Transformers only look once with nonlinear combination for real-time object detection

Xia, R., Li, G., Huang, Z., Pang, Y. and Qi, M.

Xia, R., Li, G., Huang, Z. *et al.* Transformers only look once with nonlinear combination for real-time object detection. *Neural Comput & Applic* (2022).
https://doi.org/10.1007/s00521-022-07333-y

Transformers Only Look Once with Non-linear Combination for Real-time Object Detection

Ruiyang Xia[a], Guoquan Li[a,*], Zhengwen Huang[b], Yu Pang[c], Man Qi[d]

[a]School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[b]Department of Electronic and Electrical Engineering, Brunel University London, London, UB8-3PH, United Kingdom
[c]Key Laboratory of Photoelectric Information Sensing and Transmission Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[d]School of Engineering, Canterbury Christ Church University, England Kent, CT1-1QU, United Kingdom

Abstract

In this article, a novel real-time object detector called Transformers Only Look Once (TOLO) is proposed to resolve two problems. The inefficiency of building long-distance dependencies among local features for amounts of modern real-time object detectors and the lack of inductive biases for vision Transformer networks with heavy computational cost. TOLO is composed of Convolutional Neural Network (CNN) backbone, Feature Fusion Neck (FFN), and different Lite Transformer Heads (LTHs), which are used to transfer the inductive biases, supply the extracted features with high-resolution and high-semantic properties, and efficiently mine multiple long-distance dependencies with less memory overhead for detection, respectively. Moreover, to find the massive potential correct boxes during prediction, we propose a simple and efficient non-linear combination method between the object confidence and the classification score. Experiments on the PASCAL VOC 2007, 2012, and the MS COCO 2017 datasets demonstrate that TOLO significantly outperforms other state-of-the-art methods with small input size. Besides, the proposed non-linear combination method can further elevate the detection performance of TOLO by boosting the results of potential correct predicted boxes without increasing the training process and model parameters.

Keywords: Real-time object detector, TOLO, Vision Transformer networks, Non-linear combination.

1. Introduction

Object detection is an attractive and challenging task of Computer Vision (CV). The attraction comes from its widespread applications such as autonomous driving and robot navigation while the challenge is credited to the changing scales, complicated shapes, and multiple categories. With the quick development of the Convolutional Neural Network (CNN), the number of object detection models has increased rapidly. Although there are various models, all of them can be divided into anchor-based [1, 2, 3, 4] and anchor-free methods [5, 6, 7] and are built by deep stacks of convolution operations which are sensitive to local interested regions and requires fewer parameters than Multi-Layer Perception (MLP).

However, a remarkable character of these methods is image features extracted from detection networks are only limited to the local regions. This lacks the long-distance dependencies which are important for the network to focus on the interesting regions and ignore noisy ones within a whole feature map [8]. Besides, the work in [9] has already mathematically proved that the effective receptive fields of the extracted features are much smaller than they theoretically own, which means the mechanism of deep stacks of convolution operations is not useful to build long-distance dependencies among local image features.

Therefore, to overcome the limitation of the intrinsic locality of convolution operations, there are some self-attention mechanisms based on local features have been proposed [10, 11, 12]. On the other hand, Transformer [13], a creative network which was major in Nature Language Processing (NLP) to mine the mul-

*Corresponding author
Email addresses: S190101135@stu.cqupt.edu.cn (Ruiyang Xia), ligq@cqupt.edu.cn (Guoquan Li), Zhengwen.Huang@brunel.ac.uk (Zhengwen Huang), pangyu@cqupt.edu.cn (Yu Pang), man.qi@canterbury.ac.uk (Man Qi)

tiple long-distance dependencies among time sequence information parallelly, has recently been introduced in CV and achieve state-of-the-art results in many vision tasks. The success of adopting variants of vision Transformer networks has strongly proved the necessity of building long-distance dependencies among image features and the redundancy of repeating convolution operations [14, 15, 16, 17, 18]. However, due to the lack of inductive biases such as translation equivariance and locality compared with CNN, vision Transformer networks can not generalize well during prediction, which means the number of data should be sufficient [14] or the reasonable combinations of training tricks should be needed [15] during training. On the other hand, images with high resolution are processed under the accumulation of the Self-Attention Network (SAN) and MLP in the vision Transformer networks will lead to the rapid increasing of computational complexity.

Some works have already tried to combine the inductive biases of CNN and the globality of vision Transformer network in image recognition [15, 16, 17, 18, 19], semantic segmentation [20] for the sake of improving model performance without the highly increasing complexity. Inspired by this thinking, we try to incorporate both the CNN and the vision Transformer networks into the area of real-time object detection reasonably and hence propose our detector named Transformers Only Look Once (TOLO). Specifically, our detector consists of a CNN backbone, Feature Fusion Neck (FFN), and different lightweight vision Transformer detection heads. The CNN backbone is employed to transfer inductive biases to the vision Transformer networks behind. FFN plays a role in the transition from CNN to vision Transformer networks and offers abundant high-resolution and high-semantic features. Finally, the proposed vision Transformer detection head called Lite Transformer Head (LTH) concentrates on building multiple long-distance dependencies among local features and finding different objects with less memory overhead.

In addition, as we stated above, the high complexity of samples leads to the difficulty of detection, the works in [21] and [22] found many correct boxes with low results and incorrect boxes with high results during prediction for detectors, which we called false negative predicted boxes and false positive predicted boxes, respectively. Similarly, this phenomenon has also existed in our detector, especially for the false negative predicted boxes. Therefore, to elevate the results of the false negative predicted boxes without increasing the training process and model parameters, we propose a simple and efficient method to combine the object confidence and the classification score non-linearly only in the inferring stage.

Our main contributions are summarized as follows:

(1) To efficiently build the multiple long-distance dependencies among local features and transfer the inductive biases to the lightweight vision Transformer network. We propose a novel real-time object detector called TOLO, which contains CNN backbone, FFN, and LTHs. It combines the inductive biases of CNN and the globality of LTHs, which can improve the detection performance while keeping high inferring speed and adopting less memory overhead.

(2) To further elevate the results of false negative predicted boxes. We propose a simple and efficient non-linear combination method between the object confidence and the classification score without increasing the training process and model parameters.

(3) To validate the effectiveness of TOLO and our non-linear combination method, we conduct extensive experiments on different datasets. Experimental results indicate that our detector can achieve 83.2%, 79.3% mean Average Precision @50 (mAP@50), and 36.6% Average Precision (AP) on the PASCAL VOC 2007, 2012 [23] and the MS COCO 2017 datasets [24] respectively, which significantly outperforms the existing state-of-the-art methods with even smaller input size and our proposed non-linear combination method can further boost the detection performance with only introducing marginal inferring process.

2. Related Works

2.1. Object Detection Models

The work in [25] has already pointed out that the difference between anchor-based and anchor-free methods is just the way to define the positive and negative samples. Therefore, to better distinguish these methods, they can be divided into one-stage or two-stage detectors by determining whether to generate Region of Interests (RoIs). Specifically, unlike the one-stage detectors which directly classify and regress anchor boxes [3, 4] or anchor points [5, 6], the series of R-CNN methods use Region Proposal Network (RPN) to generate RoIs from massive anchor boxes and then focus on these regions with fixed size after adopting RoI Pooling [2] or RoI Align [26].

Therefore, the advantage of one-stage detectors is the fast detection speed but the disadvantage is naturally the low detection precision. To ameliorate the existed weakness, researchers mainly focusing on balancing the ratio between negative and positive samples [27], detecting multi-scale objects from feature maps with different levels [4], and introducing long-distance dependencies among local features [28].

2.2. Transformer in CV

In NLP, the success of the Transformer network can be attributed to regard building the multiple long-distance dependencies as the main task and avoid the influence of distance among time sequence information [13]. The attention operation is computed as follow:

$$\text{Attention}(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}) = \text{softmax}\left(\frac{\boldsymbol{q}\boldsymbol{k}^T}{\sqrt{d_{\text{head}}}}\right)\boldsymbol{v} \qquad (1)$$

Where $\boldsymbol{q}$, $\boldsymbol{k}$, and $\boldsymbol{v}$ are vectors which indicate the query, key, and variable of input, respectively. d_{head} represents the number of channel information for each attention head in Multi-head SAN (MSAN).

While these kinds of dependencies are also needed in visual elements as massive features are extracted within local regions based on convolution operations. Therefore, Vision Transformer (ViT) [14] is proposed for the task of image recognition by dividing a whole image into non-overlapped patches and treat each patch as an embedding vector. However, the lack of inductive biases inherent to CNN leads to the intensive requirement of data, training tricks, and model parameters. Hence, the following proposed methods begin to incorporate the inductive biases into the different components of their vision Transformer networks. DeiT using the knowledge distillation method which considers the pretrained CNN as the teacher with multiple training strategies [15]. Swin-T imitates the calculation of CNN by only focusing on features within progressively large window size and adjusts the window location to build the complete long-distance dependencies indirectly [16]. ConViT tries to modify the structure of SAN to be generalized to convolution operation [17]. LocalViT incorporates convolution operation into MLP with a slight increase of parameters [18].

2.3. Unreasonable Predicted Boxes Problem

Since the complexity of samples, the detection bias results in massive false negative and false positive predicted boxes (i.e., precise boxes have low results and biased boxes yet own high results). To resolve the bias existed in many detectors, the work in [22] incorporated a new network branch which is responsible for outputting the variances related to the distribution of the predicted boxes and adopts variance voting to adjust the coordinates of predicted boxes. In addition, the work in [21] proposed the Intersection over Union (IoU) network branch with IoU prediction loss which shares the parameters with the regression branch. Moreover, a mutual non-linear combination method with different importance between the predicted IoU result and the classification score is proposed in their work to greatly elevate the low results for both prediction branches. However, these methods not only increase the process of training and inferring but also the model parameters.

As false negative and false positive predicted boxes are two boxes with conflict properties, our method aims to find these false negative predicted boxes to elevate the detection ability for better capture objects with nearly free cost. We hence propose a simple and efficient non-linear combination method to focus on boosting the result of false negative predicted boxes and avoiding the increasing result of false positive ones at the same time without adding extra network branch and training process compared with [21] and [22].

3. TOLO with Non-linear Combination

In this section, we firstly introduce TOLO network architecture in details which can be divided into backbone, detection neck and different detection heads and then introduce our non-linear combination method in the second part.

3.1. TOLO

TOLO is a creative real-time object detector which combines the inductive bias of CNN and globality of vision Transformer network to precisely detect object with high speed. Our model can be mainly divided into three parts called CNN backbone, FFN and LTHs. When images go through the network, the backbone in charge of extracting their features from local regions and transfers the inductive biases of CNN to the LTHs. Then, the proposed detection neck focuses on combining the features from different layers for the sake of incorporating high-resolution as well as high-semantic properties. Finally, different LTHs are responsible to efficiently build multiple long-distance dependencies among local features and find objects with various scales. The pipeline of our proposed model is shown in Fig. 1.

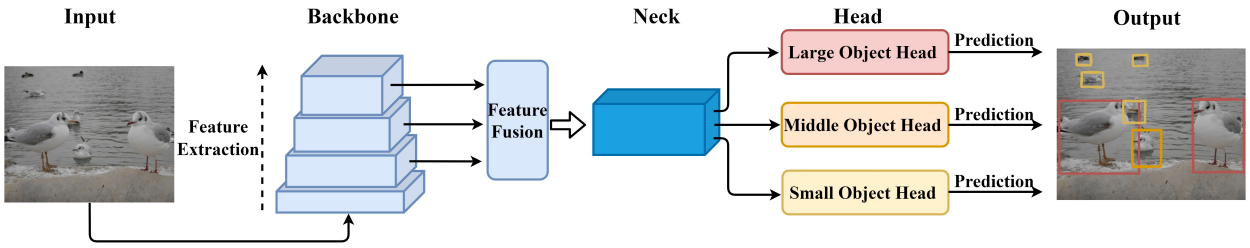

Figure 1: The pipeline of TOLO. It is composed of CNN backbone, Feature Fusion Neck and Lite Transformer detection heads for objects with different sizes.

In the structure of YOLOv3 [4] and RetinaNet [28], the detection neck is built by the Feature Pyramid Network (FPN) with concatenation and addition mode, respectively. Although they achieve competitive results, two problems that existed in these structures are the inconsistent connection among different layer features and the massive increase of computational complexity, which leads to the limited information that can be learned for different detection heads and the decreasing of model efficiency, respectively.

To resolve these problems above, we propose Feature Fusion Neck (FFN). As can be seen from Fig. 2, we focus on the last three network layers with different sizes. To the features in the last two layers, we use depth-wise and point-wise convolution operations to squeeze the channel dimension by half because the operations need fewer parameters and FLOPs than the traditional convolution operation [29]. It is worthy to note that the number of channels for squeezed features is related to the original ones instead of a small fixed number in [30]. Therefore, we suggest this operation is not only better to avoid the substantial loss of semantic information especially in deep layers but also efficient as the computational complexity is sufficiently decreased.

After squeezing, these features are upsampled with bilinear interpolation to make the spatial size same as the first layer features. Then, features in different layers are directly concatenated together followed by fusion operation which is also involved in depth-wise as well as point-wise convolutions to avoid the massive increase of computational complexity. Therefore, each layer features can combine with others in the same path instead of crossing multiple layers.

A novel vision Transformer network, Lite Transformer (LT), is proposed for real-time object detection. Distinguish from the previous variants of vision Transformer networks, the characteristic of LT is lightweight, which means the speed of training and inferring is fast. The comparisons of model parameters and FLOPs among different vision Transformers are shown in Table I. Here we assume the width, height and the number of channels about the input is 52, 52, and 256, respectively. The patch size and the number of long-distance dependencies we set for all models are 4. It can be apparently noticed that our LT can effectively decrease the requirement of model parameters and FLOPs, which means it can be well applied to elevate the model efficiency.

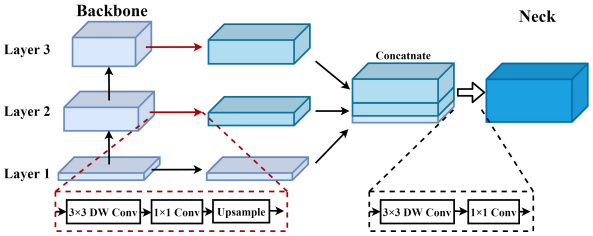

Figure 2: The model structure of Feature Fusion Neck. Here we focus on the last three network layers with different sizes.

Table 1: Comparisons of parameters and FLOPs among different vision Transformers

Model	Params (M)	FLOPs (G)
PVT [19]	2.28	0.66
Local ViT [18]	1.24	0.72
ViT [14]	1.23	0.71
Swin-T* [16]	1.05	2.21
LT	**0.71**	**0.23**

* Here we only compute a Transformer block without shifting and the window size is 4.

Specifically, LT can be divided into three different parts in Fig. 3. The first part concentrates on dividing the input features into non-overlapped patches and each patch will be added with learnable positional embedding to ensure its unique location. Lightweight convolution operation (depth-wise with point-wise convolution operation) will be considered to decrease the computational complexity of the Transformer during the process of dividing. Moreover, we incorporate GELU [31], a non-linear activation function calculated in Eq. (2) to

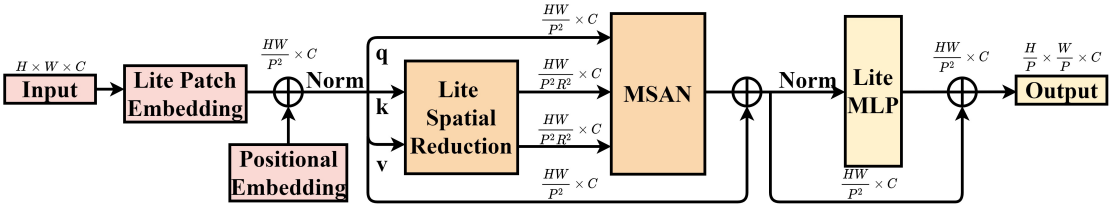

Figure 3: The structure of Lite Transformer. It is composed of three lightweight components that focus on generating embeddings, computing long-distance dependencies among local features, and enlarging the capacity of the model, respectively.

strengthen the representation of complex knowledges.

$$GELU(x) = 0.5x\left(1 + \tanh\left(\sqrt{\frac{2}{\pi}}\left(x + 0.044715x^3\right)\right)\right) \tag{2}$$

Algorithm 1: Lite Transformer

Input: F is the input features with fixed size;
N is the number of patches;
P is a hyperparameter about patch size;
PE is a set of positional embedding
Output: O is a set of long-distance dependencies among local features
compute a set of patch vectors V from F in Lite Patch Embedding module;
for *each patch* $i \in [1, N]$ **do**
  compute patch embedding E_i: $E_i = LN(GELU(V_i) + PE_i)$;
  compute query q_i: $q_i = W_q \times E_i$;
end
compute E' by reducing the number of patch embeddings in Lite Spatial Reduction module;
for *each path* $j \in [1, N/P^2]$ **do**
  compute key k_j: $k_j = W_k \times E'_j$;
  compute variable v_j: $v_j = W_v \times E'_j$;
end
compute the long-distance dependencies S based on Eq. (1) in MSAN;
for *each path* $i \in [1, N]$ **do**
  $S_i = S_i + E_i$;
  $S_i = LiteMLP(LN(S_i)) + S_i$
end
return O by reshaping S

After getting the patch embeddings added with positional embeddings, in the second part, the multiple long-distance dependencies will be built in MSAN after getting the queries, keys, and variables of patch embeddings. It can be apparently noticed that we use lite spatial reduction in both the key and variable branches. The aim is to decrease the computational complexity by compressing their spatial sizes with the lightweight non-overlapped convolution operation. The complexity ratio of self-attention operation between LT and ViT is:

$$\frac{\Omega(MSAN_{LT})}{\Omega(MSAN_{ViT})} = \frac{\left(\frac{2N^2C}{P}\right)}{2N^2C} = \frac{1}{P} \tag{3}$$

Where P, N, and C represent the times of spatial reduction, the number of patches and channels, respectively. Here we omit the complexity of computing $\boldsymbol{q}$, $\boldsymbol{k}$, $\boldsymbol{v}$, and the final output as they are the same for both vision Transformer networks. The lite spatial reduction can decrease the complexity by P times. Moreover, compare with Pyramid Vision Transformer (PVT) [19] which also considers decreasing the spatial size. LT needs fewer parameters and FLOPs according to Eq. (4):

$$\frac{k^2M + MN}{k^2MN} = \frac{\frac{WH}{k^2}\left(k^2M + MN\right)}{\frac{WH}{k^2}k^2MN} = \frac{1}{N} + \frac{1}{k^2} \tag{4}$$

Where k, W, H, M, and N, indicate the size of convolution kernel, input width and height, the number of former and latter channel dimensions, respectively. The first and second items are the ratio of parameters and FLOPs between LT and PVT in the branch of computing keys and variables. 4.

Lastly, MLP with shortcut connection will be used to avoid the degradation of expressive power for SAN with the increasing of network depth [31]. Nevertheless, massive parameters that existed in MLP also slow down the efficiency of network. Hence, our proposed MLP in LT replaces the original large hidden layer with the hidden layer that the size is same as the input, the ratio of their computational complexity can be computed as:

$$\frac{\Omega(MLP_{LT})}{\Omega(MLP_{ViT})} = \frac{2NC^2}{8NC^2} = \frac{1}{4} \tag{5}$$

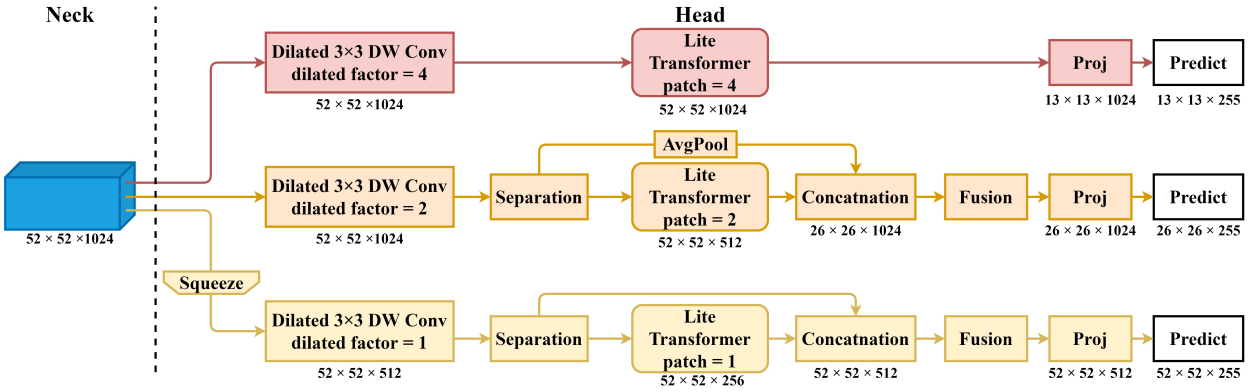

Figure 4: The model structure of different Lite Transformer Heads. Here we assume that the feature size of detection neck is 52×52×1024 and the number of detection categories is 80.

Where N and C represent the number of patches and channels, respectively. It can be observed that the MLP in LT will save the parameters by four times compared with the original one. Continue to squeeze the hidden layer can further decrease the complexity but we find the detection performance will degrade a lot. Lastly, we use Algorithm 1 to summarize how Lite Transformer work for the input features with fixed size.

Our detector consists of three different LTHs which focuses on the large, middle, and small size of objects, respectively. For all heads, LT will be used to efficiently mine the long-distance dependencies among features. Moreover, many previous experiments have already proved that different scale objects need information with different properties [30]. Specifically, small objects are sensitive to the resolution information but large ones are more likely to need high-semantic information, which means the channel dimension should be large. Therefore, it is natural to believe that the structure of each head should have its property. The structures of different heads are shown in Fig. 4.

As can be seen from the figure, the dilated depth-wise convolution will firstly be set to further enlarge the receptive field of feature for different heads and each head has a different patch size in LT. In our implementation, the dilated factor and patch size we set are 4, 2, and 1 for large, middle, and small heads, respectively.

To the large object head, the abundance of channel information from FFN was considered and the patch size in LT was set to large. To the middle and small object head, since the local changes of the image become more and more important [30], we hence gradually focus on the spatial information from FFN and the patch size is set to small. Moreover, we suggest it is also vital to incorporate local information in these two heads. Therefore, in the separation operation, we divide the features from FFN into two parts which the channel dimension of the divided features is half of the original one. One part goes through LT and the other part concatenates with the outputs from LT. Then, fusion operation is responsible to fuses the concatenation features with 1×1 convolution kernels followed by the Leaky ReLU activation function. It is worth to note that in the small object head, squeeze operation will firstly be used to decrease the channel dimension of features from FFN, and in the middle object head, AvgPool operation will be used to compress the spatial size of local features. Finally, projection operation adjusts the channel information to adapt to the predicted results.

3.2. Non-linear Combination

Since the existing of massive false negative predicted boxes decreases the ability of capturing objects, we try to verify whether there can be a simple and efficient way to mine the potential correct predicted boxes without introducing the extra model parameters and prolonging the training process. We therefore investigate the inferring stage and find the combination between object confidence and classification score in TOLO is direct multiplication which is calculated as:

$$R = C \cdot S \tag{6}$$

Where R, C and S indicate the combination result, object confidence and classification score, respectively.

Following this observation, we further devise a non-local combination method which can be computed as follow:

$$R = \log_3(1 + \alpha \cdot C) \cdot S \tag{7}$$

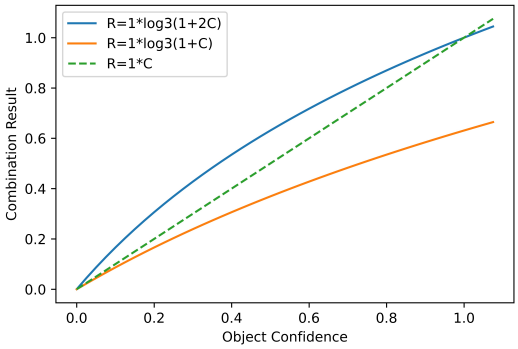

Figure 5: Comparisons between linear and non-linear combination functions.

Where α is a hyperparameter which controls the result of predicted boxes.

In Fig. 5, to show different combination functions clearly, we analyze the combination result between the object confidence and the classification score by assuming the classification score of the most probable category is one. The dashed line seen as a reference is a standard linear combination. Compared with Eq. (6), it can be apparently noticed that the object confidence in Eq. (7) will be changed non-linearly and the predicted boxes with higher confidence also own stronger suppression when α equals 1. When α equals 2, too low confidence will not change the final result a lot, which means it can avoid the influence of true negative predicted boxes (i.e. boxes with low IoU and low predicted result). However, the combination result with relatively low confidence will be elevated a lot within the range from 0.4 to 0.6. Therefore, if there are massive potential correct predicted boxes (i.e., false negative predicted boxes), increase the value of α will help to elevate their combination results.

There is a potential problem that the result of false positive predicted boxes will also be increased. However, according to our observation from the predicted results, we found that compare with false negative predicted boxes, there are many false positive and true positive predicted boxes own relatively high classification score but the former predicted boxes have relatively low object confidence. That is why the non-linear function we set is only for the object confidence as it can elevate the combination result of false negative predicted boxes and avoid the increasing of false positive ones at the same time.

4. Performance evaluation

In this section, we evaluate the effectiveness of the proposed object detector and the non-linear combination method on three datasets: the PASCAL VOC 2007, the PASCAL VOC 2012 [23], and the MS COCO 2017 dataset [24]. We firstly introduce the information of different datasets, the relevant metrics of evaluation, and the training details. Then, the corresponding experiments about TOLO and our non-linear combination method will be described.

4.1. Datasets, Evaluation Metrics and Training Strategies

The PASCAL VOC 2007 and 2012 datasets have 9, 962, and 22, 531 images for 20 object classes, respectively. These two datasets are divided into train, validation, and test sets. Here we choose the trainval set (5, 011 images for 2007 and 11, 540 images for 2012) to train our detector and follow the standard PASCAL VOC protocol, i.e., the mAP as the evaluation metric to test our detector on the test sets.

The MS COCO 2017 dataset has 80 object classes and is divided into *train*-2017, *val*-2017, and *test-dev* sets, respectively. We train our model on the MS COCO *train*-2017 set (about 115K images) and test it on the *val*-2017 and *test-dev* set (about 5K and 20K images), respectively. For evaluation, we use three metrics AP_{50}, AP_{70} and AP, which are the standard PASCAL criterion, i.e., IoU>0.5, IoU>0.7 and the standard MS COCO criterion, i.e., computing the average of mAP for IoU $\in$ [0.5 : 0.05 : 0.95], respectively. Moreover, objects with small, middle and large sizes will also be evaluated by adopting AP_{small}, AP_{middle}, and AP_{large}, respectively.

As for the training strategies, Stochastic Gradient Descent (SGD) is used to optimize our model by setting the initial learning rate as 0.001 trained on 2 GPUs (GTX 3090). The cosine learning rate schedule is also set from 0.001 to 0.00001. The weight decay and momentum is 0.0005 and 0.9, respectively. Moreover, according to [32], some training tricks are applied to avoid the overfitting and improve the model generalization such as mixup and label smoothing.

The batch size for all datasets we set is 24. The image size is 448 with multi-scale training (320, 352, 384, 416, 448, 480, 512, 544, 576, 608) for 50 epochs on the PASCAL VOC and 320 with three scales training (320, 352, and 384) for 300 epochs on the MS COCO, respectively.

Table 2: Comparisons among detection heads with different combinations of LT and CNN

Architecture	mAP@50	Params (M)
Only convolutional sets	82.2	55.54
Only LTs	82.8	**55.09**
Local features in the middle head	82.9	56.44
Local features in the small head	83.1	55.43
Local features in the both head	**83.2**	56.79

Table 3: Comparisons among TOLO equipped with different detection neck networks

Neck	Params (M)	mAP@50	mAP@75
FPN (256) [30]	2.23	82.4	45.1
FPN (512)	8.00	83.1	48.6
Concatenated FPN [2]	14.94	82.9	**50.2**
FFN	**1.41**	**83.2**	49.7

4.2. Experimental settings

In this part, we discuss the different combinations in TOLO, the effectiveness of FFN, and the suitable parameter settings in LTHs and perform the relevant experiments on the PASCAL VOC 2007 and 2012 datasets. Then we compare TOLO with other state-of-the-art methods on the PASCAL VOC and the MS COCO datasets, respectively. Finally, some quantitative results are illustrated to clearly show the responses of different LTHs.

We firstly analyze the effectiveness of combining CNN and our proposed vision Transformer network. As we stated above, the inductive biases of CNN such as locality can be adapted to image features. However, this locality will lead to the weakness of building long-distance dependencies among local features. Vision Transformer network is adept at mining long-distance dependencies but at the cost of more data or training tricks to achieve strong generalization. Therefore, the ablation study for the combinations of CNN and LT is shown in Table 2. We can see that the best performance is generated when they combine reasonably.

Specifically, for the first row in Table 2, we replace all LTs with convolution sets and keep the converted model owns comparable parameters with the original one. However, it can be apparently observed that the performance drops to 82.2% even the converted model has slightly larger parameters, which means the long-distance dependencies among features are important for the task of object detection and massive stacks of convolution sets can not generate long-distance dependencies well compared with LT. Moreover, we also notice that only use LTs among detection heads also can not get the best performance compared with the model incorporated local feature branch, especially in the small object head. This suggests that local features are also useful to help the detector find small and unobvious objects.

We then evaluate the contribution of FFN. Results are reported in Table 3, where we compare with FPN [30] and concatenated FPN [4]. It can be noticed that our FFN can achieve good performance with only occupying fewer parameters.

To be specific, the better performance of FPN (512) compared with FPN (256) indicates the importance of abundant features in the neck. Our FFN is fewer than concatenated FPN nearly 10 times of parameters but acquires better performance in mAP@50 and comparable performance in mAP@75. We suggest this detection performance can be attributed to the features from each layer can combine with features from other layers by the same path. On the other hand, due to the fact that shallow layers contain high-resolution features and deep layers contain high-semantic features, which means the fused features can own both characteristics simultaneously for different detection heads to analyze.

To further analyze the proposed detector, we next focus on the part of detection heads and find the suitable parameter settings about LTHs. In particular, the number of attention heads in MSAN, the size of MLP, and the times of spatial reduction in the Lite Spatial Reduction module are all significant for detection heads to achieve good performance.

The work in [33] has already indicated that there is a tradeoff between the attention heads and the number of channel information in MSAN. Hence, finding a suitable setting is important for our model to achieve good performance. As shown in Table 4, these experimental results indicate that multiple attention heads can build different long-distance dependencies among features to help our model pay attention to different interesting regions on a whole feature map. To the LTHs, the best performance we achieve is when the number of attention heads equals 2. On the other hand, there is not a positive correlation between the attention heads and the performance as the capacity of information for each head is dropped a lot when increasing the number of attention heads.

MLP is also useful for vision Transformer networks to avoid the degradation of long-distance dependencies expression [31]. Here we analyze the influence of MLP with different hidden layer sizes to LTHs. In Table 5, we find that with the increasing of hidden layer size in MLP, the detection performance also increases and at-

Table 4: Comparisons among LTHs with different attention heads

Head	Dim/head			mAP@50
	Large	Middle	Small	
1	1024	512	256	83.18
2	512	256	128	**83.21**
4	256	128	64	83.17
8	128	64	32	83.11
16	64	32	16	83.04

Table 5: Comparisons among LTHs with different scales of hidden layer in MLP

Hidden scale	Params (M)	mAP@50
0.25×	**54.72**	83.05
0.5×	55.41	83.10
1×	56.79	**83.21**
2×	59.54	83.16
4×	65.05	82.81

tains 83.21% when the layer size equals the input layer size. However, the detection performance begins to drop when the size of the hidden layer continues to increases. We suggest the smaller hidden size also only owns the lower expression power of long-distance dependencies among local features but the higher hidden size results in the overfitting of MLP, especially in the large object heads as the number of channel information has already large enough.

Lite Spatial Reduction can reduce the FLOPs by slightly increasing the model parameters. As shown in Table 6, compared with the LT without the spatial reduction, the default setting can help the detector save 0.63 G FLOPs (43.39G vs 44.02G) by increasing 0.34 M (56.45M vs 56.79M) parameters. More importantly, we find this operation can further bring the gain of performance especially when expand the reduction scale. Specifically, by adjusting the scales of spatial reduction in the middle or small detection head and fixing the rest of the ones, we can find that the larger scale we set, the better mAP@50 we will get. Also, there are some similar characteristics for mAP@75. These experimental results indicate that incorporating the local operation into the self-attention mechanism can boost the model performance and further demonstrate the importance of the combination between CNN and vision Transformer network.

We then compare TOLO with other state-of-the-art detectors included one-stage or two-stage categories. We can see from Table 7 that our model can produce 83.2% mAP@50 and 79.3% mAP@75 on the

Table 6: Comparisons among LTHs with different scales of spatial reduction

Head		mAP@50	mAP@75	Params (M)	FLOPs (G)
Middle	Small				
1	1	82.65	49.06	**56.45**	44.02
1	2	82.79	48.81	56.52	43.77
1	4	83.01	48.61	56.52	43.65
2	1	82.77	49.16	56.71	43.77
2	2	83.06	49.58	56.78	43.51
2	4	**83.21**	**49.68**	56.79	**43.39**

Table 7: Detection results on PASCAL VOC datasets for two-stage and one-stage detectors

Method	Backbone	mAP@50	
		VOC 2007	VOC 2012
two-stage			
Fast R-CNN [1]	VGG-16	70.0	68.4
OHEM [39]	VGG-16	74.6	71.9
HyperNet [40]	VGG-16	76.3	71.3
Faster R-CNN [2]	ResNet-101	76.4	73.8
ION [38]	VGG-16	76.5	76.4
MR-CNN [41]	VGG-16	78.2	73.9
R-FCN+ASM [37]	ResNet-101	81.8	78.3
one-stage			
RON [34]	VGG-16	75.4	73.0
DSSD [35]	ResNet-101	78.6	76.3
SSD [3]	VGG-16	79.8	78.5
RefineDet [36]	VGG-16	80.0	78.1
YOLOv3	DarkNet-53	82.3	78.4
TOLO	DarkNet-53	**83.2**	79.3*

* http://host.robots.ox.ac.uk:8080/anonymous/VTTBNT.html

VOC 2007 test set and VOC 2012 test set, which surpasses other one-stage detectors, e.g., RON [34], DSSD [35], and RefineDet [36]. Similarly, comparing with the two-stage detectors such as Faster R-CNN [2] ,R-FCN+ASM [37] and ION [38], our detector still performs better performance with lower input size.

Moreover, according to Table 7, we also reimplement YOLOv3 with the same training strategies as TOLO. It can be noticed that our detector can outperform YOLOv3 over 0.9% and 0.8% on PASCAL VOC 2007 and 2012 respectively with fewer parameters (56.79 M vs 61.63 M).

To further validate the proposed detector, we conduct experiments on the MS COCO 2017 dataset [15]. As pointed out in [30], the input size significantly influences detection performance as the inputs with high resolution can make the detectors see small objects more

Table 8: Detection results on MS COCO *val*-2017 set with different combination methods

Combination	ms/img	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	AR_1	AR_{10}	AR_{100}	AR_S	AR_M	AR_L
$S * C$	14.63	36.2	56.1	38.8	16.0	39.7	51.5	28.8	46.4	50.8	28.4	56.5	66.8
$S * \log_2(1+C)$	14.89	36.2	56.3	38.9	16.1	39.7	51.5	28.9	46.5	51.1	28.7	56.9	67.0
$S * \log_3(1+2C)$	14.95	36.2	56.3	38.9	16.1	39.7	51.5	28.9	46.6	51.3	29.0	57.2	67.2
$S * \log_4(1+3C)$	15.06	36.1	56.3	38.9	16.1	39.7	51.4	28.9	46.6	51.4	29.1	57.2	67.2
$S * C^{0.6}$	15.29	36.1	56.3	38.9	17.0	39.7	51.4	28.9	46.9	52.2	30.6	58.3	67.7
$S * C^{0.5}$	16.43	36.1	56.3	38.8	16.9	39.6	51.2	28.9	46.9	52.2	30.7	58.4	68.1
$S * C^{0.4}$	18.09	35.9	56.2	38.7	16.8	39.5	50.9	28.9	46.9	52.2	30.7	58.4	68.1

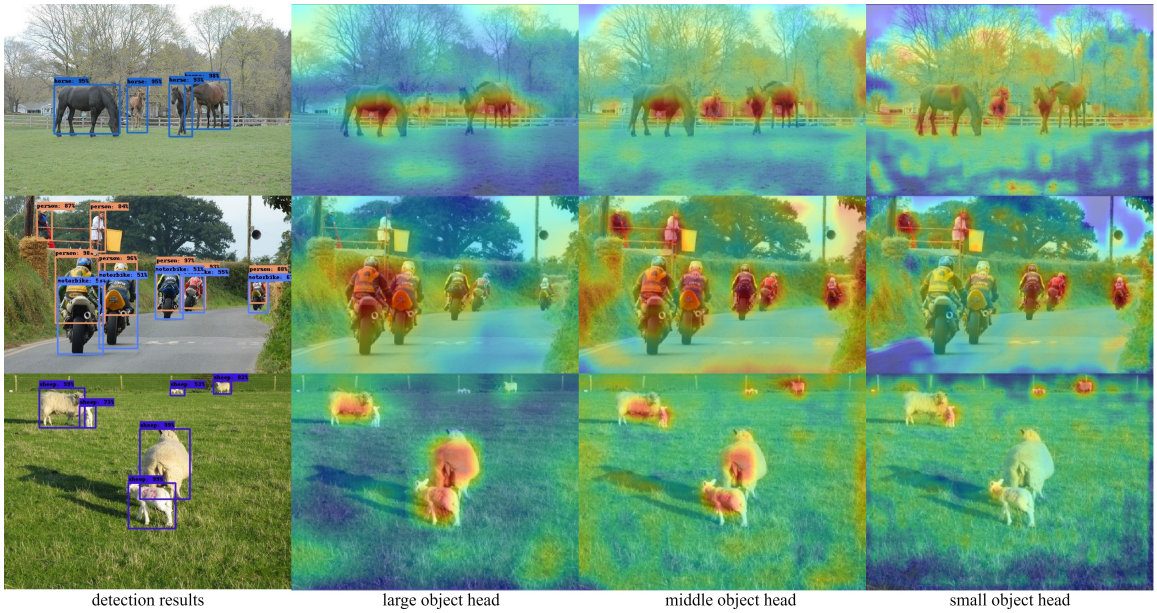

Figure 6: Qualitative examples for different Lite Transformer Heads.

Table 9: Detection results on MS COCO 2017 *test-dev* set for real-time detectors

Method	Size	FPS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
SSD [3]	300	43	25.1	43.1	25.8	6.6	25.9	41.4
RefineDet [36]	320	39	29.4	49.2	31.3	10.0	32.0	44.4
YOLOv3 [4]	416	35	31.0	55.3	32.3	15.2	33.2	42.8
EFGRNet [42]	320	48	33.2	53.4	35.4	13.4	37.1	47.9
M2Det [43]	320	33	33.5	52.4	35.6	14.4	37.6	47.6
HSD [44]	320	40	33.5	53.2	36.1	15.0	35.0	47.8
DAFS [45]	512	35	33.8	52.9	36.9	14.6	37.0	47.7
EfficientDet [46]	512	63	33.8	52.2	35.8	12.0	38.3	**51.2**
LRF [47]	300	53	34.3	54.1	36.6	13.2	38.2	50.7
RFBNet-E [48]	512	30	34.4	55.7	36.4	**17.6**	37.0	47.6
TOLO	320	60	**36.6**	**57.0**	**39.7**	16.2	**39.2**	50.3

clearly and increase successful detection. Therefore, Table 9 shows the results on MS COCO *test-dev* set. We can see from Table 9 that TOLO with small input size (i.e. 320×320) can produce 36.6% AP with high inferring speed, which also outperforms state-of-the-art real-time detectors even their input size is much larger and keeps high inferring speed.

To better understand the response of each head in TOLO for an input image, we visualize some qualitative results. The detection results and the corresponding heatmaps are shown in Fig. 6. The first column is the input images with detection boxes and the rest of the columns from left to right indicate the heatmaps from the large, middle, and small detection heads, respectively. It can be apparently observed that each head has its interesting fields. Specifically, for the large object detection head, its interesting regions are focused on relatively large objects. However, for the small object detection head, the strong responses are mainly dis-

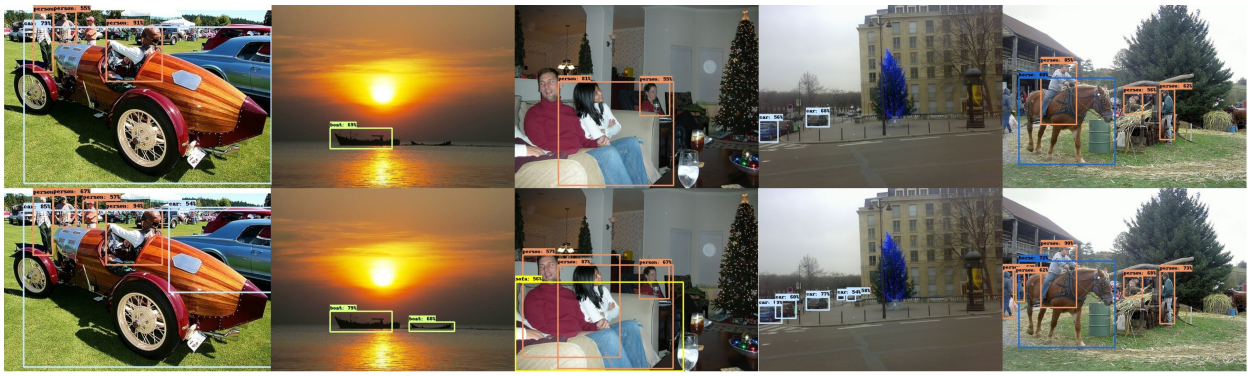

Figure 7: Qualitative examples by adopting linear combination method (top) and the proposed non-linear combination method (bottom).

tributed in the regions of relatively small objects.

4.3. Experiments on Non-linear Combination

In this part, we firstly compare various non-linear combination methods on the MS COCO *val*-2017 dataset and then validate the effectiveness of our selected method on PASCAL VOC 2007 and 2012 datasets. Finally, some qualitative results are visualized to further prove the effectiveness. The comparisons of detection results among different combination functions are shown in Table 8.

According to the experimental results, we observe that $S * \log_3(1 + 2C)$ can get better detection performance than the linear combination method and a good tradeoff between the detection performance and the inferring speed compared with other non-linear combination methods without increasing model parameters and training process. Moreover, there is slightly dropped performance appeared in some methods such as $S * C^{0.4}$, which we suggest the true negative predicted boxes also get too large improvement.

We further conduct the experiments on PASCAL VOC 2007 and 2012 datasets. The detection results of the reimplemented YOLOv3 and TOLO are shown in Table 10. It can be also noticed that our combination method can boost the detection performance for both detectors, which indicates the effectiveness of our non-linear combination method..

We then illustrate some detection results on Fig. 7. It can be apparently observed that the predicted boxes with high results can not get great improvement but the potential correct predicted boxes with low results can be elevated greatly.

Table 10: Comparisons between YOLOv3 and TOLO with/without the proposed non-linear combination method on PASCAL VOC datasets

Model	Params (M)	mAP@50	
		VOC 2007	VOC 2012
YOLOv3 without NC	61.63	82.3	78.4
YOLOv3 with NC	61.63	82.5	78.6
TOLO without NC	56.79	83.2	79.3
TOLO with NC	56.79	**83.4**	**79.5***

NC means non-linear combination
* http://host.robots.ox.ac.uk:8080/anonymous/I1QETG.html

5. Conclusion and future work

In this paper, we proposed a new real-time object detector called TOLO, which is composed of CNN backbone, FFN, and LTHs. The CNN backbone transfers the inductive biases to different LTHs. Then, the FFN plays a role in the transition and provides the detection heads with abundant features owned high-resolution and high-semantic properties. Finally, LTHs efficiently build multiple long-distance dependencies among local features and detect complex objects with less memory overhead. Besides, in the inferring stage, the proposed non-local combination method can further elevate the detection performance of capturing objects. We carry out experiments on the PASCAL VOC 2007, the PASCAL VOC 2012, and the MS COCO 2017 datasets and achieve 83.2% mAP@50, 79.3% mAP@50, and 36.6% *AP*, respectively, which are better than YOLOv3 with larger model parameters and other state-of-the-art real-time detectors. Moreover, our experimental results have also demonstrated the proposed non-linear combination method can further elevate the detection performance on different datasets.In the future, TOLO will achieve higher detection performance by extending to the de-

sign of backbone and higher efficiency by adjusting the way of building long-distance dependencies in LT.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61971079); the Brunel University London BREIF Award (No. 11937115); the National Key Research and Development Program of China (No. 2019YFC1511300); the Basic Research and Frontier Exploration Project of Chongqing (No. cstc2019jcyj-msxmX0666) and the Innovative Group Project of the National Natural Science Foundation of Chongqing (No. cstc2020jcyj-cxttX0002).

References

[1] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[2] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, IEEE transactions on pattern analysis and machine intelligence 39 (6) (2016) 1137–1149.

[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, Ssd: Single shot multibox detector, in: European conference on computer vision, Springer, 2016, pp. 21–37.

[4] J. Redmon, A. Farhadi, Yolov3: An incremental improvement, arXiv preprint arXiv:1804.02767 (2018).

[5] H. Law, J. Deng, Cornernet: Detecting objects as paired keypoints, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 734–750.

[6] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, Q. Tian, Centernet: Keypoint triplets for object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6569–6578.

[7] Z. Tian, C. Shen, H. Chen, T. He, Fcos: Fully convolutional one-stage object detection, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 9627–9636.

[8] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7794–7803.

[9] W. Luo, Y. Li, R. Urtasun, R. Zemel, Understanding the effective receptive field in deep convolutional neural networks, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 4905–4913.

[10] Z. Nan, J. Peng, J. Jiang, H. Chen, B. Yang, J. Xin, N. Zheng, A joint object detection and semantic segmentation model with cross-attention and inner-attention mechanisms, Neurocomputing 463 (2021) 212–225.

[11] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, Cbam: Convolutional block attention module, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.

[12] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 5998–6008.

[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929 (2020).

[15] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, Training data-efficient image transformers & distillation through attention, in: International Conference on Machine Learning, PMLR, 2021, pp. 10347–10357.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, arXiv preprint arXiv:2103.14030 (2021).

[17] S. d'Ascoli, H. Touvron, M. Leavitt, A. Morcos, G. Biroli, L. Sagun, Convit: Improving vision transformers with soft convolutional inductive biases, arXiv preprint arXiv:2103.10697 (2021).

[18] Y. Li, K. Zhang, J. Cao, R. Timofte, L. Van Gool, Localvit: Bringing locality to vision transformers, arXiv preprint arXiv:2104.05707 (2021).

[19] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pyramid vision transformer: A versatile backbone for dense prediction without convolutions, arXiv preprint arXiv:2102.12122 (2021).

[20] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, Y. Zhou, Transunet: Transformers make strong encoders for medical image segmentation, arXiv preprint arXiv:2102.04306 (2021).

[21] S. Wu, X. Li, X. Wang, Iou-aware single-stage object detector for accurate localization, Image and Vision Computing 97 (2020) 103911.

[22] Y. He, X. Zhang, M. Savvides, K. Kitani, Softer-nms: Rethinking bounding box regression for accurate object detection, arXiv preprint arXiv:1809.08545 2 (3) (2018).

[23] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, International journal of computer vision 88 (2) (2010) 303–338.

[24] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: European conference on computer vision, Springer, 2014, pp. 740–755.

[25] S. Zhang, C. Chi, Y. Yao, Z. Lei, S. Z. Li, Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 9759–9768.

[26] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[28] J.-S. Lim, M. Astrid, H.-J. Yoon, S.-I. Lee, Small object detection using context and attention, arXiv preprint arXiv:1912.06319 (2019).

[29] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861 (2017).

[30] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.

[31] Y. Dong, J.-B. Cordonnier, A. Loukas, Attention is not all you need: Pure attention loses rank doubly exponentially with depth, arXiv preprint arXiv:2103.03404 (2021).

[32] Z. Zhang, T. He, H. Zhang, Z. Zhang, J. Xie, M. Li, Bag of freebies for training object detection neural networks, arXiv preprint arXiv:1902.04103 (2019).
[33] H. Touvron, M. Cord, A. Sablayrolles, G. Synnaeve, H. Jégou, Going deeper with image transformers, arXiv preprint arXiv:2103.17239 (2021).
[34] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, Y. Chen, Ron: Reverse connection with objectness prior networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 5936–5944.
[35] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, A. C. Berg, Dssd: Deconvolutional single shot detector, arXiv preprint arXiv:1701.06659 (2017).
[36] S. Zhang, L. Wen, X. Bian, Z. Lei, S. Z. Li, Single-shot refinement neural network for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4203–4212.
[37] K. Wang, L. Lin, X. Yan, Z. Chen, D. Zhang, L. Zhang, Cost-effective object detection: Active sample mining with switchable selection criteria, IEEE transactions on neural networks and learning systems 30 (3) (2018) 834–850.
[38] S. Bell, C. L. Zitnick, K. Bala, R. Girshick, Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2874–2883.
[39] A. Shrivastava, A. Gupta, R. Girshick, Training region-based object detectors with online hard example mining, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 761–769.
[40] T. Kong, A. Yao, Y. Chen, F. Sun, Hypernet: Towards accurate region proposal generation and joint object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 845–853.
[41] Z. Liu, J. Du, F. Tian, J. Wen, Mr-cnn: A multi-scale region-based convolutional neural network for small traffic sign recognition, IEEE Access 7 (2019) 57120–57128.
[42] J. Nie, R. M. Anwer, H. Cholakkal, F. S. Khan, Y. Pang, L. Shao, Enriched feature guided refinement network for object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9537–9546.
[43] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, H. Ling, M2det: A single-shot object detector based on multi-level feature pyramid network, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 33, 2019, pp. 9259–9266.
[44] J. Cao, Y. Pang, J. Han, X. Li, Hierarchical shot detector, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9705–9714.
[45] S. Li, L. Yang, J. Huang, X.-S. Hua, L. Zhang, Dynamic anchor feature selection for single-shot object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6609–6618.
[46] M. Tan, R. Pang, Q. V. Le, Efficientdet: Scalable and efficient object detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781–10790.
[47] T. Wang, R. M. Anwer, H. Cholakkal, F. S. Khan, Y. Pang, L. Shao, Learning rich features at high-speed for single-shot object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1971–1980.
[48] L. Deng, M. Yang, T. Li, Y. He, C. Wang, Rfbnet: deep multimodal networks with residual fusion blocks for rgb-d semantic segmentation, arXiv preprint arXiv:1907.00135 (2019).