Turkish Journal of Computer and Mathematics Education Vol.12 No.13 (2021), 7593 – 7600

Research Article

Examination of the Basics and Concepts of Deep Learning Networks and Object Detection for Tomato Detection

Mahyar Gohari Moghaddam[a]

[a] Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.
mahya_gm@aut.ac.ir

Article History: Received: 14 July 2020; Accepted: 2 January 2021; Published online: 5 February 2021

Abstract: The study examined the past studies, the advantages and disadvantages of each, and the history of algorithms and methods of object detection. Particularly, the studies on the detection of tomatoes and other agricultural products were examined as well. We then reviewed the proposed models and methods for object detection in the image, especially methods based on deep networks. Finally, we briefly compared some of the most widely used methods in this regard nowadays used by scholars and engineers in different problems in this field and examined the papers regarding the detection and localization of agricultural products, particularly tomatoes.

Keywords: Machine Vision, Tomato Detection, Classification, Deep Learning, Image Processing

1. Introduction

The main challenge of deep networks is the existence or possibility of collecting a large and high-quality dataset including various types of images from different categories. Another challenge associated with the nature of this problem is the different light conditions in various places of the greenhouse and the various color properties of the tomatoes compared to each other. For instance, a small part of tomato may look semi-ripe for different reasons, yet it may be ripe. This will make classifying tomatoes hard. The study examined the past studies, the advantages and disadvantages of each, and the history of algorithms and methods of object detection. Particularly, the studies on the detection of tomatoes and other agricultural products were examined as well. We then reviewed the proposed models and methods for object detection in the image, especially methods based on deep networks. Finally, we briefly compared some of the most widely used methods in this regard nowadays used by scholars and engineers in different problems in this field and examined the papers regarding the detection and localization of agricultural products, particularly tomatoes.

2. Methods of object detection

The subject of object detection is one of the most challenging problems in the field of machine vision. Here are some of the most important ways to solve these problems. One of the earliest approaches for recognizing an object in an image, introduced in 2001, is the Vaiola Jones Detector or the sliding window [1]. The method used a sliding window to search for the face in the image. The results of the method were very impressive at that time, yet the inefficiency of the model in recognizing images rotated led to the introduction of the directional gradient histogram method in 2005 [2]. This method could solve the problem of the Viola-Jones method and make a drastic change in the methods of object detection. This method is still used nowadays in some problems where little educational data is available.

Using deep networks has grown significantly and has reached significant results in the field of object detection given the increase in the information available and the increase in the hardware power of computers. Among the object detection methods using deep networks, Regional Convolutional Neural

Network (RCNN) model in 2014 could reach very good results in the field of object detection in the image [3]. Using a selective search, the model generates a specific number of regions and detects the presence of an object in that region using a Convolutional Neural Network (CNN). An improved model of this algorithm was introduced in 2015 entitled Fast-RCNN. In this model, instead of running the CNN algorithm for the proposed region, it first implements the CNN algorithm over the entire image and then identifies the proposed image regions from the map of the obtained properties [4]. Despite the sharp increase in speed in this method, the proposed areas are still bottle-necking of this method and it has a high execution time. Furthermore, selective search is less efficient than newer innovative methods for finding suggested image areas. The newest member of the RCNN family is the Faster-RCNN algorithm, introduced in 2015 [5]. The proposed regions will be generated by another neural network in this method instead of using selective search after generating the feature map. This method is much faster than the previous two models of this family.

Another method used for problems that need an immediate response because of the much higher speed than the regional methods mentioned is the You Only Look Once (YOLO) algorithm first introduced in 2016 [6]. The idea general of the method is using a CNN network that simultaneously identifies the possibility of existence and categorization in one area of the image. The advantage of the approach over the previous ones is in seeing the complete image for network training, unlike the area models where the training happens in various parts of the image. Although the early version of YOLO had lower accuracy than the previous area methods, it could reach a very good accuracy in later versions and some modifications. Table 1 compares the results of various versions of this algorithm with each other and the Faster-RCNN algorithm [5] on the standard MS COCO dataset [7].

Table 1. Results of different object detection methods on the MS COCO dataset.

Row	Method	Accuracy in terms of Mean Average Precision (MAP)	The number of image processing per second
1	Faster R-CNN [5]	34.7	5
2	YOLOv2 608 × 608 [8]	16	40
3	YOLOv3-320 [9]	28.2	45
4	YOLOv3-416 [9]	31	35
5	YOLOv3-608 [9]	33	20
6	YOLOv3-spp [9]	36.2	20
7	YOLOv4-416 [10]	41.2	38
8	YOLOv4-512 [10]	43	31
9	YOLOv4-608 [10]	43.5	23
10	PP-YOLO [11]	45.2	72.9

Methods of detection of tomatoes and other agricultural products

The need for a robot that can accurately and quickly detect and locate agricultural products, especially tomatoes, has increased with the expansion of using machinery in agriculture, and many people in different studies and industrial projects have addressed this problem.

In 2011, paper [8] addressed this problem using RGB, HSI, and YIQ color spaces and using image morphological features. Despite the high accuracy of the results, the method does not work well for images that have highly overlapping objects or the ambient light conditions are not suitable. The publishers of the paper in 2013 could enhance their results by using neural networks and wavelet transform [9]. However, this solution could not be implemented in practice as in the real environment, the color characteristics of the image alone could not separate the image background from the target objects. Other papers dealt with this problem using neural networks and support vector machines (SVM) [10, 11].

With the advancement of computer hardware and the availability of more data that resulted in the expansion of the use of deep networks, like most machine vision problems, the methods for solving the automatic tomato detection problem evolved and shifted to using deep networks. Although these networks are more accurate than older methods, researchers and developers face two major challenges too. The first is the collection of very high volumes of quality data needed for network training, and the second is the complexity and high volume of the model, resulting in the low speed of implementation of these algorithms.

In recent years, scholars have published many papers on this subject to overcome these challenges, some of which are stated below. In 2016, the paper [12] used CNN to segment the image and identify the fruit; however, the results were not accurate for areas of the image that had good quality because of poor lighting conditions. In the same year, paper [13] identified the sweet pepper using the Fast-RCNN method that besides the problem of not real-time results in location, did not have the desired accuracy. Using the Faster-RCNN algorithm in the detection of fruit in the image in 2017, the authors of the paper [14] reached good outcomes. However, as already stated in the previous section, as this network has two networks that process the image in parallel, it has high complexity and therefore the detection speed is not very high.

As the YOLO algorithm needs no region proposal network (RPN), it has a higher speed and besides performs better in the detection of small objects, it can be a great option for detecting agricultural products, especially tomatoes in the image. In 2019 and 2020, respectively, papers [16,15] presented a method based on the third version of the YOLO algorithm to solve the problem of diagnosing and determining the ripening rate of apples and tomatoes. This method is far better than the conventional method used for fruit detection operations, using Faster-RCNN and VGG16 networks [17].

Basic concepts of machine learning

This section examines some of the definitions and concepts in this regard. Moreover, the learning algorithms used are explained too.

Types of learning problems from a supervision perspective

Supervised learning refers to a set of machine learning problems and methods where the supervision taken from the existence of a label for data ends in gradual learning. In other words, for the problem, a set of input-output pairs (data with labels) is provided and the system tries to learn a function of input-output. This type of learning needs labeled data to train the system that might not be readily available for a range of machine learning problems [18].

On the contrary to the observed learning, there is another type of learning known as unsupervised learning. As the name implies, there is no labeled data for the problem in this type of learning. This type of learning is often used for datasets and features so that we can extract useful properties and structures of this dataset structure [18].

Moreover, another type of learning called semi-supervised learning is a group of problems and methods that use data with and without labels and is placed between two types of observed and unobserved [18].

In traditional machine learning, most of the problems and methods of machine learning can be divided into the mentioned categories from the observer's perspective. In modern machine learning, with the emergence and expansion of new problems and methods, special kinds of concepts and methods of learning monitoring have been proposed that cannot be included in the classification or concepts under the title of learning. For instance, one can refer to Transfer Learning or concepts such as teacher-student learning and knowledge distillation that is very significant in Hinton et al. [19].

Different learning problems from the output perspective

This section is some of the definitions of the types of problems needed are stated along with their references.

- **Classification [18]:** Classification is a method of supervised learning. In this method, the data are classified into k specific categories with a clear criterion for classification. In other words, all data has a label and each label shows that the data belongs to one of the categories.

Regression [18]: Regression is a method of supervised learning. In this type of problem, a numeric value is needed to be estimated based on the input. The algorithm is trained to estimate a numerical value for a given input to solve these problems. These types of problems are similar to classification except that the output has to be a numeric value instead of a specific category.

Image processing and deep neural networks

Deep neural networks are widely used in image processing nowadays. Traditional machine learning methods in image processing usually have two parts: feature extraction and the main model. In the feature extraction stage, we usually convert the input to low-dimension vectors, and try to make the generated vectors meaningful, and include the important input properties to solve the problem. The purpose of the main model is to solve the problem using the extracted features.

One of the problems of this method is heavy reliance between the input and the extracted features with the original model. The extracted features have a direct effect on the performance of the original model, and this dependence makes the feature extraction step very difficult for the problem-solving process, and usually, the human-extracted features are not efficient enough to solve the problem.

Unlike this traditional approach, one can delegate the task of extracting the feature to the main model so that the features extracted from the input are more appropriate to the original model. In this process, where human has no role, the extracted features are optimized as part of the problem-solving process in line with the original model. These optimized features are often more efficient for the original model during the problem-solving process and significantly enhance the accuracy of the original model.

Forward neural networks are a type of artificial neural network where the connections do not create distance. Usually, these types of networks are divided into two types of perceptron, single-layer, and multi-layer. The backpropagation method [20] is usually used to train multilayer perceptron.

Convolutional neural networks are a special type of deep neural network. Convolutional networks are designed inspired by visual receptors in the visual cortex of the brain and try to mimic the function of small sub-visual areas.

Convolutional neural network

The convolutional neural network is one of the most important methods of deep learning and in general, its main idea is to learn data patterns. As the data learning process is purposeful, the convolutional

neural network can be attributed to the supervised learning group. Typically, each convolutional neural network has many convolutional, Max Pooling layers, and dense layers, each of which has its own defined function. A dense layer is a hidden layer of neurons, to which all neurons have connections. For each convolutional neural network, two steps are considered for network training.

Backpropagation stage and feed-forward stage: In the first step, input is given to the network, and this operation is nothing but a point multiplication between the input and the weight parameters of each neuron, and finally the operation of the activator function in each layer. Then the network output is calculated and for training, we calculate the network error from the network output using the correct answer. In the feed-forward phase, according to the chain rule, all network parameters are changed according to the calculated error. Changing the network weights to reduce the calculated error is a kind of optimization problem. Several different optimization methods have been proposed to solve this problem [21-24].

Convolutional layer

The convolutional layer could be considered the most important layer of convolutional neural networks. Indeed, the convolutional layer is the core of the convolutional neural network. The output of the convolutional layer on a matrix can be thought of as a three-dimensional tensor. Overall, the convolutional layer has a set of filters to extract the appropriate features that these filters are trained for during network training. The convolutional layer consists of three-dimensional masses of neurons, each layer of which corresponds to a filter to implement these filters in the convolutional layer. Indeed, each of these filters is located as a hidden layer in the network [18].

Figure 1 is a convolutional layer, the filters or cores of which have been applied to the block. Each element of the specified vector is the result of applying a filter to the specified block. The size of the filters and the stride of applying the filter on the input are the variables of this layer specified in the architecture of each network [18].

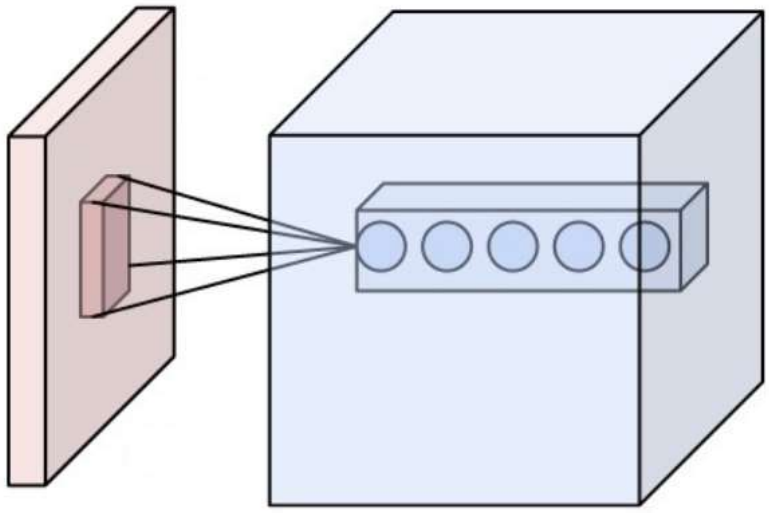

Figure 1: Convolutional layer image

Max Pooling layer

Max Pooling layer is widely used in convolutional neural networks. The function of the Max Pooling layer is to reduce the size of the matrices to reduce the number of parameters and simplify calculations within the network. The Max Pooling layer acts independently on each matrix of neuron outputs and modifies it using a maximization operation. The most common way to use this layer is to use this layer with filters of size 2 × 2 with step length two, where each section of the 2 × 2 blocks of the input matrix is mapped to several leads in the output matrix.

Figure 2-3 is an example of a Max Pooling layer where the largest value of each 2× 2 blocks of the input matrix is mapped to a number at the output. In such layers, the size is usually reduced by 75%. This reduction in features makes network training easier and better.

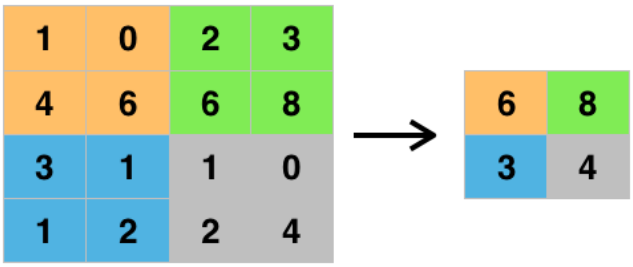

Figure 2. An example of a Max Pooling layer

Dense layer

These fully integrated layers convert two-dimensional properties into one-dimensional property vectors. Dense or fully connected layers act like hidden layers in traditional artificial neural networks. These layers allow us to present the grid result as a vector of a given size. We can use this vector to categorize images or use it to continue further processing.

One-Hot coding method

Nominal data are the variables that do not take numeric values. The number of possible values for these variables is usually limited to a fixed set. For instance, the color of a machine is a nominal variable. In machine learning, some algorithms can work directly with nominal data. For instance, in the decision tree, nominal data can be used directly, but this data cannot be used directly in some algorithms. This means that nominal data must be converted to numeric data. This process of converting to numerical data is called coding.

In the simplest mode, a numeric value can be assigned to each of the unique values. For instance, for color, white can be considered 1, black 2, and blue 3. For nominal variables with no sequential relationship between them, this coding is not very correct as this type of coding has given sequential relationship values while this is not the case. In these cases, coding can be very useful. In this encoding method, each of the possible values is mapped to a One-Hot vector that has all the values of the vector zero except the item corresponding to its numeric value that has a value of 1. For instance, each of the colors white, black and blue can be shown as follows.

• White: (1, 0, 0)

• Black: (0, 1, 0)

• Blue: (0, 0, 1)

Intersection over Union (IoU) criteria

The IoU measures the quality and accuracy of predictions generated by an object detection algorithm relative to its correct label. Indeed, this criterion shows how correctly the detected object is detected. This

criterion measures the extent of overlap between two regions or frameworks. This value is equal to, the area resulting from the commonality of these two areas, divided by, the area resulting from the union of these two areas. There are two frameworks for evaluating the quality of a detected object: the framework with the object identified and the framework defined by the label. Figure 2 shows these cases.

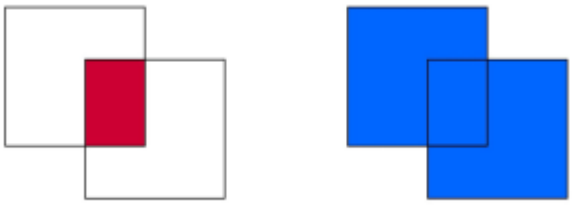

Figure 2. The blue hachure shows the community and the red border the commonality of the two frames

Thus, if the two match, the IoU criterion is the same for them, and the less compatible they are, the lower the value.

3. Conclusion

In this study, smart strategies were examined for detecting tomatoes in fields and classification of them in terms of ripeness to estimate the appropriate time to harvest them from plants. As the tomato plant is a fruit that bears fruit asynchronously, its harvesting has to be done daily. This harvest becomes more sensitive and difficult when one knows that the tomatoes have to be separated from the plant at various stages, depending on the distance from the destination to the field. As this system has good accuracy both in classifying and locating tomatoes, one can be used at a very low cost on tomato harvesting robots and enhance farm productivity both financially and in terms of product quality.

References

1. Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, vol. 1, pp. I-I. IEEE, 2001.
2. Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886-893. IEEE, 2005.
3. Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587. 2014.
4. Girshick, Ross. "Fast r-cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448. 2015.
5. Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." In *Advances in neural information processing systems*, pp. 91-99. 2015.
6. Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. "You only look once: Unified, real-time object detection." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779-788. 2016.
7. Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft coco: Common objects in context." In *European conference on computer vision*, pp. 740-755. Springer, Cham, 2014.

8. Arman Arefi, Asad Modarres Motlagh, Kaveh Mollazade, and Rahman Farrokhi Teimourlou. "Recognition and localization of ripen tomato based on machine vision." In *Australian journal of crop science(AJCS)*, pp. 1144-1149. 2011.
9. Arman Arefi and Asad Modarres Motlagh" Development of an expert system based on wavelet transform and artificial neural networks for the ripe tomato harvesting robot." In *Australian journal of crop science(AJCS)*, pp. 699-705. 2013.
10. Mhaski, Ruchita R., et al. " Determination of Ripeness and Grading of Tomato Using Image Analysis." In *2015 Communication, Control and Intelligent Systems (CCIS)*, pp. 214-220. 2015.
11. El-Bendary, Nashwa, et al. "Using Machine Learning Techniques for Evaluating Tomato Ripeness." In *Expert Systems with Applications*, pp. 1892–1905. 2015.
12. Hulin Kuang, Leanne L. H. Chan, Cairong Liu, and Hong Yan "Fruit classification based on weighted score-level feature fusion," *Journal of Electronic Imaging*, 2016.
13. E. Vitzrabin and Y. Edan, "Changing Task Objectives for Improved Sweet Pepper Detection for Robotic Harvesting," in *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 578-584, 2016.
14. Bargoti, Suchet, and James Underwood. "Deep fruit detection in orchards." In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3626-3633. 2017.
15. Tian, Yunong, et al. "Apple Detection during Different Growth Stages in Orchards Using the Improved YOLO-V3 Model." In *Computers and Electronics in Agriculture*, vol. 157, pp. 417-426. 2019.
16. Thakur, Rucha, et al. "An Innovative Approach For Fruit Ripeness Classification." In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020.
17. Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 2014.
18. Hinton, Geoffrey, Vinyals, Oriol, and Dean, Jeff. “Distilling the knowledge in a neural network”. *arXiv preprint arXiv:1503.02531*, 2015.
19. Goodfellow, Ian, Bengio, Yoshua, and Courville, Aaron. *Deep learning*. MIT press, 2016.
20. White, Halbert. "Learning in artificial neural networks: A statistical perspective." *Neural Computation*, 1(4):425–464, 1989.
21. Kingma, Diederik P and Ba, Jimmy. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980*, 2014.
22. Graves, Alex. "Generating sequences with recurrent neural networks." *arXiv preprint arXiv:1308.0850*, 2013.
23. Duchi, John, Hazan, Elad, and Singer, Yoram. "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of machine learning research*, 12(Jul):2121–2159, 2011.
24. Schaul, Tom, Zhang, Sixin, and LeCun, Yann. "No more pesky learning rates." *In International Conference on Machine Learning*, pages 343–351, 2013.