agriculture

MDPI

Article

Farmland Obstacle Detection from the Perspective of UAVs Based on Non-local Deformable DETR

Dashuai Wang [1,2], Zhuolin Li [1,3], Xiaoqiang Du [3,4], Zenghong Ma [3,4,*] and Xiaoguang Liu [1,*]

[1] School of Microelectronics, Southern University of Science and Technology, Shenzhen 518005, China
[2] Guangdong Provincial Key Lab of Robotics and Intelligent System, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518005, China
[3] School of Mechanical Engineering and Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China
[4] Key Laboratory of Transplanting Equipment and Technology of Zhejiang Province, Hangzhou 310018, China
* Correspondence: mzh2018@zstu.edu.cn (Z.M.); liuxg@sustech.edu.cn (X.L.)

Abstract: In precision agriculture, unmanned aerial vehicles (UAVs) are playing an increasingly important role in farmland information acquisition and fine management. However, discrete obstacles in the farmland environment, such as trees and power lines, pose serious threats to the flight safety of UAVs. Real-time detection of the attributes of obstacles is urgently needed to ensure their flight safety. In the wake of rapid development of deep learning, object detection algorithms based on convolutional neural networks (CNN) and transformer architectures have achieved remarkable results. Detection Transformer (DETR) and Deformable DETR combine CNN and transformer to achieve end-to-end object detection. The goal of this work is to use Deformable DETR for the task of farmland obstacle detection from the perspective of UAVs. However, limited by local receptive fields and local self-attention mechanisms, Deformable DETR lacks the ability to capture long-range dependencies to some extent. Inspired by non-local neural networks, we introduce the global modeling capability to the front-end ResNet to further improve the overall performance of Deformable DETR. We refer to the improved version as Non-local Deformable DETR. We evaluate the performance of Non-local Deformable DETR for farmland obstacle detection through comparative experiments on our proposed dataset. The results show that, compared with the original Deformable DETR network, the mAP value of the Non-local Deformable DETR is increased from 71.3% to 78.0%. Additionally, Non-local Deformable DETR also presents great performance for detecting small and slender objects. We hope this work can provide a solution to the flight safety problems encountered by UAVs in unstructured farmland environments.

Keywords: UAVs; obstacle detection; deformable DETR; non-local deformable DETR

Citation: Wang, D.; Li, Z.; Du, X.; Ma, Z.; Liu, X. Farmland Obstacle Detection from the Perspective of UAVs Based on Non-local Deformable DETR. *Agriculture* **2022**, *12*, 1983. https://doi.org/10.3390/agriculture12121983

Academic Editor: Xiuliang Jin

Received: 13 October 2022
Accepted: 18 November 2022
Published: 23 November 2022

1. Introduction

With the development of agricultural robot technology, UAVs are becoming an important part of global agriculture aviation [1]. Specifically, UAVs with high-performance onboard sensors and task-specific action systems have been successfully deployed in farmland information collection and fine management [2–5]. However, the advantages and performance of UAVs have not been fully realized at present yet. One of the main reasons is that randomly distributed obstacles, such as trees, poles, buildings, people, and power towers pose a serious threat to its flight safety and operational efficiency [6]. Image sensors are widely used as the eyes of UAVs [7], so giving them human-like intelligent environmental awareness is an intuitive solution. How to quickly and accurately detect objects of interest in information-rich images is a technical bottleneck [8].

Previously, researchers have used a monocular camera [9], stereo camera [10], event camera [11] and other sensors to detect the obstacles based on various image processing techniques. Recently, deep learning neural networks have been used in the obstacle

detection [12,13], but they usually rely on the specific dataset and the detection of narrow and small object remains the challenging problem [8].

Deep learning offers a power tool to process agricultural images [14,15]. Since AlexNet [16] won the ImageNet competition in 2012, convolutional neural networks (CNNs) have significantly advanced the computer vision tasks. For example, object detection algorithms, such as YOLO [17], Faster R-CNN [18] and other networks, can quickly obtain the category and boundary box of targets; instance segmentation, such as Mask R-CNN [19] and PointRend [20], can obtain category, bounding box and mask information at the same time. Within local receptive fields, convolutional operations collect spatial and channel-wise features as powerful image representations in a hierarchical manner. Although it has advantages in local feature extraction, CNNs have difficulties in capturing global image information, such as the long-distance relationship, which is often critical to advanced computer vision tasks [21,22]. An intuitive solution is to expand the receptive field by stacking convolution layers, but this will make it difficult to optimize the model.

The attention mechanism has been widely used to increase the CNN's global representation capacity. The visual attention mechanism is the visual characteristic of the human visual system to actively select the object of attention and focus on it, which can effectively improve image processing capabilities such as image content screening and target retrieval [23]. In the perspective of artificial intelligence, the attention mechanism is a data processing method in machine learning, which essentially uses the relevant feature map to learn the weight distribution, then applies the learned weights on top of the original feature map, and finally performs weighted summation to quickly extract important features of sparse data [24–26]. It can be broadly divided into three categories, namely spatial attention: Non-local Network (NLNet [27]), channel attention: Squeeze-and-Excitation Network (SENet [28]) and temporal attention: Global-Local Temporal Representation (GLTR [29]). Non-local block in NLNet is a spatial self-attention variant that can capture long-rang dependencies within deep neural networks. Hu et al. introduced a squeeze-and-excitation (SE [28]) block to explicitly model the interdependence between feature channels. GLTR designed the temporal self-attention model to exploit multi-scale temporal cues in a video sequence. Additionally, there are some combinatorial variants. Woo et al. proposed an attention module-Convolutional Block Attention Module (CBAM [30]) that combines spatial and channel attention, in which the features extracted by channel attention are used as the input of the spatial attention module. Cao et al. proposes Global Context Network (GCNet [31]) based on non-local block and SE block to globally model the context. It has been proven that after inserting these modular blocks in the classical convolutional neural network architectures, the model performance can be greatly improved.

Transformer that exclusively rely on the self-attention mechanism to capture global dependencies has achieved remarkable success in natural language processing (NLP) [32]. Recently, many pioneering works have demonstrated that transformer architecture and its variants can also handle downstream computer vision tasks, such as image recognition: Vision Transformer (ViT [33]), Data-efficient image Transformers (DeiT [34]), Tokens-To-Token Vision Transformer (T2T [35]), Transformer in Transformer (TNT [36]), Conditional Positional encoding Vision Transformer (CPVT [37]), Shifted Windows Transformer (Swin Transformer [38]), object detection: Detection Transformer (DETR [39]), Deformable DETR [40], Swin Transformer, image segmentation: SEgmentation Transformer (SETR [41]), Pyramid Vision Transformer (PVT [42]), Transformer for semantic segmentation (Segmenter [43]), Swin Transformer, and video object tracking: Swin Transformer Tracker (SwinTrack [44]), Video Vision Transformer (ViViT [45]), Video Transformer (VidTr [46]) and Transformer Tracking (TransT [47]).

Convolution operation in CNNs is good at extracting local features, but have difficulty to capture global representation. The hierarchical self-attention in the transformer is conducive to building long-rang dependencies, but ignores local features. Currently, some works use a combination of CNN and Transformer to obtain local features, global representation and long-rang dependences: Convolutional vision Transformer (CvT [48]),

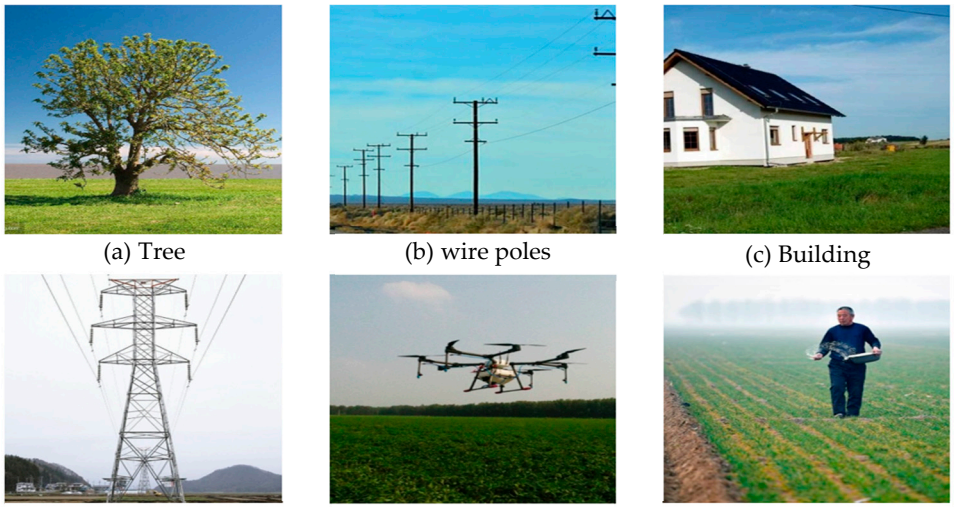
Conformer [49] and CNNs meet transformers (CMT [50]). Specifically, DETR is the first end-to-end baseline network for deploying transformer in object detection. Different from the R-CNN and YOLO, DETR regards object detection as a direct set prediction problem, and simplifies the detection pipeline by dropping some hand-crafted components such as anchor generation and non-maximum suppression. DETR uses ResNet [51] to extract image features, then outputs 100 prediction results in parallel based on the transformer encoder-decoder architecture and finally determines the final prediction classes and bounding boxes through bipartite matching. Although DETR significantly outperforms competitive baselines, there are still three problems with DETR. First, compared to existing object detection methods, DETR requires more epochs to converge. Second, insufficient detection performance of DETR for small objects. Lastly, the computational complexity of DETR is still sensitive to the resolution of the image or feature map. To address these issues, Deformable DETR introduces the idea of deformable convolution and multi-scale feature maps to form the so-called Multi-scale Deformable Attention Module. The experimental results show that Deformable DETR not only alleviates the problems of slow convergence and high computational complexity of DETR, but also achieves better performance than DETR.

Random and discrete obstacles in the natural farmland environment pose a direct threat to the flight safety of UAVs. Usually, the images captured by the UAV's onboard camera are filled with a lot of background noise, which increases the difficulty for obstacle detection. In this paper, we try to deploy the modified Deformable DETR for the task of agricultural UAV-based farmland obstacle detection. In Deformable DETR, the ResNet-style CNN architecture models the spatial and local features of input images, while the transformer builds the long-distance dependencies. However, the global modeling ability of Deformable DETR is still insufficient for detecting the small farmland objects. The motivation of this work is to further improve the global modeling capability of Deformable DETR by introducing the global modeling capability in the front-end CNN. In this work, we achieve this by introducing a Non-Local module into the CNN feature extraction network in the Deformable DETR front-end. The main reason is that non-local operation can capture long-range dependencies by computing the response of a location as a weighted sum of all location features in the input feature map. Our proposed Non-local Deformable DETR combines the local feature extraction ability of CNN, the global modeling ability of non-local and the self-attention mechanism of transformer to improve the object detection accuracy while maintaining the efficiency of the Deformable DETR model.

2. Materials and Methods

2.1. Dataset

The dataset proposed by our previous work [6] contained 3700 samples served as the basis for this study. Additionally, it can be classified into six categories: tree, wire poles, building, power tower, UAVs and person. In this work, we collected more images containing obstacles through various methods (manual photography, UAV photography and web search) and added them to the raw dataset. In the preprocessing stage, we manually selected the raw dataset through data cleaning to remove some low-quality samples. In addition, we also resize the images of different resolutions to the same resolution through a cropping operation. As shown in Figure 1, our dataset contains six classes of typical obstacles which are common in the farmland. The percentage values of tree, wire poles, building, power tower, UAVs and person are 14.48%, 15.44%, 16.81%, 15.99%, 15.40% and 21.87% respectively. There are a total of 6000 images, each with a resolution of 416 $\times$ 416. All 11,578 objects in our dataset were annotated by Labelme [52]. We randomly selected 4800 images as the training set, 600 images as the validation set and 600 images as the test set, with a ratio of 8:1:1.



(a) Tree (b) wire poles (c) Building

(d) power tower (e) UAVs (f) Person

Figure 1. Examples of field obstacle images.

2.2. Model structure

2.2.1. Deformable DETR

Without the need of hand-designed components such as NMS or anchors, DETR can predict the final set of detections in parallel by combining a common CNN with a transformer architecture. However, DETR requires long training time to converge and has relatively poor performance for small object detection. To solve these two issues, Zhu et al. [40] introduced the idea of deformable convolution and multi-scale features in convolutional neural networks into DETR and proposed the Deformable DETR. Deformable DETR uses ResNet-50 [51] as the backbone to extract the multi-scale features. Deformable transformer (encoder and decoder) extracts and strengthens the feature maps from the output feature maps of stages C_3-C_5 in ResNet by using multi-scale deformable attention module. The core of Deformable DETR is the deformable attention module and multi-scale deformable attention module

The deformable attention module is a local attention mechanism, which means it only pays attention to a small set of key sampling points around the reference point, independent of the spatial size of the feature map [40]. Given an input feature map $x \in \mathbb{R}^{C\times H\times W}$, query elements with content features z_q and 2D reference points p_q, the equation of the deformable attention feature is calculated by:

$$DeformAttn(z_q, p_q, x) = \sum_{m=1}^{M} W_m \left[\sum_{k=1}^{K} A_{mqk}\cdot W'_m x\left(p_q + \Delta p_{mqk}\right)\right], \quad (1)$$

where m is the attention head, k is the sampled keys, K is the total sampled keys ($K \ll HW$), Δp_{mqk} is the sampling offset and A_{mqk} is the attention weight of the k^{th} sampling point in the m^{th} attention head.

The deformable attention module and multi-scale form the multi-scale deformable attention module. Given the input multi-scale feature maps $\left\{x^l\right\}_{l-1}^{L}$, where $x^l \in \mathbb{R}^{C\times H\times W}$. Let $\hat{p}_q \in [0,1]^2$ be the normalized coordinates of the reference point. The equation of multi-scale deformable attention feature can be calculated by:

$$MSDeformAttn(z_q, \hat{p}_q, \left\{x^l\right\}_{l=1}^{L}) = \sum_{m=1}^{M} W_m \left[\sum_{l=1}^{L}\sum_{k=1}^{K} A_{mlqk}\cdot W'_m x^l\left(\varnothing_l(\hat{p}_q) + \Delta p_{mlqk}\right)\right], \quad (2)$$

where l is the input feature level, $\hat{p}_q$ is the normalized coordinates of the reference point, Δp_{mlqk} is the sampling offset of the k^{th} sampling point in the l^{th} feature level and the m^{th} attention head and A_{mlqk} is attention weight of the k^{th} sampling point in the l^{th} feature

level and the m^{th} attention head. $\varnothing_l(\hat{p}_q)$ rescales the normalized coordinates $\hat{p}_q$ to the input feature map of the l^{th} level.

Compared to DETR, Deformable DETR replaces the multi-head attention module in the transformer encoder with the multi-scale deformable attention module and replaces the cross-attention module in transformer decoder with multi-scale deformable cross-attention module. The self-attention module in the transformer decoder remains unchanged.

2.2.2. ResNet

Both DETR and Deformable DETR use ResNet to extract original feature maps. ResNet is a popular backbone in many state-of-the-art deep learning algorithms. The basic idea of ResNet is to introduce a "shortcut connection" that can skip one or more layers to solve the model degradation problem. As shown in Figure 2, the residual block uses the shortcut connection to perform identity mapping, which connects the input x with the $F(x)$ obtained through the stacked weight layers, without adding additional parameters or increasing the computational complexity.

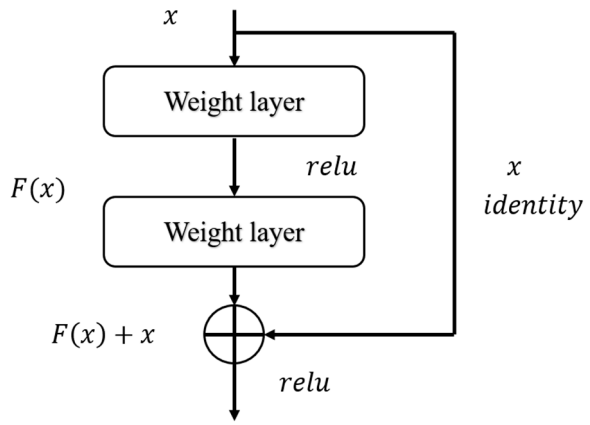


Figure 2. The building block of ResNet.

When x and F are of the same dimension, the output is given by:

$$y = F(x, \{W_i\}) + x \tag{3}$$

where x, y are the input and output vector of residual block and $F(x, \{W_i\})$ is the residual mapping to be learned. When the dimensions of x and F are different, the input x needs to match the dimensions by:

$$y = F(x, \{W_i\}) + W_s x, \tag{4}$$

where W_s is the linear mapping function.

2.2.3. Non-Local Neural Networks

Traditional convolution operations lack the ability of global modeling due to the limitation of local receptive fields. Long-range dependencies are usually achieved through hierarchical convolution and pooling. Inspired by the self-attention mechanism in NLP, non-local neural networks introduce self-attention to CNN to capture long-distance dependencies in the feature extraction process. A generic non-local operation in deep neural networks is defined as:

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f(x_i, x_j) g(x_j) \tag{5}$$

where x is the input feature, y is the corresponding output feature, i is the index of output position, j is the index of all possible positions in feature, f is the function (Embedded Gaussian) that calculates the relationship between i and all j, g is the function that computes

the representation of the input signal at position j and $C(x)$ is a factor that normalizes the response.

Non-local operations can be implemented in the form of non-local blocks, which means it can be easily plugged into conventional convolutional layers within standard networks. Based on Equation (5), the non-local block is defined as:

$$z_i = W_z y_i + x_i \tag{6}$$

where "$+x_i$"denotes residual shortcut connection and $W_z y_i$ represents linear transformation.

An example of non-local block is shown in Figure 3. W_v, W_k, W_q and W_z are weight matrixes to be learned and "$\oplus$" denotes element-wise sum after shortcut connection, while "$\otimes$" denotes matrix multiplication.

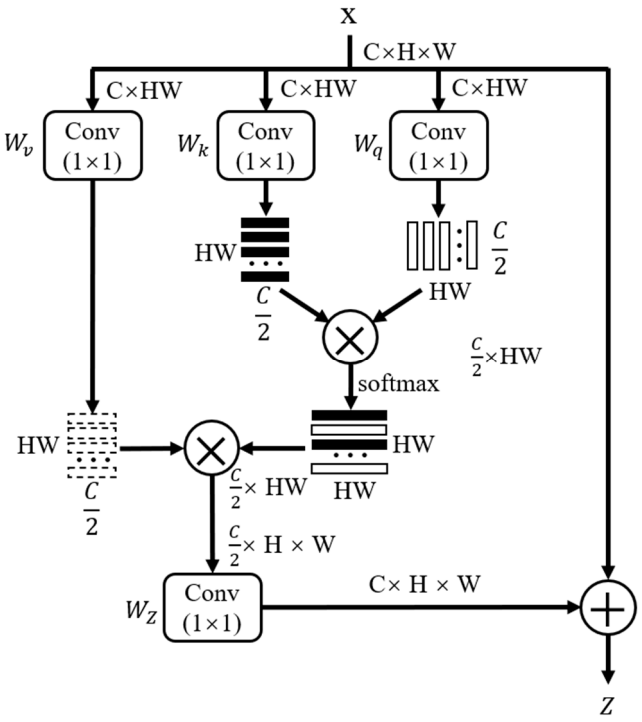


Figure 3. The structure of a non-local block.

2.2.4. Non-Local Deformable DETR

In Deformable DETR, convolution operations in ResNet architecture capture multi-scale local features and the encoder-decoder in the transformer architecture conducts local self-attention. Therefore, Deformable DETR lacks the ability to learn global representations over long distances. Based on the non-local structure, we introduce the global modeling capability to the front-end ResNet to further improve the overall performance of Deformable DETR.

As shown in Figure 4, non-local blocks are inserted into all the residual blocks in Stage 4 and 5 in ResNet-50. Specifically, in each optimized residual block, the non-local block is added after the 3×3 convolution layer to establish long-distance dependency and improve the feature extraction ability of the model.

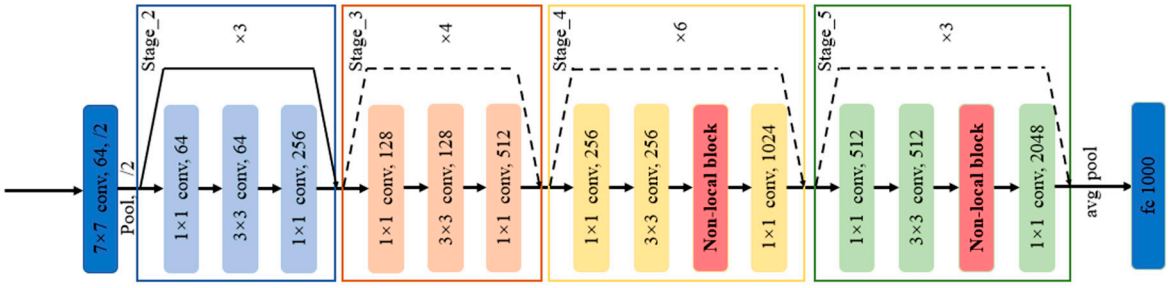


Figure 4. Improved ResNet based on non-local block.

The transformer architecture in Deformable DETR remains unchanged. The overall structure of Non-local Deformable DETR is shown in Figure 5.

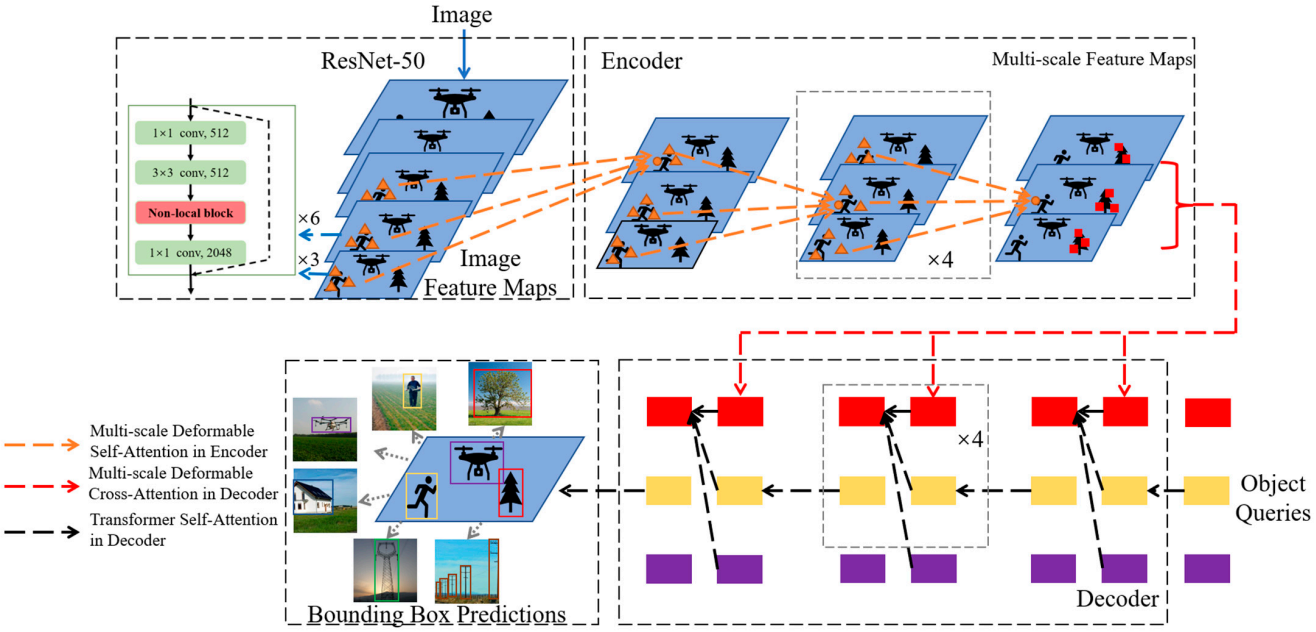


Figure 5. The structure of Non-local Deformable DETR.

2.3. The Overview of Data Flow

In this paper, we improved the Deformable DETR by Non-local block to enhance the detection accuracy of farmland obstacles; an overview of the data flow is shown in Figure 6. First, the raw dataset was cleaned and cropped into the pre-processed dataset, and then it was divided into training set, validation set and test set with a ratio of 8:1:1. Secondly, we used the training set and validation set to train the proposed Non-local Deformable DETR. Finally, the test set was used to evaluate the model's predicting performance.

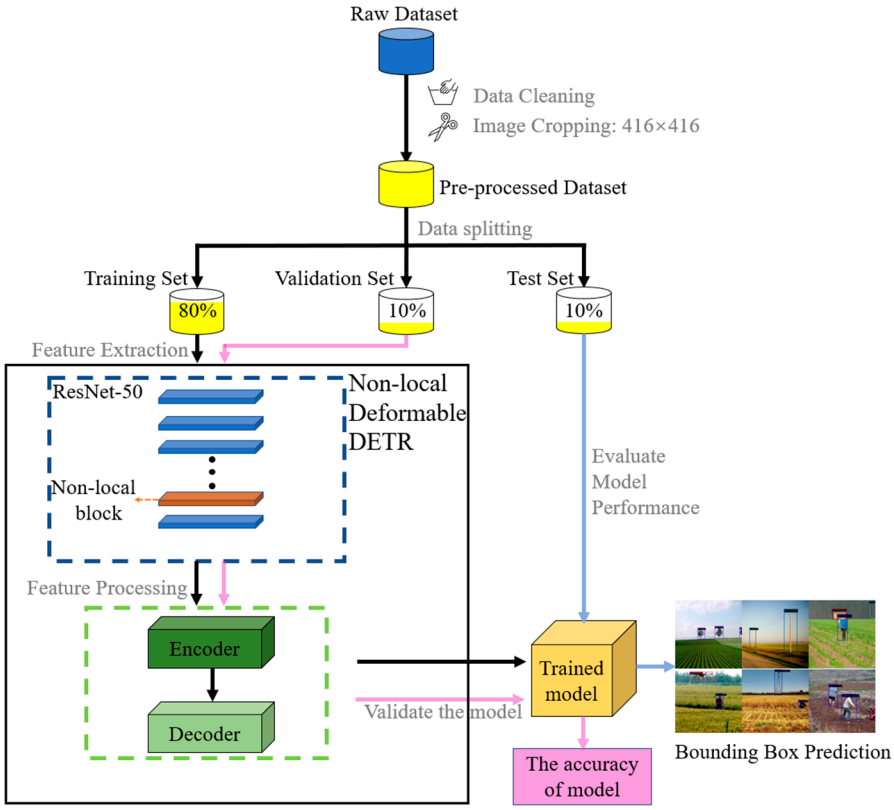


Figure 6. Overview of the data flow.

2.4. Evaluation Metrics

In this study, AP and mAP were used to evaluate the performance of the model with Equations (7) and (8):

$$AP = \int_0^1 P(R)dr, \tag{7}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n}(AP)_i, \tag{8}$$

where AP indicates the average precision of a single category, mAP indicates the average of multiple category's AP, P represents the accuracy rate which can be calculated by Equation (9), R is the recall rate that can be obtained by Equation (10) and P(R) denotes the mapping function of P and R:

$$P = \frac{TP}{TP + FP} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

where TP (True positive) indicates the number of positive samples that are correctly predicted as positive, FP (False Positive) represents the number of samples that the model predicts as positive, but which are actually negative, FN (False Negative) means the number of misclassified samples that are actually positive but are classified as negative and TN (True Negative) stands for the number of negative samples that are correctly classified as negative.

3. Results and Discussion

3.1. Implementaion Details

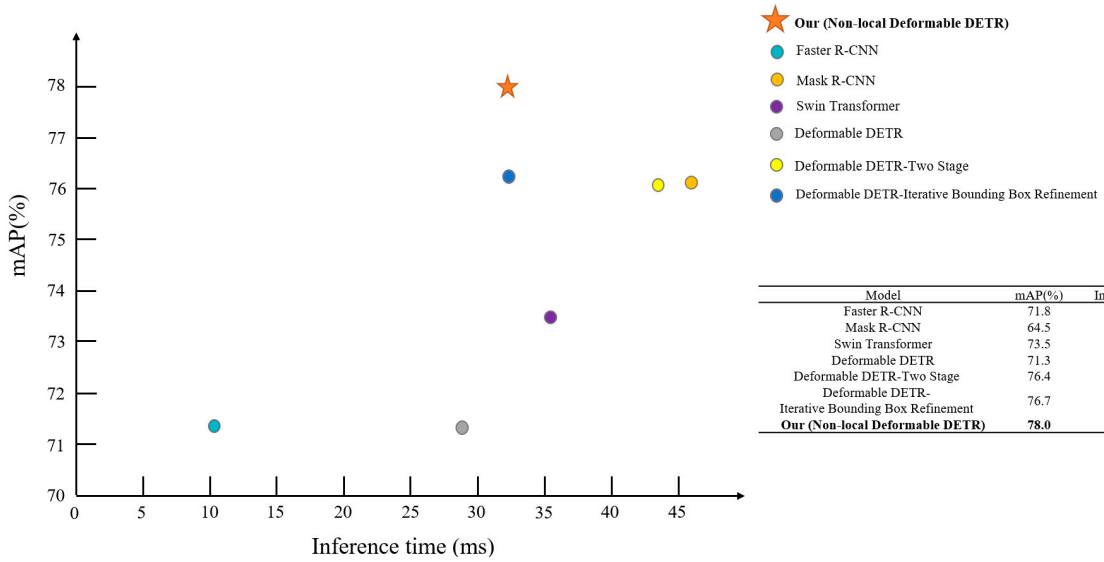
The configuration of the computer used for algorithm development is as follows: the central processing unit (CPU) is Intel Core i9-12900K; the graphics processing unit

(GPU) is an NVIDIA GeForce RTX 3090Ti with 24 GB on-board memory; the physical memory is DDR5 5200 (16 G); the running operation system is Ubuntu 20.04 LTS; the PyTorch deep learning framework and is used to build, train and validate the Non-local Deformable DETR.

Considering the model training effect and experimental conditions, this paper adopts the transfer learning training strategy. The backbone network is initialized with ResNet-50 weights pretrained on ImageNet. Training epochs and iterations are set to 50 and 1200, respectively. In order to avoid the instability of the model caused by large learning rate at the beginning of training, a warmup strategy is adopted to adjust the learning rate. In the initial 500 iterations, the learning rate is gradually adjusted from 2.4×10^{-4} to 2.5×10^{-3}. The momentum factor is 0.9 and the weight decay coefficient is 1×10^{-4}.

3.2. Results and Analysis

Focusing on three metrics (AP value, parameters and inference time), we conducted two kinds of comparative experiments based on our farmland obstacle dataset to evaluate Non-local Deformable DETR. Firstly, we reproduced Deformable DETR and its two variants, Deformable DETR-Iterative Bounding Box Refinement and Deformable DETR-Two Stage [40]. Secondly, we repeated some other classic object detection algorithms, such as Faster R-CNN, Mask R-CNN and Swin Transformer. The overall comparison results are shown in Figure 7. Non-local Deformable DETR achieves the best mAP with moderate inference time.



Model	mAP(%)	Inf time(ms)
Faster R-CNN	71.8	10.7
Mask R-CNN	64.5	20.3
Swin Transformer	73.5	35.5
Deformable DETR	71.3	28.8
Deformable DETR-Two Stage	76.4	43.1
Deformable DETR-Iterative Bounding Box Refinement	76.7	32.6
Our (Non-local Deformable DETR)	**78.0**	32.0

Figure 7. Performance comparison of different models.

As shown in Tables 1 and 2, the overall AP value and the AP value of each category of the two variants are higher than the vanilla Deformable DETR. In terms of the mAP value, Deformable DETR-Iterative Bounding Box Refinement and Deformable DETR-Two Stage are 5.4% and 5.1% higher than the vanilla Deformable DETR, respectively. In particular, the AP_S value is increased by 8.8% and 18.5%, respectively. Meanwhile, parameters increased slightly, by 0.68 million and 0.99 million, and the inference time increased by 3.8 ms and 14.3 ms, respectively. Compared to Deformable DETR-Iterative Bounding Box Refinement, Deformable DETR-Two Stage achieves a slight performance gain at the cost of introducing larger latency (10.5 ms). This work takes the Deformable DETR-Iterative Bounding Box Refinement as the baseline, and forms Non-local Deformable DETR by inserting non-local blocks on it. As shown in Table 1, Non-local Deformable DETR secures the best mAP (78.0%), with an inference time of 32.0 ms, which is slightly lower than DETR-Iterative Bounding Box Refinement (32.6 ms). Although the detection speed of Non-local Deformable DETR is only one-third that of Faster R-CNN, it achieves an mAP gain of 6.2%. For UAVs-

based farmland obstacle detection task, we need a better trade-off between detection accuracy and speed. Therefore, we believe that the current detection speed of Non-Local Deformable DETR is acceptable, although it needs to be further improved.

Table 1. Performance comparison between different models.

Model	Bounding Box						Parameters (Million)	Inference Time (ms)
	mAP (%)	AP_{50} (%)	AP_{75} (%)	AP_S (%)	AP_M (%)	AP_L (%)		
Faster R-CNN	71.8	91.6	83.8	46.6	73.4	79.8	41.15	10.7
Mask R-CNN	64.5	85.8	77.6	27.8	67.9	76.7	43.77	20.3
Swin Transformer	73.5	92.1	85.1	45.5	74.8	82.2	68.71	35.5
Deformable DETR	71.3	92.5	81.0	35.0	73.3	80.6	39.82	28.8
Deformable DETR-Iterative Bounding Box Refinement	76.7	93.3	84.2	43.8	77.7	86.4	40.50	32.6
Deformable DETR-Two Stage	76.4	93.4	83.7	53.5	77.7	84.6	40.81	43.1
Non-local Deformable DETR	78.0	94.5	85.5	48.2	79.0	85.2	42.86	32.0

Note: APs, AP_M and AP_L correspond, respectively, to the AP value based on pixel area sizes less than 32^2, between 32^2 and 96^2 and larger than 96^2.

Table 2. Performance comparison of different models in each category.

Model	AP (Bounding Box)					
	UAVs (%)	Building (%)	Power-Tower (%)	Person (%)	Tree (%)	Wire Pole (%)
Faster R-CNN	85.5	66.3	78.2	69.9	76.5	54.4
Mask R-CNN	81.0	63.1	64.6	65.5	72.6	40.1
Swin Transformer	85.5	69.5	77.8	74.8	76.9	56.4
Deformable DETR	86.0	68.7	79.1	70.9	72.6	50.7
Deformable DETR-Iterative Bounding Box Refinement	90.6	75.9	82.2	76.7	79.5	55.6
Deformable DETR-Two Stage	89.7	73.0	80.9	77.5	76.9	60.5
Non-local Deformable DETR	90.2	75.8	83.1	78.2	78.5	62.2

Table 2 presents the detection results of different algorithms for six classes of farmland obstacles. For power-tower and person detection, our proposed Non-local Deformable DETR achieves the highest AP. For UAVs and buildings detection, Non-local Deformable DETR does not secure the best results (0.04% and 0.01% lower than Deformable DETR-Iterative Bounding Box Refinement respectively), but also performs well. Specifically, in farmland, wire poles and UAVs pose a serious danger to each other. Given the slender shape of wire pole, its detection is more challenging. Fortunately, our model obtains the best outcomes again by outperforms vanilla Deformable DETR by 11.5% in AP. We attribute the benefits to the enhanced global modeling capability for CNN feature extraction by non-local operations.

Figure 8 shows some samples containing the detected objects. It can be seen that Non-local Deformable DETR can accurately detect different objects with a suitable bounding box. Specially, the detection results of the small power pole in the lower right image are also good. However, as shown in Figure 9, there are also some falsely detected objects. In Figure 9a, our model cannot detect the second person because it is blurred. In Figure 9b, our model wrongly detected the UAV as building, because the number of such kind of UAV in the training set is less, and the feature of the image is closed to the building. In Figure 9c, our model cannot detect the person due to the backlight environment.

Figure 8. Test results of different objects.

(a) (b) (c)

Figure 9. The wrongly detected objects: (**a**) Our model failed to detect the person behind. (**b**) An airplane is mistakenly identified as a building. (**c**) Our model failed to detect a motorcyclist.

4. Conclusions

Focusing on the task of UAV-based unstructured farmland obstacle detection, this work proposed the Non-local Deformable DETR to enhancing the performance of the original Deformable DETR. Specially, we introduced the non-local blocks into the front-end ResNet to improve the model's global representation capacity when extracting feature maps. Combing the local self-attention mechanism in deformable transformer, our Non-local Deformable DETR can not only capture local features, but also model long-distance dependencies. Based on our farmland obstacle dataset, we conducted a series of experiments to investigate the performance of our improved model. Compared with Deformable DETR and other high-performance object detection algorithms (Faster R-CNN, Mask R-CNN and Swin Transformer), Non-local Deformable DETR achieved the best mAP (78.0%) with moderate inference time (32.0 ms). Additionally, Non-local Deformable DETR also demonstrated advantages detecting small and slender objects, such as wire poles. Taking detection accuracy and speed into account, the proposed Non-local Deformable DETR has great potential to be deployed in UAVs-based farmland obstacle detection tasks. In the future, we will continue to optimize the model to accelerate the detection speed.

Author Contributions: D.W., X.D. and Z.M. designed the research. D.W. and Z.L. participated in the measurements and data analysis. D.W. and Z.L. wrote the first draft of the manuscript. D.W., Z.L. and X.L revised and edited the final version of the manuscript. D.W. and X.L. are responsible for funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (32001424) and Shenzhen Science and Technology Program (JCYJ20210324102401005).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available upon request to the authors.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Maes, W.H.; Steppe, K. Perspectives for remote sensing with unmanned aerial vehicles in precision agriculture. *Trends Plant Sci.* **2019**, *24*, 152–164. [CrossRef]
2. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Lagkas, T.; Moscholios, L. A compilation of UAV applications for precision agriculture. *Comput. Netw.* **2020**, *172*, 107148. [CrossRef]
3. Maimaitijiang, M.; Sagan, V.; Sidike, P.; Hartling, S.; Esposito, F.; Fritschi, F.B. Soybean yield prediction from UAV using multimodal data fusion and deep learning. *Remote Sens. Environ.* **2020**, *237*, 111599. [CrossRef]
4. Guo, S.; Li, J.; Yao, W.; Hu, X.; Wei, X.; Long, B.; Wu, H.; Li, H. Optimization of the factors affecting droplet deposition in rice fields by rotary unmanned aerial vehicles (UAVs). *Precis. Agric.* **2021**, *22*, 1918–1935. [CrossRef]
5. Xue, X.; Lan, Y.; Sun, Z.; Chang, C.; Hoffmann, W.C. Develop an unmanned aerial vehicle based automatic aerial spraying system. *Comput. Electron. Agric.* **2016**, *128*, 58–66. [CrossRef]
6. Wang, D.; Li, W.; Liu, X.; Li, N.; Zhang, C. UAV environmental perception and autonomous obstacle avoidance: A deep learning and depth camera combined solution. *Comput. Electron. Agric.* **2020**, *175*, 105523. [CrossRef]
7. Park, J.; Cho, N. Collision avoidance of hexacopter UAV based on LiDAR data in dynamic environment. *Remote Sens.* **2020**, *12*, 975. [CrossRef]
8. Badrloo, S.; Varshosaz, M.; Pirasteh, S.; Li, J. Image-Based Obstacle Detection Methods for the Safe Navigation of Unmanned Vehicles: A Review. *Remote Sens.* **2022**, *14*, 3824. [CrossRef]
9. Liu, J.; Li, H.; Liu, J.; Xie, S.; Luo, J. Real-Time Monocular Obstacle Detection Based on Horizon Line and Saliency Estimation for Unmanned Surface Vehicles. *Mob. Netw. Appl.* **2021**, *26*, 1372–1385. [CrossRef]
10. Barry, A.J.; Florence, P.R.; Tedrake, R. High-speed autonomous obstacle avoidance with pushbroom stereo. *J. Field Robot.* **2018**, *35*, 52–68. [CrossRef]
11. Falanga, D.; Kleber, K.; Scaramuzza, D. Dynamic obstacle avoidance for quadrotors with event cameras. *Sci. Robot.* **2020**, *5*, eaaz9712. [CrossRef]
12. Qiu, Z.; Zhao, N.; Zhou, L.; Wang, M.; Yang, L.; Fang, H.; He, Y.; Liu, Y. Vision-based moving obstacle detection and tracking in paddy field using improved yolov3 and deep SORT. *Sensors* **2020**, *20*, 4082. [CrossRef]
13. Haris, M.; Hou, J. Obstacle Detection and Safely Navigate the Autonomous Vehicle from Unexpected Obstacles on the Driving Lane. *Sensors* **2020**, *20*, 4719. [CrossRef]
14. Wang, D.; Cao, W.; Zhang, F.; Li, Z.; Xu, S.; Wu, X. A review of deep learning in multiscale agricultural sensing. *Remote Sens.* **2022**, *14*, 559. [CrossRef]
15. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
17. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1–9. [CrossRef]
19. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2961–2969.
20. Kirillov, A.; Wu, Y.; He, K.; Girshick, R. Pointrend: Image segmentation as rendering. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 9799–9808.
21. Zhu, Z.; Xu, M.; Bai, S.; Huang, T.; Bai, X. Asymmetric non-local neural networks for semantic segmentation. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October –2 November 2019; pp. 593–602.
22. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In Proceedings of the 36th International Conference on Machine Learning PMLR, Long Beach, CA, USA, 10–15 June 2019; pp. 7354–7363.

23. Katsuki, F.; Constantinidis, C. Bottom-up and top-down attention: Different processes and overlapping neural systems. *Neuroscientist* **2014**, *20*, 509–521. [CrossRef]
24. Mnih, V.; Heess, N.; Alex, G.; Kavukcuoglu, K. Recurrent models of visual attention. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2204–2212.
25. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6077–6086. [CrossRef]
26. Chaudhari, S.; Mithal, V.; Polatkan, G.; Ramanath, R. An attentive survey of attention models. *ACM Trans. Intell. Syst. Technol.* **2021**, *12*, 1–32. [CrossRef]
27. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
28. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
29. Li, J.; Wang, J.; Tian, Q.; Gao, W.; Zhang, S. Global-local temporal representations for video person re-identification. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3958–3967.
30. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 3–19.
31. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27–28 October 2019.
32. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, L. Attention is all you need. Adv. *Neural Inf. Process. Syst.* **2017**, *30*, 6000–6010.
33. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
34. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jegou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Xiamen, China, 8–11 November 2021; pp. 10347–10357.
35. Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.; Tay, F.E.H.; Feng, J.; Yang, S. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 558–567.
36. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15908–15919.
37. Chu, X.; Tian, Z.; Zhang, B.; Wang, X.; Wei, X.; Xia, H.; Shen, C. Conditional positional encodings for vision transformers. *arXiv* **2021**, arXiv:2102.10882.
38. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
39. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 213–229.
40. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv* **2020**, arXiv:2010.04159.
41. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.S.; et al. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 6881–6890.
42. Wang, W.; Xie, E.; Li, X.; Fan, D.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 568–578.
43. Strudel, R.; Garcia, R.; Laptev, I.; Schmid, C. Segmenter: Transformer for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 7262–7272.
44. Lin, L.; Fan, H.; Xu, Y.; Ling, H. Swintrack: A simple and strong baseline for transformer tracking. *arXiv* **2021**, arXiv:2112.00995.
45. Arnab, A.; Dehghani, M.; Heigold, G.; Sun, C.; Lučić, M.; Schmid, C. Vivit: A video vision transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 6836–6846.
46. Zhang, Y.; Li, X.; Liu, C.; Shuai, B.; Zhu, Y.; Brattoli, B.; Chen, H.; Marsic, I.; Tighe, J. Vidtr: Video transformer without convolutions. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 13577–13587.
47. Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; Lu, H. Transformer tracking. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 8126–8135.

48. Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, Y.; Zhang, L. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 22–31.
49. Peng, Z.; Huang, W.; Gu, S.; Xie, L.; Wang, Y.; Jiao, J.; Ye, Q. Conformer: Local features coupling global representations for visual recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 367–376.
50. Guo, J.; Han, K.; Wu, H.; Tang, Y.; Chen, X.; Wang, Y.; Xu, C. Cmt: Convolutional neural networks meet vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 12175–12185.
51. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
52. wkentaro/labelme-Image Polygonal Annotation with Python. Available online: https://github.com/wkentaro/labelme (accessed on 19 November 2022).