The Investigation of Performance Comparison for VGG, YOLO, and DINO in Image Classification

Yanqi Chen *

School of software, South China Normal University, Guangzhou, China

* Corresponding author: 20192033007@m.scnu.edu.cn

Abstract. The rise of artificial intelligence has led to a proliferation of deep learning models, yet there remains a noticeable shortage of comparative analyses, particularly among computer vision models rooted in different design philosophies. As such, this study seeks to delve into the strengths of various models through an examination of their structural attributes, with the aim of offering insights that can inform the development of more high-performing models in the future. This study first selects three representative models with different design ideas in their respective research directions, and preliminarily distinguishes the differences between different models. Then, through experiments on the dataset, the performance of different models is obtained, and the reasons for their current performance are analyzed. In this experiment, four models, VGG16, YOLOv5, YOLOv8, and DINOv2, were deployed and tested using the Fruit 360 dataset. The final accuracy was 0.955, 0.997, 0.998 and 0.986, respectively. The accuracy of YOLO model and DINO model was much higher than that of VGG model. The reason for this result may be related to the introduction of anchor boxes in the YOLO model and attention mechanisms in the DINO model, both of which indirectly increase the receptive area for feature extraction. The YOLOv8 model has a slight improvement in accuracy compared to the YOLOv5 model, possibly due to its use of a decoupled head, which reduces the impact of location information on classification tasks.

Keywords: YOLO, VGG, DINO, deep learning.

1. Introduction

As computer technology develops rapidly, the discussion on the advantages and disadvantages of artificial intelligence model frameworks has become increasingly intense. With the continuous deepening of neural network research, artificial intelligence has had a profound impact on people's lives. It can provide constructive suggestions for future development by analyzing past data, comprehending human language and communicating with humans naturally, as well as recognizing and distinguishing various objects individually. An increasing number of models using different network structures and training algorithms are being proposed, but it is significant to select the best model to apply in the industry.

Experts started to do experiments to identify factors that affect model performance and attempted to find the most optimal model in different subdivision fields. For example, Madhiarasn et al had an insight into the artificial neural network performance of temperature forecasting applications [1]. Convolutional Neural Network (CNN) was a notion proposed by LeCun in 1989 [2]. Its emergence provided a foundation for the research of computer vision and was later widely applied in low-level feature extraction. Visual Geometry Group (VGG) is a specific implementation of CNN, which uses smaller convolutional kernels so that the number of layers of the neural network is able to increase. Simultaneously, this experiment also proved that a deeper neural network structure has better performance [3]. Subsequently, You Only Look Once (YOLO) emerged as a product of the shift in research objectives of the computer vision field from image classification to object detection. YOLO inputs the entire image into the network and divides it into multiple cells. Each cell is only responsible for determining the object with the centre located within it. Then, the model merges the cells that determine the same object. Consequently, the model outputs an object detection map [4]. Nowadays, foundation models are extremely popular in the field of natural language processing. Since more parameters they have the potential for higher validation accuracy. At the same time, their self-supervised characteristic and pre-training process bring high commercial value. Therefore, the

foundation model is the future research direction in the field of artificial intelligence [5]. As models increase, comparative studies between different models burgeon relatively. In the field of data prediction, CNN, Recurrent Neural Network (RNN), Long Short-term Memory (LSTM), Bidimensional LSTM (Bi-LSTMs), Gated Recurrent Unit (GRU), and Transformers models were used for comparison. By analyzing the accuracy of these models, it was ultimately found that Transformers has better predictive performance compared to other models [6]. However, most model comparisons are only based on results, without a deeper comparative analysis of the performance improvement brought by model structure. At the same time, there is hardly any research about the performance comparison of models in the field of computer vision, but most of it focuses on the field of data prediction. Generally, a model which performs better in comparison may also be able to get a high score in practical applications. Furthermore, the models currently being compared are relatively old, and advanced foundation models have not been included.

Therefore, it is meaningful to conduct comparative research on models focusing on the field of computer vision. This research specifically focuses on comparing the differences between VGG, YOLO and self-distillation with no labels (DINO) model architectures, attempting to figure out their performance under specific datasets. This study will deploy the virtual environments of these three models separately, install relevant dependencies, and conduct performance testing on the Fruit 360 dataset.

2. Method

2.1. Dataset Preparation

In this project, Fruits 360 is chosen as the dataset, which is obtained from the Kaggle [7]. This dataset is totally comprised of 131 categories of fruits and vegetables. There are 90483 images in the dataset, including 67692 pictures for training and 22688 pictures for testing. Moreover, all of their sizes are 100 × 100 pixels. And they all belong to RGB images. The following Fig 1 is some original visualizations of the images.

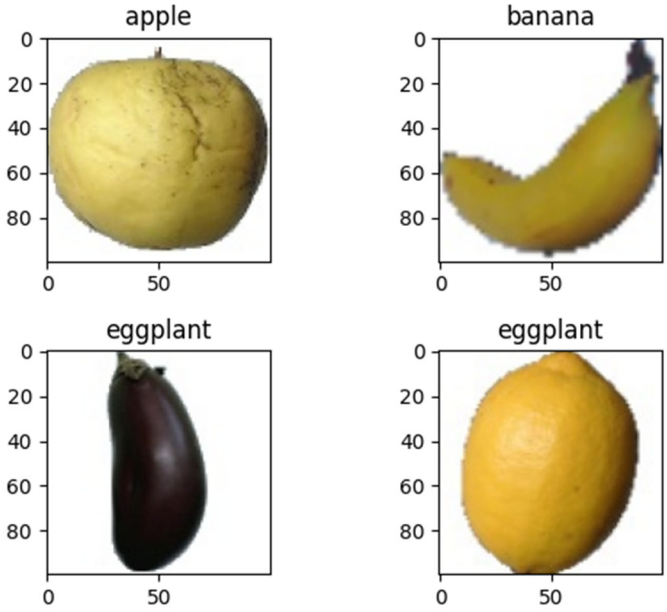

Figure 1. The sample images of the collected dataset [7].

In this experiment, data was preprocessed in the VGG16 model, but not in the other two models. As the VGG model is relatively simple, increasing the amount of data can achieve a relatively better effect. For example, the random rotation angle range of the image is set to 20, the cropping degree is set to 0.1, and the magnitude of random scaling is set to 0.2.

2.2. Deep Learning Models

There are a variety of methods that emerged during previous years. Most of them were proved feasible and effective. However, this paper aims to have a deep insight into three deep learning models. They are VGG, YOLO and DINO. By comparing their structures, their drawbacks and benefits will be discovered. The conclusion will be verified by running the implementation codes of these three methods.

2.2.1. Visual Geometry Group (VGG)

A significant feature of the VGG model is that it uses smaller convolutional kernels on top of the previously proposed models. There are many benefits in this way, one of which is making the model deeper. The process of training convolutional neural networks is the process of continuously extracting abstract features through convolutional kernels, in other words, extracting useful information from images. A deeper network structure means that the network has the ability to extract more important features [8]. For example, VGG uses three 3x3 convolutional kernels instead of the 7x7 convolutional kernels in AlexNet, which changes the model from only extracting features once to three times. Moreover, these three times of extraction are continuously refined, meaning that the latter abstraction continues on top of the previous one, resulting in higher-level abstractions. Another advantage is that it has fewer parameters while ensuring receptive fields, which means it is easier to train [9]. For example, a 7x7 convolutional kernel has 49 parameters, while three 3x3 convolutional kernels only have 27 parameters. At the same time, compared to large convolutional kernels, small convolutional kernels need to be moved more times for images of the same size in both step sizes of 1, which means they can better read the details of images [10]. The following Fig 2 shows the structure of the classic VGG16 model.

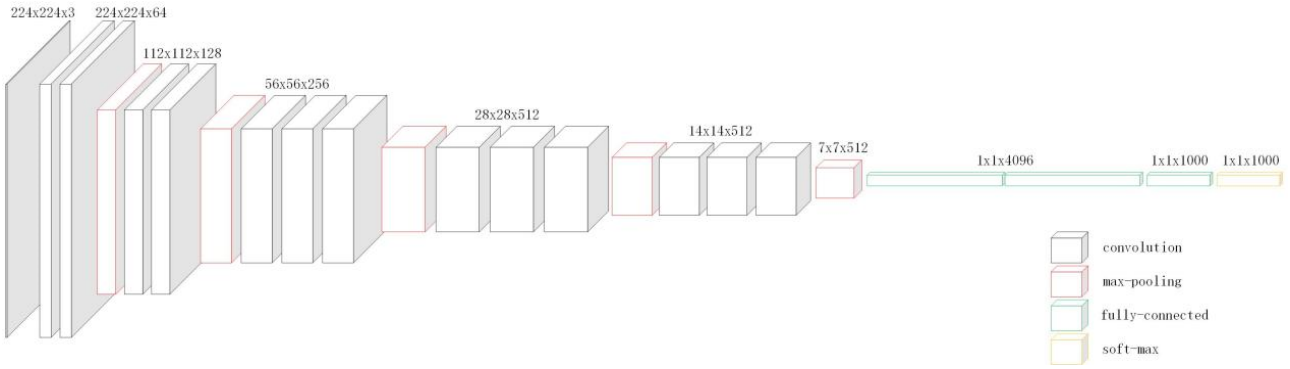

Figure 2. The architecture of VGG16 [3].

In the experiment, the input layer of the model was changed to 100x100x3 to match the image size in the dataset. In order to enable the model to train quickly, the model adopts pre-training parameters except for the last three fully connected layers. The number of neurons in the Soft max layer is set to 131, which is the same as the number of categories. The loss function uses the CategoricalCrossentropy class in Probabilistic losses, while the optimizer uses RMSprop. Meanwhile, the epochs are 10 and the batch size is 32.

2.2.2. You Only Look Once (YOLO)

In this project, two models, YOLOv5 and YOLOv8, were used for testing. The most significant feature of YOLO class models is that they simplify the two steps in object detection into one step. The conventional approach for object detection is to first identify candidate regions of suitable size, and then classify the candidate regions [11]. YOLO directly scans the entire image and provides both category and location information in one step [12]. YOLO first divides the image into several small pieces, and then creates bounding boxes for each small piece. The bounding boxes can be large or small, and may even exceed the size of the original small block, but YOLO only requires the object centre of the bounding box to be in the small block. In this way, YOLO transforms the problem into a regression problem. Fig 3 shows the workflow of YOLO in the objective detection.

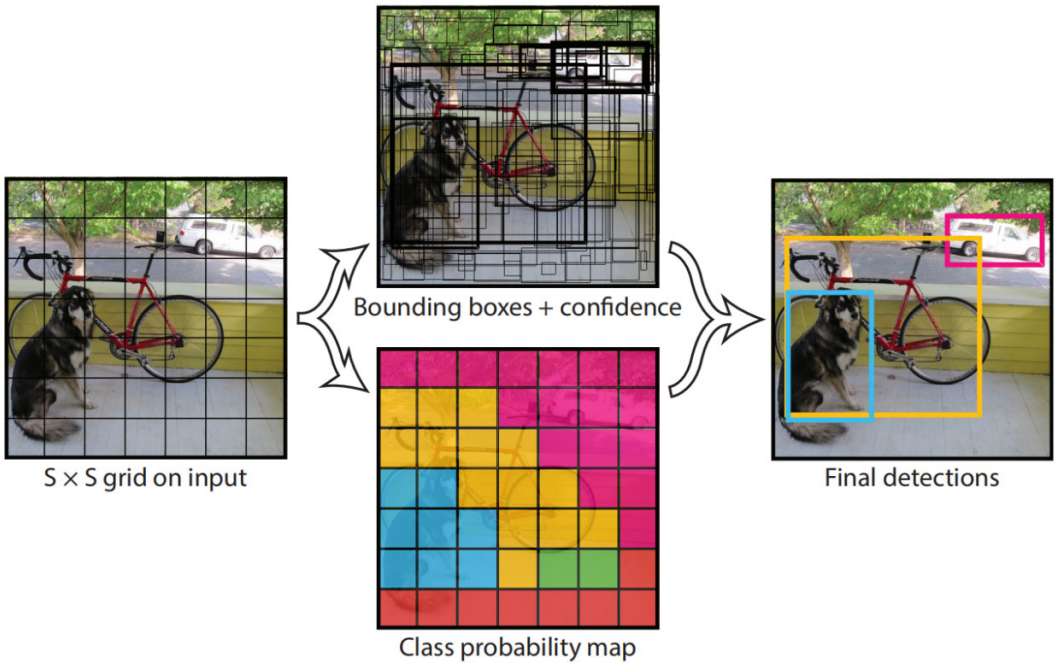

Figure 3. The workflow of YOLO in the objective detection [4].

YOLOv5 has a significant feature, which is its first development based on a better ecosystem of Pytorch, making it easier to deploy compared to previous models. Whereas compared to YOLOv5, a significant feature of YOLOv8 is its use of the decoupled head and the anchor-free.

In the YOLOv5 experiment, yolov5s was used as a pre-training parameter to import the model. The optimizer uses Adam. Meanwhile, the epochs are set to 20 and the batch size is 64. In the YOLOv8 experiment, yolov8s cls was used as a pre-training parameter to import the model, while the optimizer used auto. Meanwhile, the epochs are 25 and the batch size is 16.

2.2.3. Self-Distillation with No Labels (DINO)

In this study, the DINOv2 model was used for comparison. An innovation of the DINO model lies in its combination of unsupervised training methods with Transformers and its application in the field of computer vision [13]. Prior to this, this research direction was mainly popular in the field of natural language processing [13]. The general structure framework of it is shown in the following Fig. 4.

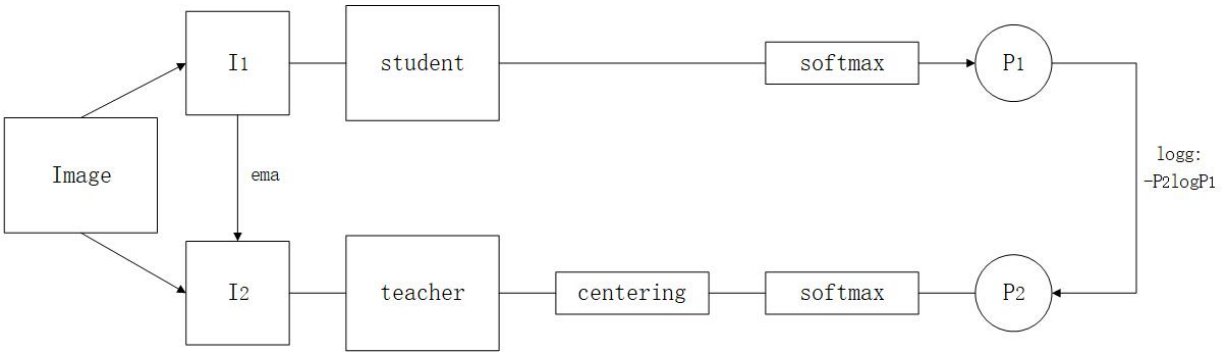

Figure 4. The structure framework of DINO [13].

Firstly, the input images are randomly cropped into two different sizes and then input into the student network and teacher network, respectively. Then, the output of the teacher network is centred. Subsequently, the output of the student network and teacher network is processed in softmax. The output of the two softmax layers will be transmitted to the loss function, and the student network will perform gradient propagation through SGD. DINO processes the parameters of the student network through an empirical moving average (ema) and passes them on to the teacher network. It is worth mentioning that the models of the student network and the teacher network must be consistent, but

parameters should set different values. In other words, these two sub-networks can use existing models, such as ResNet-50 and ViT-B/8.

In the experiment, the pre-trained model parameter ViT-L/14 was imported, and the output of the last fully connected layer was changed to 113. The optimizer used Adam. Meanwhile, the epochs is 5 and the batch size is 16.

3. Results

Fig 5 illustrates these four models' performance. During the research, there are totally four models implemented. They have slightly different accuracy in the Fruit 360 dataset, but all of them are above 0.95, which means all of them have nice performance. VGG16 model accuracy is nearly 0.952. YOLOv5 model accuracy is about 0.997. YOLOv8 model accuracy is around 0.998. DINOv2 model accuracy is about 0.986. Based on the information mentioned above, the YOLOv8 model ranks first place, which means it is the most reliable model among these models.

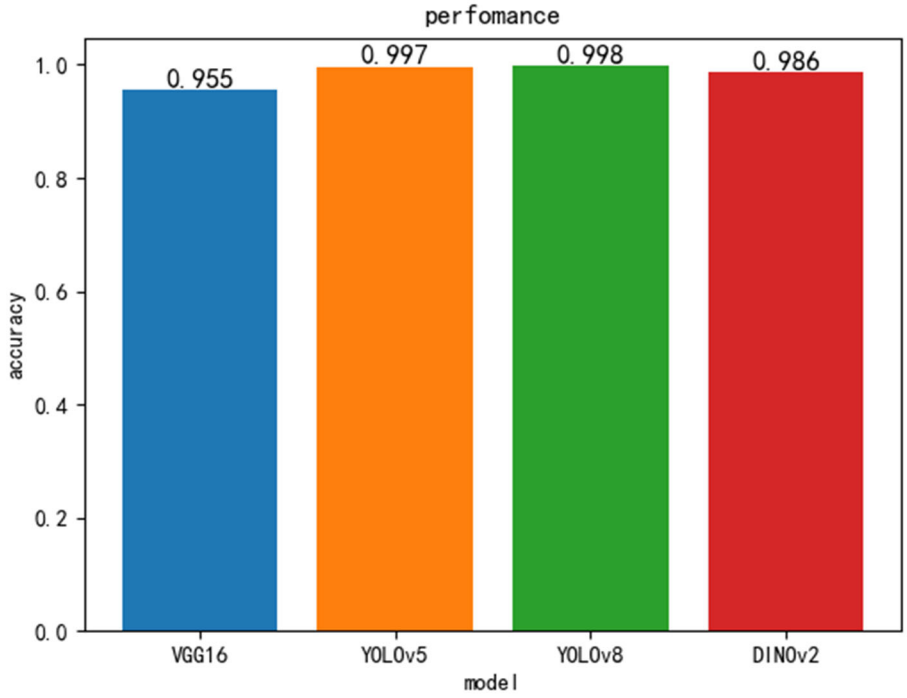

Figure 5. Bar plot of four models' performance (Photo/Picture credit: Original).

4. Discussion

The YOLO model performs better than the other two models, possibly because it uses anchor boxes to obtain corresponding objects [14]. This process makes the image referenced by the model for classification more easily distinguishable by removing some of the background. The YOLOv8 model performs better than the YOLOv5 model, possibly by replacing the coupled head with the decoupled head. The decoupled head extract's location information and category information separately, and learns through different network branches before merging. The coupled head calculates location information and classification information together [15]. For the dataset used this time, there is only one object per image, and it only focuses on classification tasks. Therefore, separate learning can ensure that classification accuracy is not affected by positional parameters. The DINOv2 model ranks third in accuracy because it adopts an attention mechanism [16]. Through the attention mechanism, the model can have a more comprehensive and clear understanding of the entire input object. The VGG16 model only uses CNN, which can merely associate local information each time, meaning that CNN has a certain receptive field. Therefore, VGG16 has the worst effect.

In the short term, when the amount of data to be processed is small, using YOLO can meet the demand. But in the long run, DINO has greater potential. Nowadays, in the era of big data, more and more information are being digitized. DINO is developed based on Transformer and is a foundation model with a deeper model structure, which means it has more parameters so that it possesses stronger learning ability. Meanwhile, DINO adopts self-supervised learning. In the future, it is unrealistic to manually annotate a large amount of data to achieve the goal of training models. Self-supervised learning means that the model can autonomously recognize and induce, which is more in line with the needs of future development.

However, there are still some drawbacks to this project. Firstly, the dataset used in this experiment is not large enough to fully utilize the performance of some models. In the future, testing is planned on the COCO dataset. Secondly, the comparison of models is not comprehensive enough, and there are still some classic models that have not been compared. In the future, Fast Region-based Convolutional Network method (Faster R-CNN) and Google Net will also be added for comparison. In addition, the comparison parameters should be more comprehensive. Thus, the next step is to include the running time in the comparison, making the results more convincing. Some advanced optimization strategies or algorithms will be also considered for improving the model's performance [17, 18].

5. Conclusion

By implementing four models and testing them on the Fruit 360 dataset, this article found that YOLO and DINO both performed well. Then, by comparing their structural characteristics, this article found that the reasons for high accuracy may come from two aspects. Firstly, the YOLO model has an additional operation of selecting anchor boxes compared to other models, while the DINO model adopts attention mechanisms. The fundamental purpose of these two designs is to increase the proportion of the reference part in the original image when the model extracts features. For example, YOLO increases the perception range during the feature extraction stage by cropping the image, while DINO directly correlates with the global when extracting features. Meanwhile, YOLOv8 has made marginal improvements to YOLOv5 by changing the structure of the head, allowing the model to focus more on classification tasks. In this study, various computer vision models rooted in distinct design concepts are compared to assess their strengths and weaknesses. The objective is to inspire the development of innovative model structures for future research. It's important to note that this research is still in its preliminary exploration phase. In the future, we aim to further our investigation by merging YOLO's anchor box algorithm with DINO's attention mechanism to craft a novel model structure tailored for image classification.

References

[1] Madhiarasan M, Massimo Tipaldi, and Pierluigi Siano. Analysis of artificial neural network performance based on influencing factors for temperature forecasting applications. Journal of High-Speed Networks 26.3, 2020, 209 - 223.

[2] LeCun Yann et al. Backpropagation applied to handwritten zip code recognition. Neural computation 1.4, 1989, 541 - 551.

[3] Simonyan Karen and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[4] Redmon Joseph et al. You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.

[5] Bommasani Rishi, et al. On the opportunities and risks of foundation models. arXiv preprint arXiv: 2108. 07258, 2021.

[6] Koohfar Sahar, Wubeshet Woldemariam, and Amit Kumar. Performance Comparison of Deep Learning Approaches in Predicting EV Charging Demand. Sustainability 15.5, 2023, 4258.

[7] Kaggle Fruits 360 https://www.kaggle.com/datasets/moltean/fruits/data, 2021.

[8] Zhong Guoqiang, Xiao Ling and Li-Na Wang From shallow feature learning to deep learning: Benefits from the width and depth of deep architectures. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9.1, 2019, e1255l.

[9] Cheng Yu et al. A survey of model compression and acceleration for deep neural networks. arXiv preprint arXiv:1710.09282, 2017.

[10] Wang Huaqing et al. An enhanced intelligent diagnosis method based on multi-sensor image fusion via improved deep learning network. IEEE Transactions on Instrumentation and measurement 69.6, 2019, 2648 - 2657.

[11] Yang Junqing, Peng Ren and Xiaoxiao Kong. Handwriting text recognition based on faster R-CNN. 2019 Chinese Automation Congress (CAC). IEEE, 2019.

[12] Wang Xinning, et al. Data-attention-YOLO (DAY): A comprehensive framework for mesoscale eddy identification. Pattern Recognition 131, 2022, 108870.

[13] Caron Mathilde, et al. Emerging properties in self-supervised vision transformers. Proceedings of the IEEE/CVF international conference on computer vision, 2021.

[14] Jiang Peiyuan, et al. A Review of Yolo algorithm developments. Procedia Computer Science 199, 2022, 1066 - 1073.

[15] Lou Haitong, et al. DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor. Electronics 12.10, 2023, 2323.

[16] Oquab Maxime et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv: 2304. 07193, 2023.

[17] Qiu Yuhang et al. Pose-guided matching based on deep learning for assessing quality of action on rehabilitation training. Biomedical Signal Processing and Control 72, 2022, 103323.

[18] Bayoudh Khaled, Fayçal Hamdaoui, and Abdellatif Mtibaa. Hybrid-COVID: a novel hybrid 2D/3D CNN based on cross-domain adaptation approach for COVID-19 screening from chest X-ray images. Physical and engineering sciences in medicine 43, 2020, 1415 - 1431.