Applied Intelligence
https://doi.org/10.1007/s10489-023-04799-8

Fruit ripeness identification using transformers

Bingjie Xiao[1] · **Minh Nguyen**[1] · **Wei Qi Yan**[1]

Accepted: 14 June 2023

Abstract
Pattern classification has always been essential in computer vision. Transformer paradigm having attention mechanism with global receptive field in computer vision improves the efficiency and effectiveness of visual object detection and recognition. The primary purpose of this article is to achieve the accurate ripeness classification of various types of fruits. We create fruit datasets to train, test, and evaluate multiple Transformer models. Transformers are fundamentally composed of encoding and decoding procedures. The encoder is to stack the blocks, like convolutional neural networks (CNN or ConvNet). Vision Transformer (ViT), Swin Transformer, and multilayer perceptron (MLP) are considered in this paper. We examine the advantages of these three models for accurately analyzing fruit ripeness. We find that Swin Transformer achieves more significant outcomes than ViT Transformer for both pears and apples from our dataset.

Keywords Visual Object Detection · Vision Transformer · Swin Transformer · Mask R-CNN · MLP

1 Introduction

In recent years, deep learning has been increased exponentially, with a profession of breakthroughs in theory and architecture [1]. As a branch of deep learning, visual object detection from digital images has also achieved great outcomes in development. Visual object detection is essentially with classification problem [2]. So far, visual object detection has been able to accurately locate and identify multiple targets. Before using deep learning for visual object detection, conventional machine learning algorithms usually have three stages: Region selection, feature extraction, and classification. Traditional algorithms usually take use of sliding window algorithms, but the algorithms have a huge number of redundant bounding boxes, correspondingly computational complexity is high.

Visual object detection [3, 4] usually refers to detect the location of a visual object in an image and assign the label of corresponding class. The detector is required to output 5-tuple: Label of object class, the coordinates of the four corners of the bounding box [5].

The motivation of this article stems from the news that we usually lack professional labors picking up fruits in mature season. In absence of workers, how to efficiently complete fruit selection and pickup is a problem. In this paper, we chose apples and pears as the representatives of fruits, implemented the classification of fruits by classifying the maturity of different fruits [6].

The purpose of this paper is to locate and classify fruits in the given images [7]. As the name implies, the model is required to accurately locate fruits in the given image and identify whether the fruit is ripe or overripe [8, 9].

Compared with Region Proposal Network (RPN) of Faster R-CNN (region-based CNN) [10, 11], Transformer is completely based on self-attention mechanism [12, 13]. The complexity of a Transformer model ensures that the accuracy of the model is higher than that of R-CNN net. Therefore, we employed Transformer model to conduct our experiments. According to the characteristics of Transformer model in visual object detection, we select Swin Transformer [14] and Vision Transformer (ViT) for experimental evaluations. During our experimenting, we found that the MLP block is an integral part of the Transformer model. Therefore, in this paper, MLP-based object detection model is implemented with axial displacement for comparative experiments.

As shown in Fig. 1, visual object detection using a bounding box to mark fruits and label the classes is implemented,

✉ Wei Qi Yan
wyan@aut.ac.nz

[1] Auckland University of Technology, Auckland 1010, New Zealand

Published online: 29 June 2023

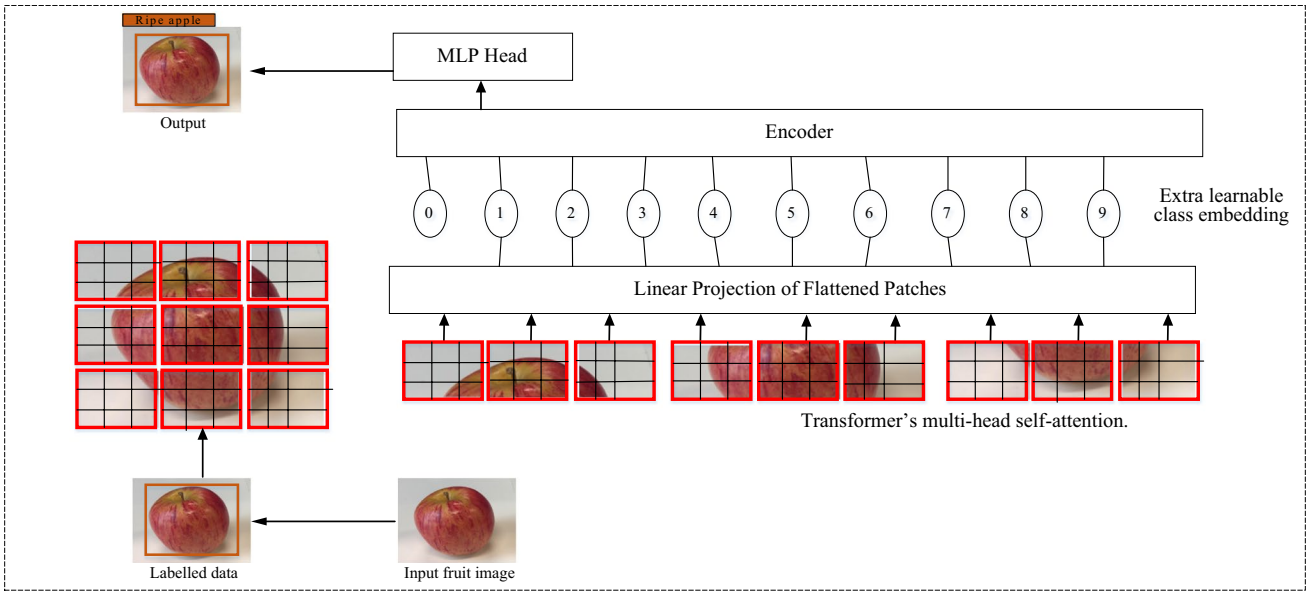


Fig. 1 The flowchart of object detection using Vision Transformer

the coordinates are sent to Transformer model for the process of encoding and decoding [15]. In the experiments, we analyze the difference between attention mechanisms of ViT and Swin Transformer, introduce Mask R-CNN weights to get the best accuracy. Overall, the contributions of this paper are:

(1) We created our own dataset and adopted Swin Transformer model to achieve fruit object detection and achieve 87.43% precision.
(2) We combined Transformer module with YOLO module together to achieve accurate classification of fruits, so that the model can distinguish the maturity of apples or pears.

In the second part of this paper, we depict previous research work related to Transformers including embedding, attention mechanism, and MLP. In the third part, we will detail our proposed models and related work. The results are then presented in the fourth section. Our conclusion regarding the experiments and inspiration for future work will be drawn in Section 5.

2 Literature review

2.1 Fruit recognition

Fruit recognition in deep learning is use of mathematical models to detect the position and class label of fruits based on given digital images. Fruit recognition based on pixel intensities is an initial idea [16]. R-CNN model extracts region of interest for locating fruits [17]. Segmentation provides image regions of interest for wide selection [18].

In practical applications of fruit recognition, visual object has a small area in the image, or mutual occlusion between the targets [19, 20]. For example, in fruit images we have collected, we have clearly separated pears from the images, there may also be dense apples piled on the trees that are difficult to be distinguished. Faster R-CNN takes use of the overlapping ground truth and predicted bounding boxes to achieve the detection of small objects [21]. Fruit surface disease detection [22] is associate with fruit ripeness detection. In actual experiment, we locate the position of fruits in the input image and determine the located fruit class. Finally, we analyze the ripeness of fruits by using the fruit appearance in the input images.

Traditional feature extraction methods were applied to identify diseases of fruits such as tomato [23]. Wu et al. experimented two Transformers to obtain feature information, and took advantage of patches of multiple resolutions for multi-granularity feature extraction. Jia et al. improved DenseNet by using residual network (ResNet), optimized the training parameters, and made the model identify apples with an accuracy rate of 97.31% [24]. Regarding the Transformers, more encoding modules are encapsulated to extract effective feature information. Therefore, in this paper, a multi-level attention feature extraction module was created. Compared with visual features that CNN can capture, the Transformers can identify details.

2.2 Transformer and mask R-CNN model

Transformer is based on self-attention mechanism, which has virally spanned in the field of Natural Language Processing (NLP). Multilayer Perceptron (MLP) is the earliest and simplest neural network in the NLP. In order to handle more complex problems, the mainstream architecture of artificial neural networks has undergone the evolution of MLP-CNN [25, 26] and recurrent neural network (RNN).

Similar to the research work in fruit recognition, the project was initialized with Swin Transformer [27]. Similar to the models that automatically recognize pests encountered in rice growth, an experiment essentially was conducted that the model can identify the maturity of fruits, and ultimately achieved the goal of agricultural automation. Sliding windows of Swin Transformer model were taken advantage for hierarchical design, which achieved the accuracy 93.4%.

Han, et al. also studied the use of Transformer model to realize the control of machines [28]. The robot grasping frame takes use of tactile and visual information to achieve safe grasping of visual objects. Similar to our experiments, Han's team also made use of the characteristics of predefined objects to perform training on the Transformer model by comparing with CNN + LSTM model.

Small object detection by using deep learning has been taken into account in practice [29]. Transformer and CNN models were employed for local perception network of Swin Transformer, a Spatial Attention Interleaved Execution Cascade (SAIEC) network was designed to enhance the segmentation of digital images. The final model is 1.7% more than the base Swin Transformer network. The multi-perceptual design of Transformer model outperforms residual network to realize the cross-channel transfer of each feature of visual object [30]. If the data is massively enhanced or distillable, the Transformer model does not need to make global adjustments to the convolutional layer, maximum pooling layer, and global average pooling layer (GAP) like the CNN model [31] or Mask R-CNN model [32].

The mainstream algorithms [33] of visual object detection were explored. Unlike CNN, which completes the extraction of local image information and constructs global information by stacking convolutional layers, Transformer models obtain complete global information from the beginning, it has stronger long-term dependence. In the ViT model, the average attention distance increases from small to large with the deepening increases of the layers, which has a similar paradigm of CNN [34]. In ViT, if the scaling ability of the Transformer is stronger, the transmission effect will be better [35]. However, because the Transformer does not have bionic characteristics like CNN, in the learning process, the training set of the Transformer model needs to be enhanced or the number of datasets can be increased so as to acquire better results.

Mask R-CNN was combined with ResNet-50 to detect wheat diseases and achieved an accuracy 88.19% [36]. In Mask R-CNN model, ResNet-50 was employed to extract RPN and generate various anchors. During anchor box extraction, mask loss and bounding box loss are taken into consideration. RPN generated a binary mask for each visual object. Anchor box regions were applied to ROI alignment features [37]. After ROI alignment, fully-connected layers are employed for bounding boxes regression and classification, each object is detected by using a mask-form convolutional layer.

An axially displaced AS-MLP architecture [38] was employed to encode global spatial features. In the experiments, the axial displacement of the feature maps enabled MLP model to achieve the same function of local feature extraction as CNN architecture. The MLP-Mixer [39] pays much attention to the changes of the ViT based on the MLP architecture. MLP-Mixer splits the image into multiple non-overlapping patches, and then takes use of the fully connected layer to convert each patch into feature embedding and sends it to the mixer layer. The MLP-Mixer model can be understood to replace the blocks of ViT with the Mixer layer.

Inspired by the previous work, we adopted Swin Transformer combined with Mask R-CNN [40] and ViT model to achieve fruit ripeness classification [41–43]. We also have the MLP object detection model to compare with MLP block in the Transformer model.

3 Methodology

3.1 Fruit ripeness identification

Fruit ripeness identification is essentially an object detection task. The object detection task is to describe the whole input image as the content, and then detect the specified object. As shown in Fig. 2, the input is a given image. We are use of bounding boxes to segment the fruit of interest from the background and determine the class and location. After the model has been trained along with the Transformer model, the output is a list, each item in the list includes the fruit class and location of the detected object.

3.2 Swin transformer

Swin Transformer has four stages in Fig. 2, each of which is similar. As shown in Fig. 3, the red box represents a window to perform self-attention, and the black box shows each patch. The input size of the image for the Swin Transformer is $W \times H \times C$, and the image is grouped into a patch collection of $\frac{H}{4} \times \frac{W}{4}$ by using 4×4 patch. The first stage of Swin model is to use a linear embedding and convert

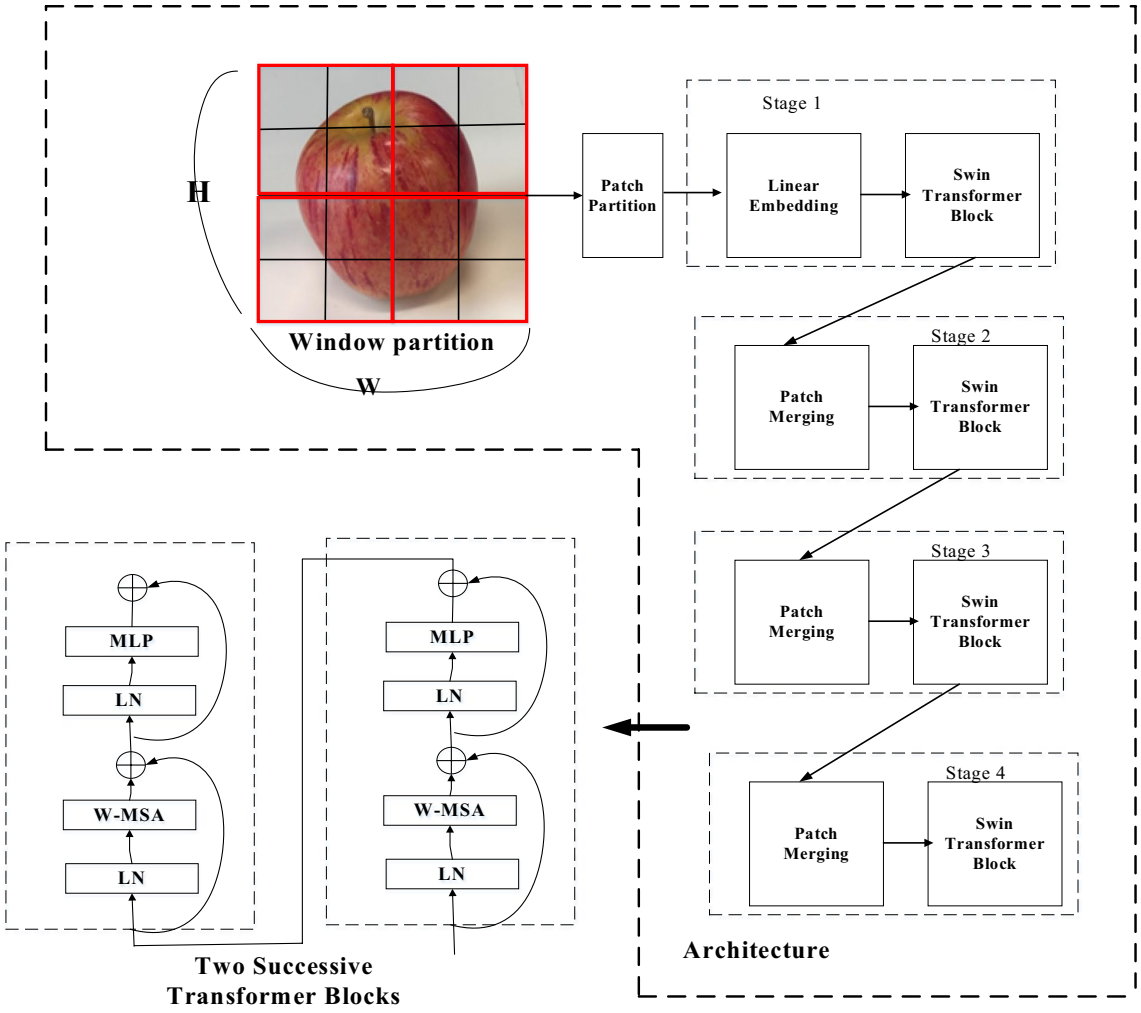


Fig. 2 The architecture of Swin Transformer

the input patch features into C, then send them to a Swin Transformer block. Stages 2 to 4 are the same, using a patch merging to merge adjacent patches and feed them into the next Swin Transformer block. As shown in Fig. 2, the role of patch merging is to complete the down sampling of features.

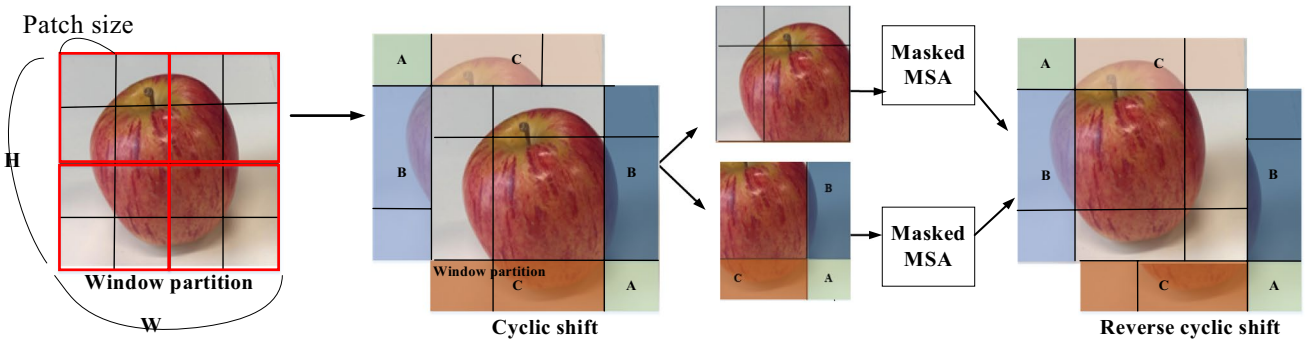


Fig. 3 The shift window of Swin Transformer based on MSA

The original feature size of Swin Transformer is $[H_1, W_1, C_1]$. Window partition progress is based on the original size of the reshape, the size is,

$$Reshape\ Size = [\frac{H_1 \times W_1}{window_{size} \times window_{size}}, window_{size}, window_{size}, C_1] \tag{1}$$

The Swin Transformer Block is characterized by using a shift window to replace the standard Multi-head Self-Attention (MSA) module. The attention of Swin Transformer is,

$$Attention(Q, K, V) = SoftMax\left(\frac{QK^T}{\sqrt{d}} + B\right)V \tag{2}$$

where B stands for position code, Q means query vector, K represents a vector representing the queried information, V shows queried information of vector. The variance is d.

Mask R-CNN is a two-stage framework [44, 45]. In the first stage, the proposals are generated. In the second stage, the proposals are classified, bounding boxes and masks are generated. Mask R-CNN includes FPN to solve the degrade of training process. FPN adopts the top-down structure and horizontal connection to conduct the fusion of the feature map from the bottom to the top, which can implement fast connection and extraction of all scales. FPN is also a sliding window with a fixed window size. Feature extraction is conducted through the backbone network, the generated feature map is input into the Region Proposal Network (RPN) for sub-network selection.

In Fig. 4, in order to implement the intersection of the upper window partition of each block, Swin Transformer adds 3×3 shift window to 2×2 window to improve feature transfer. Since the size of shift window is not the same, the shift processing is fulfilled. The cyclic shift modifies the size from 3×3 to 2×2, then the reverse cyclic shift is conducted according to the attention model so as to obtain the shift window attention [46, 47]. The 3×3 window feature map is shifted and becomes a 2×2 window, but the actual calculation is still expected to be carried out in 3×3 windows, that is, the results of 9 attentions are implemented with the help of masks.

Transformer self-attention is set up by using a specific mask. While performing attention analysis, only the effective part in one window is calculated, and the rest is masked. The original calculation method of attention can be changed. The shaded area B shown in Fig. 3 is the part that needs to be masked out.

Window-based local self-attention (W-MSA) segments the input image into non-overlapping windows, and conducts self-attention calculations in different windows. Assume that an image has $h \times w$ patches, each window contains $M \times M$

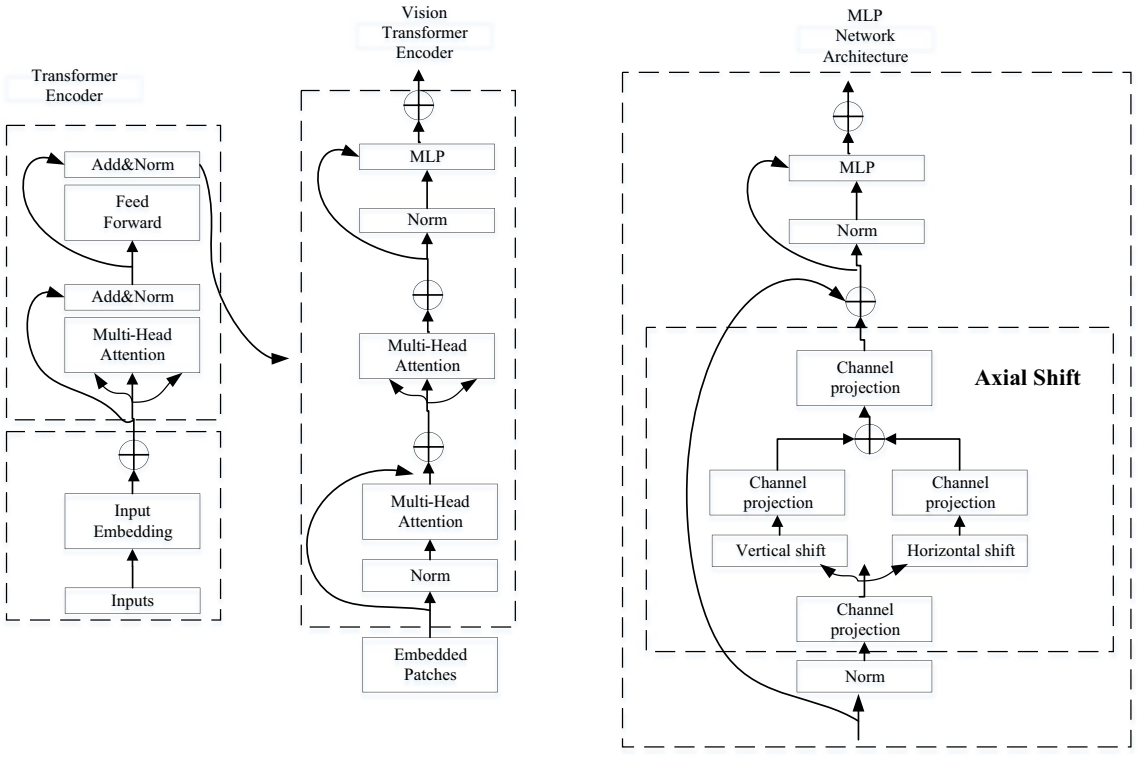


Fig. 4 The difference between Transformer and MLP methods

regions, then the computational complexities of MSA and W-MSA are respectively shown as:

$$\Omega(MSA) = 4whC^2 + 2(hw)^2C \tag{3}$$

$$\Omega(W-MSA) = 4whC^2 + 2M^2hwC \tag{4}$$

3.3 Vision transformer

Transformer is employed in natural language processing [48, 49]. The attention mechanism of Transformer is also broadly employed, such as Se module, CBAM module and other attention modules, these attention modules can improve network performance. The ViT model demonstrates that a structure that does not rely on CNNs can achieve perfect results for image classification, which is also very suitable for transfer learning. The ViT blocks of the original image are input into the encoder of the original Transformer model, and finally a fully-connected layer is applied to classify the image.

As shown in Fig. 1, ViT model is mainly composed of three modules: 1) Linear projection (i.e., Embedding layer of patch + position); 2) Transformer encoder; 3) MLP head (i.e., classification layer). The Transformer encoder module inputs the patch shown in the black box in Fig. 4. In ViT Transformer, each small image is regarded as a token (representing a word in NLP), and the correlation between each token is calculated in the model.

In Fig. 5, the relative encoding of Swim Transformer model is mainly to solve the problem of arrangement invariance in self-attention, that is, tokens input in different orders will get the same result. In ViT, it is not enough to just split the image into small patches. What the encoder module needs is a vector with a shape as $[num\_token, token\_dim]$. For the input image data, the shape $[H, W, C]$ does not meet the input requirements, so it is necessary to convert the image data into tokens through the embedding layer. Transformer encoder module is to stack the encoder block several times, mainly composed of the following parts:

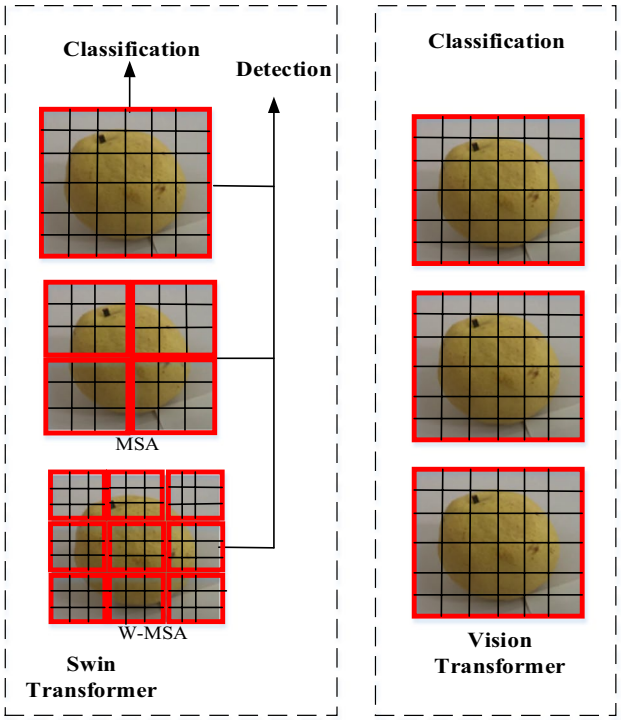


Fig. 5 Swin Transformer and Vision Transformer calculate the self-attention of regions in non-overlapping windows

1) Layer normalization

Layer normalization is to calculate the mean and variance of all feature maps of the sample, and then normalize it. ViT also splits the input image into patches. The process of using patch embedding is to compress each patch into a vector with a dimension through a fully-connected network.

2) Multi-head attention

$$MultiHead(Q, K, V) = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{5}$$

In Fig. 4, ViT takes use of self-attention to express the relationship between each patch and other patches. ViT generates q,

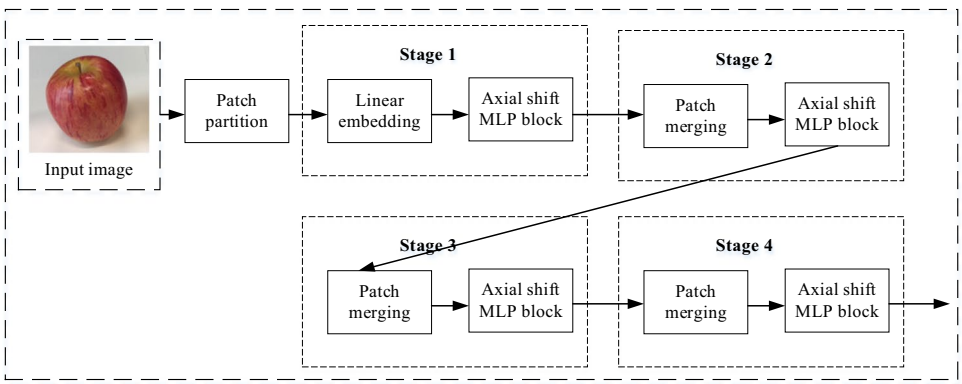


Fig. 6 Horizontal displacement process of MLP object detection

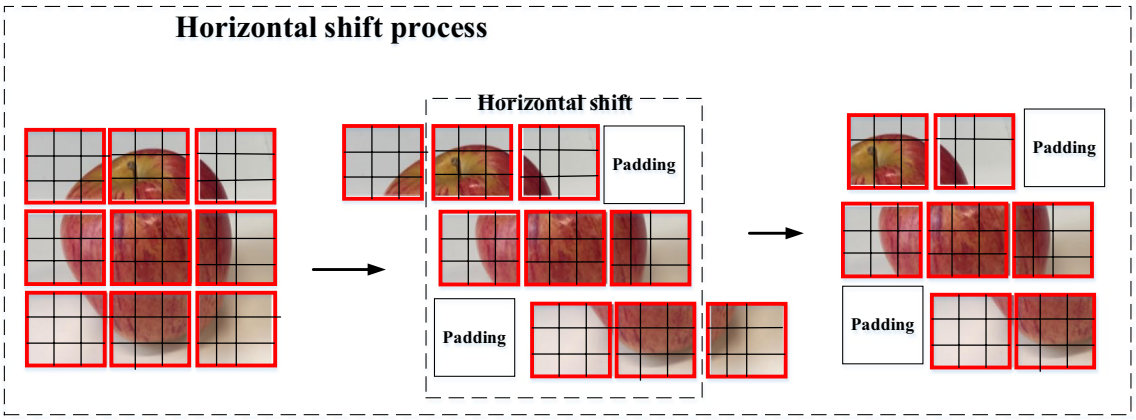


Fig. 7 Horizontal displacement process of MLP object detection

k, and *v*, it integrates *q*, *k*, and *v* into *num_heads*, and then performs self-attention operations on each of them, finally merges them together. Multi-head self-attention isolates parameters and can better focus associated features together for training.

3) MLP block

A MLP block is an inverted bottleneck structure consisting of connected layer, GELU activation function and DropOut. It should be noted that there is no decoder module in ViT Transformer. Therefore, there is no need to calculate the cross-attention value of encoder and decoder.

3.4 MLP object detection mechanism

Multi-layer perceptron (MLP) neural network is a kind of neural networks that are use of a combination of multiple perceptron to implement the segmentation and transmission of feature information. MLP neural network consists of an input layer, multiple hidden layers and an output layer. All neurons in an MLP are similar, each neuron has a number of inputs that connect to the previous layer and output neurons that connect to the next layer. Each neuron will pass the same value to multiple connected output neurons.

MLP network has less inductive bias, so MLP-based backbone can achieve visual object detection [50, 51]. The structures shown in Fig. 7 and Fig. 8 indicate the process of axial displacement using MLP object detection method.

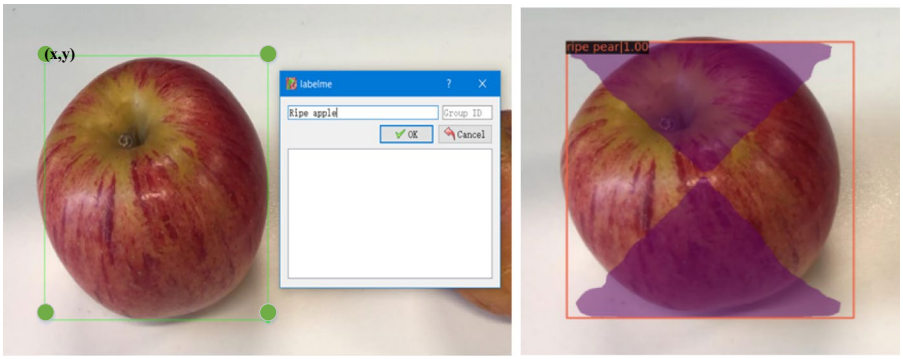
Figure 6 shows the four stages of MLP. Similar to the Transformer model shown in Fig. 3, the MLP object detection model also splits the original image into multiple 4×4 patches.

As shown in the Fig. 7, MLP is an operation of a local receptive field, which is more suitable for extracting features with local dependencies. For the fruit image in the experiment, the position of the apple is based on the position of the given output, so MLP relies on the weighting of local features to better extract local features.

4 Results

4.1 Dataset and parameter settings

We collected apple and pear datasets by using our phone cameras, we intuitively compare the learned parameters with the parameters of real models. Figure 8 is the input image from our dataset. We labelled our samples with software tool LabeleMe. We assign the number of samples in



(a) A sample of input images. (b) A sample of output images.

Fig. 8 Samples of the image dataset. (**a**) A sample of input images, (**b**) A sample of output images

Table 1 The results of precisions by training ViT model

Model	Epoch	Weights	AP@0.5:0.95
Vision Transformer	10	vit_base_patch_16	0.4560
		vit_base_patch_32	0.4060
		vit_large_patch_16	0.4560
		vit_large_patch_32	0.4120
	20	vit_base_patch_16	0.4310
		vit_base_patch_32	0.4310
		vit_large_patch_16	0.3560
		vit_large_patch_32	0.4250
	30	vit_base_patch_16	0.4000
		vit_base_patch_32	0.4190
		vit_large_patch_16	0.3880
		vit_large_patch_32	0.3880
	50	vit_base_patch_16	0.3810
		vit_base_patch_32	0.3940
		vit_large_patch_16	0.4060
		vit_large_patch_32	0.4000

the training set as 2,000, set the conventional parameter batch size to 1.0 and the learning rate to 0.0001 according to the computer configuration for training. The size of images is 1920×1080.

In Fig. 8 (a), the green square box is bounding box, the four points represent the coordinates of the bounding box (x, y, w, h). Figure 8 (b) shows the predicted result by using the trained detector. The experiment is analyzed with the accuracy of the model through observing the changes of iteration parameters. A diversity of fruits is defined as different classes, and the same fruit defines ripeness according to the smoothness of the skin. A smooth peel is defined as the class "Ripe", a folded or decayed surface is defined as the class "Overripe". In multiclass classification, each class can be drawn as a curve according to recall and precision rate. The average precision is the area under the curve, the mean average precision refers to average the AP of each category. We take use of the value of mean average precision (mAP) to evaluate the quality of the proposed model.

4.2 Result analyze

We took use of two scales of ViT models: Base and large. *vit_base_patch_16* represents the ViT base model; the size of image patch is 16×16. *vit_large_patch_32* means that the ViT large model is applied, and the image patch size is 32×32. In Table 1, the ViT model does not show better performance. More iterations did not achieve better results, ViT model did not perform well in the trade-off between small datasets and large datasets. The ViT model is usually pre-trained based on large datasets. Compared with the ViT model, CNN can perform better in small datasets.

Although the accuracy of the ViT model in Table 1 is not high, in the training process, the model takes use of less computing resources that can better allocate computing resources. ViT performs structural pruning on the Transformer model, and then quantizes the pruned model to obtain the final optimized model. However, during the pruning process, the ViT model requires an additional training process, which limits the practicability of the model. Although the memory usage and execution time are reduced in the process of ViT model pruning, it cuts off the accuracy of this model.

In Table 2, MLP object detection model has very strong performance in small models. MLP model pays much attention to local feature extraction, but if the model capacity is expanded, there will be overfitting problems. The overfitting problems will lead to a roadblock to the success of MLP. The self-attention structure of the ViT model also includes the MLP block. Self-attention is related to a sequence, which mainly emphasizes that each position of the sequence has the same set of MLP parameters, and then conducts a weighted average operation in the new space. The MLP model is a nonlinear mapping. We see from Table 1 and Table 2 that the MLP model can better capture the features of the model.

Different from ViT and MLP, Swin Transformer's self-attention calculation based on moving window ensures that the model can extract more features of visual objects.

We chose three weights to train the Swin Transformer model. In Table 3 and Table 4, the patch can make up four windows after moving, it is impossible for the patch to slide

Table 2 The results of MLP by training Mask R-CNN small weights

Model	Epoch	Weights	AP50	AP@0.5:0.95	Average inference time(seconds)
MLP using Axial shift MLP block	10	mask_rcnn_small_patch4_1x	0.9450	0.8310	0.5850
	30		0.9560	0.8430	0.5370
	50		0.9600	0.8470	0.5400
	10	mask_rcnn_tiny_patch4_1x	0.9330	0.8270	0.3820
	30		0.9550	0.8440	0.3850
	50		0.9580	0.8440	0.3760

Table 3 The results of Swin Transformer by training Mask R-CNN small weights

Model	Epoch	Weights	AP50	AP@0.5:0.95
Swin Transformer	10	mask_rcnn_small_patch4_1x	0.9390	0.8210
	20		0.9400	0.8230
	30		0.9480	0.8300
	50		0.9510	0.8390
	10	mask_rcnn_small_patch4_3x	0.8340	0.6810
	50		0.9460	0.8350

through each window, the mask is employed to contact and calculate the attention in each window. Therefore, Swin Transformer is a hierarchical representation that has the ability to perform complex linear calculations.

In Swin Transformer, shifting the window segmentation results in more windows, and leads to a large number of computations while filling smaller windows into larger ones. By setting a reasonable mask, shifted windows achieve equivalent calculation results under the same number of windows as window attention. We observe that Swin Transformer achieves better training results. In contrast, larger weights and more iterations allow the model to achieve better results.

MLP pays attention to feature transfer. The MLP model is use of axial displacement to arrange the features of spatial positions in the same position, so that the model can obtain local dependencies, the model can achieve performance comparable to that of the Transformer model. However, Eq. (3) and (4) show how many computations are required for MSA and WMSA. The complexity of MSA is related to $(h \times w)^2$, and the complexity of W-MSA is related to $(h \times w)$. Therefore, the amount of W-MSA calculation will be small. If the original image is large, the amount of W-MSA calculation has obvious advantages. Hence, in Table 2, Table 5, and Table 6, we observe more intuitively that experiments with the Swin Transformer module can achieve faster speeds.

As shown in Table 5 and Table 6, we harnessed different frameworks to implement the Swin Transformer. We see that with the increase of iterations, the results of classification training gradually become better and tend to be stable. Conventional Transformers take use of pre-normalization at the beginning of each residual branch, which normalizes the magnitude of the input and has not restrictions on the output. Under pre-normalization, the output activation values of each residual branch are directly merged back into the main branch and accumulated layer by layer, so the amplitude of the main branch increases with depth.

Swin Transformer takes use of residual-post-normalization. The normalization layer was moved from the beginning to the end of each residual branch, so that the output of each residual branch is normalized before being merged back into the main branch, as the number of layers deepens, the magnitude of the main branch will not be accumulated.

4.3 Discussion

The advantage of our experiment lies in the better accuracy achieved. The Swin Transformer model reached an average precision of 87.43%. Our model is able to accurately locate an apple or pear in the input image and tell us whether the current fruit belongs to the class “Ripe” or “Overripe”. At the same time, the model also achieves fast and accurate recognition within 0.13 s.

Similar to our experiments, an average accuracy of 89.3% was achieved for Kiwifruit detection [7], while an average accuracy of 88.45% was obtained for banana detection, respectively. Compared with previous experiments, the weakness of our experiment lies in the fact that in practical applications, the influence of the noise generated by the environment of different fruits on the collection of data sets should be more considered. At the same time, we should consider changing more kinds of pixels in the dataset to simulate the actual growth environment of the fruit during the fruit picking process.

5 Conclusion and future work

In our experiments related to fruit ripeness classification, we found that for small targets and small datasets, the Swin Transformer model showed its advantages and

Table 4 The results of Swin Transformer by training Mask R-CNN tiny weights

Model	Epoch	Weights	AP50	AP@0.5:0.95
Swin Transformer	10	mask_rcnn_tiny_patch4_1x	0.9360	0.8090
	20		0.9380	0.8230
	30		0.9450	0.8220
	50		0.9450	0.8230

Table 5 The results of each class by training Swin Transformer and YOLO module with small Mask R-CNN weights

Model	Weights	Epoch	Class	mAP	Average inference time(seconds)
YOLOX + Swin Transformer	mask_rcnn_small_patch4_3x	10	Ripe apple	0.0000	0.1217
			Over apple	0.2200	
			Ripe pear	0.0200	
			Overripe pear	0.4600	
		20	Ripe apple	0.0000	0.1210
			Over apple	0.0060	
			Ripe pear	0.0860	
			Overripe pear	0.4340	
		30	Ripe apple	0.7867	0.1205
			Over apple	0.4687	
			Ripe pear	0.8404	
			Overripe pear	0.8416	
		50	Ripe apple	0.8889	0.1212
			Over apple	0.6695	
			Ripe pear	0.8856	
			Overripe pear	0.9127	

accuracy. We have implemented the classification of fruits of different maturity, and the model can be practically applied in warehouse management, agricultural automatic picking, etc.

The actual results show that in the ViT Transformer, the CNN responding to edges is weak. CNN can only compute correlations with adjacent pixels. Due to the characteristics of sliding window convolution, non-domain pixels cannot be jointly calculated, which makes spatial information unusable. Swin Transformer can provide hierarchical feature representation, self-attention based on moving window can effectively achieve feature extraction.

In future, we will further utilize the unique self-attention mechanism of Vision Transformer to capture the pixel information between tokens to ensure that ViT model can obtain pretty rich features with the same parameters [1, 52].

Table 6 The results of each class by training Swin Transformer moudel and tiny Mask R-CNN weights

Model	Weights	Epoch	Class	mAP	Average inference time(seconds)
YOLOX + Swin Transformer	mask_rcnn_tiny_patch4_3x	10	Ripe apple	0.1978	0.1288
			Over apple	0.0100	
			Ripe pear	0.0000	
			Overripe pear	0.0061	
		20	Ripe apple	0.4142	0.1300
			Over apple	0.1392	
			Ripe pear	0.1833	
			Overripe pear	0.6463	
		30	Ripe apple	0.8702	0.1320
			Over apple	0.8270	
			Ripe pear	0.8322	
			Overripe pear	0.8426	
		50	Ripe apple	0.8791	0.1334
			Over apple	0.8292	
			Ripe pear	0.8909	
			Overripe pear	0.8981	

Authors contribution statement All authors contributed to this paper equally.

Funding and/or Conflicts of interests/Competing interests Open Access funding enabled and organized by CAUL and its Member Institutions All authors agreed with the content to submit. This paper has not relevant information regarding sources of funding, financial or non-financial interests.

Data availability and access The data appeared in this paper is available upon request.

Declarations

Ethical and informed consent for data used No ethical data in this paper.

Competing interests No conflict of interests in this paper that are directly or indirectly related to the work submitted for publication.

References

1. Yan W (2021) Computational methods for deep learning: theoretic, practice and applications. Springer Cham
2. Zhu X, Lyu S, Wang X, Zhao Q (2021) TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In: IEEE/CVF International Conference on Computer Vision, pp 2778–2788
3. Lee D, Kim J, Jung K (2021) Improving object detection quality by incorporating global contexts via self-attention. Electronics 10(1):90
4. Qi J, Nguyen M, Yan W (2022) Small visual object detection in smart waste classification using transformers with deep learning. International Conference on Image and Vision Computing (IVCNZ), Auckland. https://link.springer.com/chapter/10.1007/978-3-031-25825-1_22
5. Zhang R, Li X, Zhu L, Zhong M, Gao Y (2021) Target detection of banana string and fruit stalk based on YOLOv3 deep learning network. In: IEEE International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), IEEE, pp 346–349
6. Fu Y, Nguyen M, Yan W (2022) Grading methods for fruit freshness based on deep learning. Springer Nature Computer Science
7. Fu L, Feng Y, Majeed Y, Zhang X, Zhang J, Karkee M, Zhang Q (2018) Kiwifruit detection in field images using Faster R-CNN with ZFNet. IFAC-Papers OnLine 51(17):45–50
8. Femling F, Olsson A, Alonso-Fernandez F (2018) Fruit and vegetable identification using machine learning for retail applications. In: International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp 9–15
9. Kuznetsova A, Maleva T, Soloviev V (2020) Using YOLOv3 algorithm with pre-and post-processing for apple detection in fruit-harvesting robot. Agronomy 10(7):1016
10. Gao F, Fu L, Zhang X, Majeed Y, Li R, Karkee M, Zhang Q (2020) Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. Comput Electron Agric 176:105634
11. Wang Q, Qi F (2019) Tomato diseases recognition based on Faster R-CNN. In: International Conference on Information Technology in Medicine and Education (ITME), pp 772–776
12. Ding M, Xiao B, Codella N, Luo P, Wang J, Yuan L (2022) DaViT: Dual attention Vision Transformers. ECCV
13. Hua X, Wang X, Rui T, Zhang H, Wang D (2020) A fast self-attention cascaded network for object detection in large scene remote sensing images. Appl Soft Comput 94:106495
14. Zheng H, Wang G, Li X (2022) Swin-MLP: A strawberry appearance quality identification method by Swin transformer and multi-layer perceptron. J Food Meas Charact:1–12
15. Ji Y, Zhang H, Zhang Z, Liu M (2021) CNN-based encoder-decoder networks for salient object detection: A comprehensive review and recent advances. Inform Sci 546:835–857
16. Jimenez AR, Ceres R, Pons JL (2000) A survey of computer vision methods for locating fruit on trees. Transact ASAE 43(6):1911
17. Shalini K, Srivastava AK, Allam S, Lilaramani D (2021) Comparative analysis on deep convolutional neural network models using PyTorch and OpenCV DNN frameworks for identifying optimum fruit detection solution on RISC-V architecture. In: IEEE Mysore Sub Section International Conference (MysuruCon), pp 738–743
18. Hameed K, Chai D, Rassau A (2022) Score-based mask edge improvement of Mask R-CNN for segmentation of fruit and vegetables. Expert Syst Appl 190:116205
19. Song H, Sun D, Chun S, Jampani V, Han D, Heo B, Yang MH (2022) ViDT: an efficient and effective fully Transformer-based object detector. ICLR
20. Tu S, Pang J, Liu H, Zhuang N, Chen Y, Zheng C, Xue Y (2020) Passion fruit detection and counting based on multiple scale Faster R-CNN using RGB-D images. Precis Agricult 21(5):1072–1091
21. Behera SK, Rath AK, Sethy PK (2021) Fruits yield estimation using Faster R-CNN with MIoU. Multimed Tools Appl 80(12):19043–19056
22. Wang H, Mou Q, Yue Y, Zhao H (2020) Research on detection technology of various fruit disease spots based on Mask R-CNN. In IEEE International Conference on Mechatronics and Automation (ICMA), pp 1083–1087
23. Wu S, Sun Y, Huang H (2021) Multi-granularity feature extraction based on vision transformer for tomato leaf disease recognition. In International Academic Exchange Conference on Science and Technology Innovation (IAECST), pp 387–390. IEEE
24. Jia W, Tian Y, Luo R, Zhang Z, Lian J, Zheng Y (2020) Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. Comput Electron Agric 172:105380
25. Benz P, Ham S, Zhang C, Karjauv A, Kweon I (2021) Adversarial robustness comparison of vision transformer and MLP-mixer to CNNs. BMVC
26. Yu T, Li X, Cai Y, Sun M, Li P (2021) Rethinking token-mixing MLP for MLP-based vision backbone. BMVC
27. Zhang Z, Gong Z, Hong Q, Jiang L (2021) Swin Transformer based classification for rice diseases recognition. In: EEE International Conference on Computer Information Science and Artificial Intelligence (CISAI), pp 153–156
28. Han Y, Yu K, Batra R, Boyd N, Zhao T, She Y, Hutchinson S, Zhao Y (2021) Learning generalizable vision-tactile robotic grasping strategy for deformable objects via transformer. https://arxiv.org/abs/2112.06374

29. Xu X, Feng Z, Cao C, Li M, Wu J, Wu Z, Ye S (2021) An improved Swin Transformer-based model for remote sensing object detection and instance segmentation. Remote Sens 13(23):4779
30. Touvron H, Bojanowski P, Caron M, Cord M, El-Nouby A, Grave E, Jégou H (2023) ResMLP: Feedforward Networks for image classification with data-efficient training. IEEE Transactions on Pattern Analysis and Machine Intelligence 45:5314–5321. https://doi.org/10.1109/TPAMI.2022.3206148
31. Saedi SI, Khosravi H (2020) A deep neural network approach towards real-time on-branch fruit recognition for precision horticulture. Expert Syst Appl 159:113594
32. Ganesh P, Volle K, Burks TF, Mehta S (2019) Deep orange: mask R-CNN based orange detection and segmentation. IFAC-PapersOnLine 52(30):70–75
33. Arkin E, Yadikar N, Muhtar Y, Ubul K (2021) A survey of object detection based on CNN and transformer. In: IEEE International Conference on Pattern Recognition and Machine Learning (PRML, pp 99–108
34. Xiang AJ, Huddin AB, Ibrahim MF, Hashim FH (2021) An oil palm loose fruits image detection system using Faster R-CNN and Jetson TX2. In International Conference on Electrical Engineering and Informatics (ICEEI), pp 1–6
35. Zhang P, Dai X, Yang J, Xiao B, Yuan L, Zhang L, Gao J (2021) Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In: IEEE/CVF International Conference on Computer Vision, pp 2998–3008
36. Kumar D, Kukreja V (2022) Image-based wheat mosaic virus detection with Mask R-CNN model. In: International Conference on Decision Aid Sciences and Applications (DASA), pp 178–182
37. Chen X, Hsieh CJ, Gong B (2022) When vision transformers outperform ResNets without pre-training or strong data augmentations. CLR
38. Lian D, Yu Z, Sun X, Gao S (2022) As-MLP: An axial shifted MLP architecture for vision. ICLR
39. Tolstikhin IO, Houlsby N, Kolesnikov A, Beyer L, Zhai X, Unterthiner T, Dosovitskiy A (2021) MLP-mixer: An all-MLP architecture for vision. In: Advances in Neural Information Processing Systems 34:24261–24272
40. Liu Z, Deng Y, Ma F, Du J, Xiong C, Hu M, Ji X (2021) Target detection and tracking algorithm based on improved Mask R-CNN and LMB. In: International Conference on Control, Automation and Information Sciences (ICCAIS), pp 1037–1041
41. Pannerselvam K (2021) Adaptive parking slot occupancy detection using vision transformer and LLIE. In: IEEE International Smart Cities Conference (ISC2), pp 1–7
42. Ranftl R, Bochkovskiy A, Koltun V (2021) Vision transformers for dense prediction. In IEEE/CVF International Conference on Computer Vision, pp 12179–12188
43. Zhang Z, Lu X, Cao G, Yang Y, Jiao L, Liu F (2021) ViT-YOLO: Transformer-based YOLO for object detection. In: IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pp 2799–2808. https://doi.org/10.1109/ICCVW54120.2021.00314
44. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN. In: IEEE International Conference on Computer Vision, pp 2961–2969
45. Mai X, Zhang H, Jia X, Meng MQH (2020) Faster R-CNN with classifier fusion for automatic detection of small fruits. IEEE Trans Autom Sci Eng 17(3):1555–1569. https://doi.org/10.1109/TASE.2020.2964289
46. Luo Z, Nguyen M, Yan W (2022) Kayak and sailboat detection based on the improved YOLO with Transformer. In: International Conference on Control, Automation and Robotics (ICCAR)
47. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Guo B (2021) Swin Transformer: Hierarchical vision transformer using shifted windows. In: IEEE/CVF International Conference on Computer Vision, pp 10012–10022
48. Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: European Conference on Computer vision, Springer, pp 213–229
49. Dai Z, Cai B, Lin Y, Chen J (2021) Up-DETR: Unsupervised pre-training for object detection with transformers. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1601–1610
50. Chen S, Chen S, Xie E, Chongjian GE, Chen R, Liang D, Ping D, Luo P (2021) CycleMLP: A MLPlike architecture for dense prediction. ICLR 2022. https://openreview.net/forum?id=NMEceG4v69Y
51. Yu T, Li X, Cai Y, Sun M, Li P (2022) S2-MLP: spatial-shift MLP architecture for vision. In IEEE/CVF Winter Conference on Applications of Computer Vision, pp 297–306
52. Yan W (2019) Introduction to intelligent surveillance: surveillance data capture, transmission, and analytics. Springer Cham. https://doi.org/10.1007/978-3-030-10713-0

Bingjie Xiao is a PhD student with the Auckland University of Technology, Auckland New Zealand, her research interests are deep learning and computer vision.

Dr. Minh Nguyen is the head of the department within the School of Engineering, Computer & Mathematical Sciences at Auckland University of Technology, New Zealand. His research is primarily focused on the intersection of Computer Vision, Artificial Intelligence, and Virtual/Augmented Reality with an emphasis on their industrial applications.

Dr. Wei Qi Yan research interests include deep learning, intelligent surveillance, computer vision, multimedia computing, etc. Dr. Yan's expertise is computational mathematics and applied mathematics, computer science and computer engineering. Dr. Yan was a world's top 2% cited scientist listed by Stanford University in 2022.