International Journal of Modern Agriculture, Volume 9, No.3, 2020
ISSN: 2305-7246

Fruit Detection Using Convolution Neural Network

Prashant Kumar[1], Tamal Datta [2], Sujata Chakravarty[3]
[1, 2, 3] Centurion University of Technology & Management, Odisha, India
Email:[1]170301120110@cutm.ac.in, [2]170301120056@cutm.ac.in,
[3]sujata.chakravarty@cutm.ac.in

Abstract

Fruit classification has emerged as a research area in the recent past. An algorithm based on convolution neural network (CNN) has been applied for fruit detection in this article. We have used high-quality, fruit-containing image dataset for training a neural network to detect fruits. The image regions are retrieved using a selective search algorithm. Computer vision is one of the most popular technologies in this era of innovation. The experimental results show that deep neural networks provide more accurate results compared with other machine learning algorithms. In this paper we briefly discuss the usage of deep learning algorithms and also CNN. This model works efficiently with this architecture for fruit recognition. And also we use various hidden layer combinations and epochs for different cases and compare them.

Keyword:- Fruit recognition; CNN; Fruit-360

Introduction

In the agriculture industry, the machine vision system for fruit classification is one of the most popular topics under research [1]. Over the last few years this computer vision technique is the most widely used technique for fruit classification. As a part of artificial intelligence (AI) and machine learning algorithm (ML), deep neural network (DNN) is used for fruit identification from images[2]. Efficiency of DNN is better than other machine learning algorithms.

Convolutional neural network (CNN) is the most commonly used algorithm in DNN which also performs efficiently for visual recognition including photo and video [3], face recognition [4], handwritten digit recognition. The efficiency of CNN can match human level perfection [5].

CNN has a similar architecture like another deep learning algorithm i.e., ANN. In ANN, several neurons are there in each layer and also the fact is that all the neurons are not fully connected; instead they are connected locally as a part of the receptive field[6]. After that the cost function is also generated for training purposes.

The accurate and efficient fruit recognition system has a great importance in the agriculture industry. So in our work we have used CNN as a fruit recognition classifier. We use the fruit-360 dataset[7] as our training and testing purpose. Input image contains 100×100 pixels of RGB image. we use different combinations of hidden layer and epochs to increase the model's efficiency [8].

Related work

We review some previous attempts to use neural networks and deep learning in this section of the report. The data used in this paper for testing and training consists of 17187 images of their fruits. The process used to find and count in two steps: the fruits are placed in a single image in the first stage and several views are merged in a second step to maximise the detection rate of the fruits. The aim is to create a neural network that can be used for fruit detection. In 2 different cases: early and late fusion, the mixture of the RGB and NIR models is carried out. Early developed for multiple that the input layer has 4 channels: 3 for the RGB image and one for the NIR image[9]. Late fusion uses 2 explicitly trained models that are blended using RGB and NIR (near infra-red) images by obtaining predictions from both models and combining the results. The network is trained. The outcome is a multi modal network 4 that operates much better than the previous networks.

Dataset

Here we are using fruit-360 dataset for training and testing purposes and this dataset is also available on kaggle. The dataset contains 65429 images with 94 categories. The images were taken with a white background, so when we remove that background the image is transformed into 100*100 pixels of standard RGB images. Fig one shows the dataset.

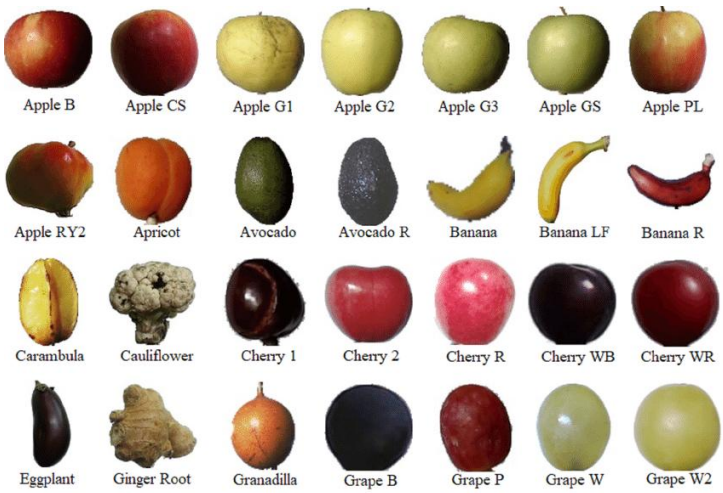

Fig.1 Dataset

From the fruit-360 dataset we select 22934 images with 34 categories. From this dataset we select 17,187 for training and 5747 for testing purposes of the model. This model is also trained for 15 epochs where the batch size is also 15.

Model Construction

As a part of AI, in deep learning a neural model understands the image data, learns from that and performs accordingly based on statistical analysis. These deep learning models are trained by huge amounts of labeled data and constructed neural network containing many layers for improving accuracy.

The concept of deep learning first released in 1980 and became popular for mostly two reasons ; a large amount of labeled data provides the most accurate result and it is more efficient with complex classification.

CNN Training Algorithm

The Deep Learning Models include convolutional neural networks (CNN). Such a network can consist of convolutional layers, pooling layers, layers of ReLU, completely linked layers and layers of loss. Every convolutional layer is preceded by a Rectified Linear Unit (ReLU) layer in a standard CNN architecture, then a Pooling layer then one or more convolutional layer and finally one or more completely connected layer. A normal neural network transforms the input into a one-dimensional array that renders spatial shifts less responsive to the qualified classifier.

Convolutional layers

After the convolution process, convolutional layers are called. Convolution is an operation on two functions in mathematics that creates a third function that is the modified (convoluted) variant of one of the original functions. The resultant function combines the pointwise multiplication of the two functions as a function of the quantity translated by one of the initial functions.

In CNN architecture two main transforms are there. One is Convolution, kernel is used for convolving pixels, and it results as a dot product of image patch and kernel. And second is subsampling (pooling), different types of pooling methods are used as per the requirement. Pooling is used to reduce the dimensionality of data and it’s quite helpful in reducing overfitting. The output can be fed to a fully-connected layer for efficient classification after using a mixture of convolution and pooling layers [[13],[14]].

- **Pooling layers** : Pooling layers are used on the one side to decrease the representation 's spatial dimensions and to decrease the amount of computation performed in the network. The other use of layers for pooling is to monitor overfitting. There are pools of scale 2 x 2 with phase 2 in the most used pooling layer. This restricts the input efficiently to a quarter of its initial size.

- **Fully connected layers :** The layers of a standard neural network are fully connected layers. Each neuron is related to every output of the previous layer from a completely connected layer. The operations behind a convolutional layer are the same as in a layer that is entirely connected. Thus, between the two, it is possible to transform.

- **Loss layers** To restrict the network for deviating from the predicted performance, loss layers are used. This is usually the network's last layer. There are different loss functions: softmax is used to forecast a class from multiple disjunct ones, and sigmoid cross-entropy is used to predict multiple independent probabilities. Fig 2 describes the model.

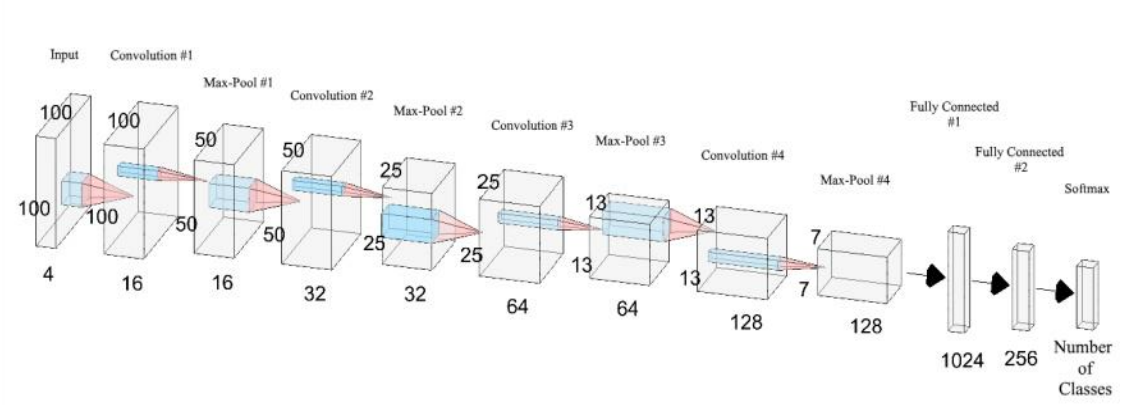

Fig.2 CNN Model

Keras

One of the most popular open source neural networks in Python is Keras. It has many purposes like pre processing,optimization, model evaluation etc. It has the running capacity on the top of Tensorflow. It is also comfortable for many high level API and also it can handle backend performance. It works more efficiently with a model to optimize the loss and optimizer function.

For a simple stack of layers where each layer has exactly one input tensor and one output tensor, a model is suitable.

A Sequential Model is not suitable if:

-There are multiple inputs or multiple outputs in your model

-Many inputs or multiple outputs in any of the layers

-Layer sharing needs to be done

Result and Analysis

Experimental Process

22934 fruit images are stored in the database. 17187 images are for training, 5747 for testing, and are used for processing. Input these images to the limited search algorithm. Around twenty regions will create an image. A good percentage of regions can be accessed via the algorithm. Some of them, though, are too short or too tall or too fat and provide some knowledge about the fruit. We exclude all lengths or widths that are less than 0.2 times the actual length or width and the original picture dimension. Regions that are smaller than 6.77 are therefore excluded by measuring the entropy of each region.

Experimental Result

In this article we have applied CNN on the most common dataset fruit-360 for finding the best classification result of the CNN model. To improve the overall accuracy we have taken

10 cases where different hidden layer (convolution and pool) combination works with 30 epoches with a batch size 20 and throughout this process the train and test accuracy is improved. In Fig 3 both training and testing accuracy is given.

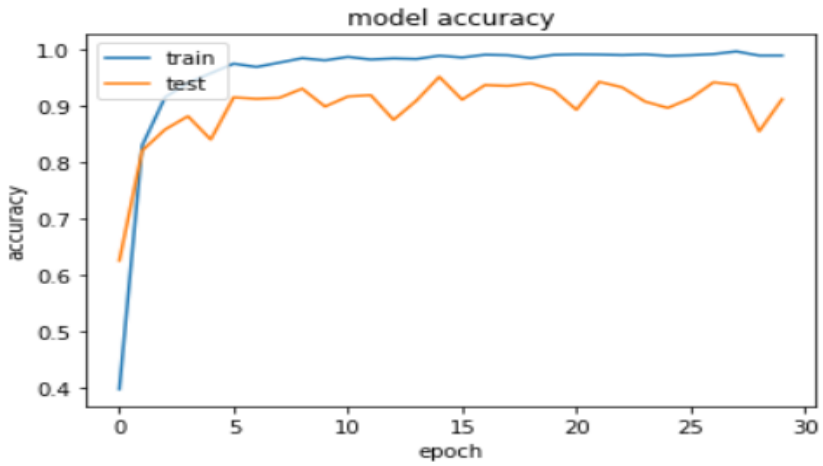

Fig.3 Model Accuracy

Table 1 Performance of the Model

Model accuracy	98%
Model loss	0.077%
Test accuracy	91%
Test loss	1.09%

Analysis:

A convolutional layer that applies 16 3 x 3 philtres is the first layer (Convolution # 1). We implement max pooling on this layer with a phase 2 form 2 x 2 filtering that specifies that the pooled regions do not overlap (Max-Pool # 1). It also lowers the height and width to 50 pixels each.

-We implement the same form of max pooling(Max-Pool # 2) on this layer as on the first layer, shape 2 x 2 and phase 2.

-the third convolutional (Convolution # 3) 64 3 x 3 images belong to the sheet. The following is another max pool layer of form 2 x 2 and step 2 (Max-Pool # 3).

-the fourth convolutional (Convolution # 4) layer, 64 3 x 3 philtres are added to the fourth convolutional (Convolution # 4) layer, after which we add a final max pool layer (Max-Pool # 4).

-The dimensions of the representation have each been reduced by a factor of 32 due to the four max pooling layers, so the fifth layer has 5 x 5 32 inputs, and is a completely connected layer(Fully Connected # 1).

-With 1600 inputs and 512 outputs, this layer feeds into another completely connected layer (Fully Connected # 2).

-A Softmax Loss Layer (Softmax) with 512 inputs is the last layer. The production number is equivalent to the number of groups.

Table 2 Layer Types

Layer type	**Dimensions**	**Output**
Convolutional #1	5 x 5 x 4	16
Convolutional #2	5 x 5 x 16	32
Convolutional #3	5 x 5 x 32	64
Convolutional #4	5 x 5 x 64	128

Conclusion and future work

In this paper, a fruit recognition classifier based on CNN algorithm has been developed. Various combinations of CNN's hidden layer makes an important effect on the model accuracy and other loss curves. So there are many test cases, among them case 4 from 11 to 15 epoches the model achieved the best accuracy of 97% and also the best training accuracy is 98%, which is also improved throughout the result. So both the train and test accuracy is higher, which is useful for stimulating the models overall performance for this fruit recognition classifier. There is also one side of loss dropout, so when the loss will be low the CNN model performs better. For future work we can perform the segmentation process on the image before applying CNN.

References

1. Bargoti, S., and Underwood, J. Deep fruit detection in orchards. In 2017 IEEE International Conference on Robotics and Automation (ICRA) (May 2017), pp. 3626–3633.
2. Barth, R., IJsselmuiden, J., Hemming, J., and Henten, E. V. Data synthesis methods for semantic segmentation in agriculture: A capsicum annuum dataset. Computers and Electronics in Agriculture 144 (2018), 284 – 296.
3. Chan, T. F., and Vese, L. A. Active contours without edges. IEEE Transactions on Image Processing 10, 2 (Feb 2001), 266–277.
4. Cheng, H., Damerow, L., Sun, Y., and Blanke,M. Early yield prediction using image analysis of apple fruit and tree canopy features with neural networks. Journal of Imaging 3, 1 (2017).
5. Cires¸an, D. C., Giusti, A., Gambardella, L. M., and Schmidhuber, J. Deep neural networks segment neuronal membranes in electron microscopy images. In Proceedings of the 25th

International Conference on Neural Information Processing Systems - Volume 2 (USA, 2012), NIPS'12, Curran Associates Inc., pp. 2843–2851.

6. Woo Chaw Seng and Seyed Hadi Mirisaee, "A New Method for Fruits Recognition System," MNCC Transactions on ICT, vol. 1, no. 1, June 2009.
7. T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learning rates," in International Conference on Machine Learning, 2013, pp. 343-351.
8. C. Hung, J. Nieto, Z. Taylor, J. Underwood, and S. Sukkarieh, "Orchard fruit segmentation using multi-spectral feature learning," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 5314-5320.
9. T. F. Chan and L. A. Vese, "Active contours without edges," IEEE Transactions on image processing, vol. 10, pp. 266-277, 2001.
10. T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learning rates," in International Conference on Machine Learning, 2013, pp. 343-351.
11. K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in Competition and cooperation in neural nets, ed: Springer, 1982, pp. 267-285.
12. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, et al., "Backpropagation applied to handwritten zip code recognition," Neural computation, vol. 1, pp. 541-551, 1989.
13. Yann LeCun, Yoshua Bengio & Geoffrey Hinton "Deep learning", Publisher: Nature, Published on: 27 May 2015
14. Agnieszka Mikołajczyk; Michał Grochowski, "Data augmentation for improving deep learning in image classification problem", Publisher: IEEE, 21 June 2018.