them for teaching the pre-trained model to perform effectively on the training data. It has 48 layers, and after the last convolutional layer, the final fully connected layer was replaced with average pooling. The accuracy and time complexity are better when transferred learning is applied to vegetable image classification.



6 Data Pre-Processing

The goal of image processing is to improve visual information for human interpretation. Basic manipulation and filtering can also lead to an increased understanding of feature extraction. We increased the color saturation and contrast and sharpened the image to draw texture and focus. The image, after processing looks appealing and brighter We have used an image data generator to augment the images like rotation, shifting, etc. The Python Imaging Library (PIL) extends the Python programming language's image processing capabilities. It has a lot of file format support, a good representation, and decent image processing features. The core image library is designed to provide rapid access to data stored in many crucial pixel formats. It gives a solid foundation for an image processing tool in general. Enhance interface Each image enhancement function is implemented as an object using the Enhance interface. The interface only has one method:

Take, for example, Image Enhance. The Brightness class regulates the Brightness of an image and inherits it.The effect is controlled by the factor, which is a number. For example, an aspect of less than 1.0 in Brightness makes the image darker (and 0.0 makes the image completely black). The idea is brighter when the factor is more significant than 1.0. The original image is unaffected by a factor of exactly 1.0. Any image enhancement follows a fundamental pattern: Create an enhancer object with the source image as the input. Enhance the item, and a new, enhanced Image object is returned by using Image enhance libraries.

imageenhanced=ImageEnhance.Color(imageduplicate).enhance(1.35)

imageenhanced=ImageEnhance.Contrast(imageenhanced).enhance(1.45)

imageenhanced=ImageEnhance.Sharpness(imageenhanced).enhance(2.5).

7 Model Building

In this project, we have trained the vegetable image dataset with CNN and used transfer learning (InceptionV3); we have observed that CNN showed less accuracy when using transfer learning. In the CNN model, we have added hidden layers, max pool, and dropout layers with activation of Relu and softmax. In the case of the InceptionV3 model, we have to use transfer learning, a pre-trained model, to classify the vegetables..

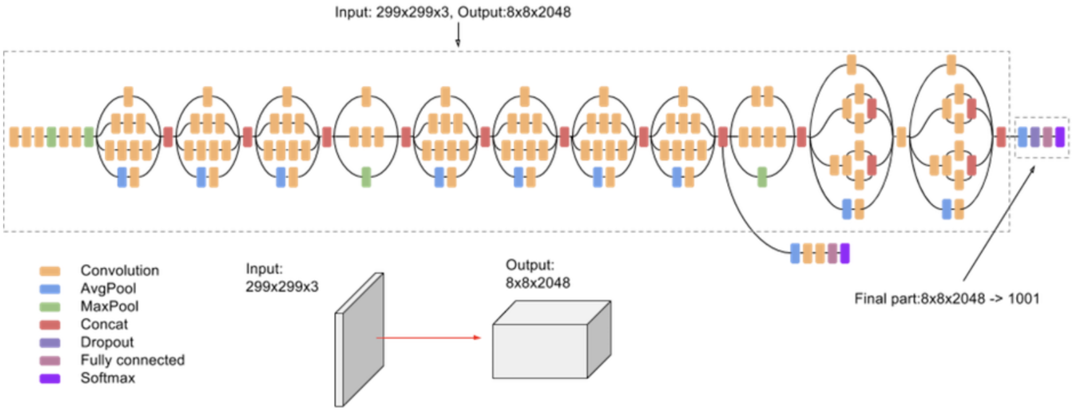
7.1 CNN

A CNN starts with an input layer and finishes with an output layer, with numerous hidden layers. The hidden layer includes convolutional, pooling, normalizing (ReLU), and fully-connected layers. The image is then modified to an appropriate size before being transmitted to the convolutional layer. The negative weighted input will be replaced with zero using an activation function; else, it will go straight to output. ReLU (Rectified-Linear Unit) is the most extensively used activation function, and it is the default activation function for many types of neural networks.

It's a non-linear function faster than Sigmoid and Scaled Exponential Linear activation functions. The pooling layer receives the features extracted from the convolutional layer. This layer preserves just significant aspects of a vast image by decreasing parameters. The fully connected layer then assigns categories to these highly filtered photos. Finally, a non-linear function called softmax offers each class decimal probability ranging from 0 to 1. The input image size is set to reduce total processing time and create an efficient model. It is also subjected to data augmentation techniques such as rotation, rescale, shear, zoom, and horizontal flip. ReLU was used as the activation function with each convolutional layer. And for improving generalization error, a dropout rate of 0.25 is used to overcome overfitting issues. Finally, softmax is used to find the probabilities of each class in decimal numbers..

7.2 InceptionV3

There are 48 layers in InceptionV3 and 24 million parameters. Convolutions, average and max pooling, concatenations, dropouts, and fully linked layers are among the model's symmetric and asymmetric building components. Batch normalization is done to activation inputs and is used extensively throughout the model. Softmax is used to calculate the loss.The input image size and the Inception architecture's built-in image pre-processing are used before sending them to the network. The Adam optimizer is employed with a learning rate of 0.0001 and a batch size of 64 in a five epoch training..



7.3 Results

After fitting the model, we observed an accuracy of 99 for transfer learning using InceptionV3; we plotted Accuracy curves with labels train accuracy and validation accuracy, which were generated after fitting the model. And also loss curves with labels train and validation loss. Now, we have tested the model on the validation data set. We have shown the single image prediction and validation directory accuracy prediction. A total number of layers and delicate tune layers are also displayed.

```
Total layers in the model :  315

First layer :  input_1
InceptionV3 layers : Layer 2 to Layer 311
Our fine tuned layers :  ['global_average_pooling2d', 'dense', 'dropout']
Final Layer :  dense_1
```

```
Accuracy for Pumpkin: 1.00 (200/200)
Accuracy for Broccoli: 0.99 (199/200)
Accuracy for Bottle_Gourd: 0.99 (199/200)
Accuracy for Carrot: 1.00 (200/200)
Accuracy for Papaya: 0.99 (199/200)
Accuracy for Capsicum: 1.00 (200/200)
Accuracy for Brinjal: 1.00 (200/200)
Accuracy for Cucumber: 0.98 (197/200)
Accuracy for Cauliflower: 0.99 (198/200)
Accuracy for Bitter_Gourd: 0.99 (198/200)
Accuracy for Potato: 1.00 (200/200)
Accuracy for Bean: 1.00 (200/200)
Accuracy for Radish: 0.99 (199/200)
Accuracy for Cabbage: 0.99 (199/200)
Accuracy for Tomato: 0.99 (199/200)
```

8 Post-processing

We have used Flask API to host our model, which will help to visualize the picture and predict the name of the vegetable,the model can predict single image and also folder of images.If we upload an image, the model returns the name of the vegetable.

9 Applications

The primary application of this project is picking and sorting in supermarkets, Identifying the diseases for vegetables and reduce the manual operation.

10 Future Work

Agriculture is the most critical sector. However, it is less digitally focused than other sectors. They are limited in scope, have a short dataset, and are less accurate. It performs vegetable picture categorization using a traditional CNN model and a CNN-based pre-trained InceptionV3. On the other hand, transfer learning approaches are used to fine-tune and apply pre-trained state-of-the-art CNN architectures. Only 15 varieties of vegetables were chosen from various species for the original research of the vegetable image categorization challenge.

A locally constructed dataset with 21000 photos from 15 classes is utilized for training and testing. Based on results, pre-trained CNN architectures are the future of machine vision. And for the time being, it's the maximum potential accuracy for the vegetable categorization challenge, which appears to be extremely promising. In supermarkets and warehouses, the sorting and labeling of vegetables can be automated to save time and human resources. We will be extending this project by including new classes and kinds and adding to the existing dataset to make it more robust..

11 References

1.https://www.kaggle.com/code/databeru/fruit-and-vegetable-classification/data

2.https://medium.com/mlearning-ai/vegetableimageclassification-using-cnn

3.https://www.researchgate.net/publication/352846889_DCNN-Based_Vegetable_Image_Classification_Using_Transfer_Learning_A_Comparative_Study

4.https://cloud.google.com/tpu/docs/inception-v3-advanced

5.https://medium.com/analytics-vidhya/transfer-learning-using-inception-v3

6.https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification