Article

TIG-DETR: Enhancing Texture Preservation and Information Interaction for Target Detection

Zhiyong Liu [1,‡,*] , Kehan Wang [1,‡] , Changming Li [2] , Yixuan Wang [1] and Guoqian Luo [1]

[1] College of Information Science and Technology, Northeast Normal University, Changchun 130117, China
[2] Engineering Technology Development Center, Changchun Guanghua University, Changchun 130033, China
* Correspondence: liuzy452@nenu.edu.cn
‡ These authors contributed equally to this work.

Abstract: In practical applications, the detection of objects with various sizes is a common requirement for most detectors. The feature pyramid network (FPN) is widely adopted as a framework to address this challenge. The field is witnessing an increasing number of transformer-based target detectors due to the widespread adoption of transformer technology. This paper initially examines the design flaws in FPN and transformer-based target detectors, followed by the introduction of a new transformer-based approach called Texturized Instance Guidance (TIG-DETR) to address these issues. Specifically, TIG-DETR comprises a backbone network, a new pyramidal structure known as Texture-Enhanced FPN (TE-FPN), and an enhanced DETR detector.The TE-FPN is composed of three components: a bottom-up pathway for enhancing texture information in the feature map, a lightweight attention module to address confounding effects resulting from cross-scale fusion, and a standard attention module to enhance the final output features.The improved DETR detector utilizes Shifted Window based Self-Attention to replace the multi-headed self-attention module in DETR, thereby accelerating model convergence. Moreover, it incorporates an Instance Based Advanced Guidance Module to enhance instance perception in the image by employing a pre-local self-attentive mechanism for recognizing larger instances. By employing TE-FPN instead of FPN in Faster RCNN with Resnet-50 as the backbone network, we achieve a 1.9% improvement in average accuracy. TIG-DETR achieves an average accuracy of 44.1 with Resnet-50 as the backbone network.

Keywords: object detection; DETR; FPN; transformer;attention mechanism

1. Introduction

With the rapid development in the field of deep learning, significant progress has been made in target detection. Many advanced detectors based on CNN and transformer [28] have been proposed to drive the smooth development of target detection techniques. Among these detectors, FPN [10] stands out as it can enhance detector performance in a straightforward and effective manner by propagating semantic information to establish a CNN feature hierarchy IRecently, transformer technology has gained increasing attention, leading to the development of several detectors [21,22,27] that rely on transformers. These detectors have shown promising results compared to conventional CNN-based approaches. However, both the FPN structure and transformer-based detectors have some design flaws, which are described as follows:

Loss of texture information in FPNs. The classical FPN-based networks have greatly enhanced the performance of detection networks through multi-scale feature learning. Subsequent FPN structures [11–13,29], proposed based on this concept, have employed similar architectures to FPN. Multiple studies have shown that low-level features play a crucial role in recognizing larger instances. The abundant texture information present in the low-level feature map aids in target localization and precise framing. However, during the downsampling process, all features extracted from the backbone network inevitably suffer from a significant loss of texture information. This loss can potentially impact the accuracy of the location information acquired by the detection network.

 doi:10.20944/preprints202306.1084.v1

Confounding effect of cross-level feature fusion. During the cross-level feature fusion process, overlaying the upsampled feature map onto the original feature map leads to feature discontinuity and confuses the fused features. This phenomenon is referred to as the feature confusion effect [10]. The confounding effect becomes more pronounced as more feature maps are superimposed.

The limitation of interaction between windows in Shifted Window based Self-Attention. Utilizing a global attention mechanism on the feature map in a transformer-based detection network incurs a substantial computational burden. However, employing Shifted Window based Self-Attention significantly alleviates the computational complexity of the transformer. Nevertheless, the interaction between each window in the Shifted Window based Self-Attention is limited to its neighboring windows. This constraint imposes limitations and could potentially impair the model's ability to perceive larger objects.

Our proposed solution, TIG-DETR, addresses these shortcomings by incorporating a backbone network, a new FPN (TE-FPN) network, and a DETR-based detector.

Our contributions:

1. To address this, we propose constructing a bottom-up path within the backbone using the low-level feature map. Unlike the downsampling approach used in the backbone, our path aims to preserve maximum texture information in the feature map and integrates it with the features at the corresponding level in the top-down path of the FPN. As a result, this path encompasses both rich semantic and texture information. It is worth noting that a similar bottom-up path has been explored in prior works [11,25,26]. However, the downsampling path implemented through convolution still leads to significant loss of texture information.

2. We introduced a novel attention module called 'Feature-wise Attention' to address the feature fusion confounding effect in FPNs. This lightweight attention module is applied after the standard attention module to enhance the final features obtained from the SRS [12] feature fusion module.

3. To reduce the computational complexity of DETR, we replaced the multi-headed self-attention module with Shifted Window based Self-Attention [26]. Additionally, we incorporated feature maps from different stages of the backbone network to enhance the final feature map with texture information. The image undergoes segmentation and fusion after a multi-headed attention module, and is then restored to its original size. This approach enhances the texture information of the image and enables the model to perceive instances in the image prior to performing more detailed self-attention. It facilitates information interaction between each window and improves the model's ability to perceive large objects.

2. Related Work

FPN. Before the introduction of FPN, various approaches for feature processing existed, including featurized image pyramids, single feature maps, and Pyramidal feature hierarchy. SSD [30] utilized Pyramidal feature hierarchy, specifically focusing on hierarchical feature prediction goals, to enable different level features to learn the same semantic information. FPN [10] proposes a method for fusing features of different resolutions by element-wise addition of the feature map from each resolution with the up-sampled low-resolution feature map. This enhancement improves the features at different levels, and subsequent models have built upon the FPN foundation. PANet [11] introduced a bottom-up path enhancement to shorten the information path by utilizing the precise localization signal stored in low-level features, thereby improving the performance of the feature pyramid architecture. EfficientDet borrowed the TopDown-BottomUp concept from PANet and incorporated residual structures in each block to reduce optimization difficulties; Furthermore, the authors recognize that features from different layers possess varying semantic information. Directly summing these features can lead to sub-optimal problems. To address this, the authors introduce a learnable parameter in front of each layer of features to automatically determine their weights. Aug-FPN [12] proposes Soft ROI Selection, which involves pooling ROI features from different levels and fusing them to enhance the performance of the feature pyramid architecture. To mitigate the loss of texture information in high-level feature

 doi:10.20944/preprints202306.1084.v1

maps, Aug-FPN incorporates a residual enhancement branch specifically designed to enhance the texture information of these high-level feature maps. In CE-FPN [13], sub-pixel enhancement and attention-guided modules are employed in FPN to fully leverage the rich channel information of each level feature map, while minimizing the loss of channel information during the downscaling process.

Target detector. Traditional image target detection can be categorized into two main types: two-stage detectors, with Faster R-CNN [8] being the most representative example, and one-stage detectors such as YOLO [9], YOLO9000 [3] and YOLOV3 [20]. r-CNN [44]] demonstrated for the first time the significant improvement in target detection performance by using CNN on the PASCAL VOC dataset [4] compared to HOG-like feature-based systems. Fast R-CNN [1], proposed subsequently, overcomes the time-consuming aspect of R-CNN's SVMs [49] classification by employing ConvNet forward propagation for each region without redundant computation. Fast R-CNN [1]extracts features from the entire input image and passes them through the ROI pooling layer to obtain fixed-size features for subsequent classification and bounding box regression in the fully connected layer. Instead of extracting features for each region separately, Fast R-CNN extracts features once from the entire image, reducing both the processing time and the storage space required for a large number of features. Fast R-CNN [1] adopts selective search to propose RoIs, but this approach is slower and has the same running time as the detection network. In contrast, Faster R-CNN [8] introduces a new RPN (region proposal network) that is composed entirely of convolutional networks and efficiently predicts region proposals. The RPN shares the same set of common convolutional layers with the detection network, and the fully convolutional Mask R-CNN [15] optimizes the model by integrating low-level and high-level features to enhance the classification task. YOLO [9] pioneered one-stage target detection, and subsequent one-stage detectors have built upon its improvements. Generally, two-stage detectors achieve higher localization and target detection accuracy, while one-stage detectors offer faster inference speed. However, both types of detectors are influenced by post-processing steps like compressing redundant prediction results, anchor frame design, and heuristics for assigning target frames to anchor frames [17]. In contrast, DETR [22] achieves an end-to-end target detector by directly predicting without relying on intermediate methods.

Transformer. Transformer was initially introduced as a Seq2Seq [16] model designed for machine translation. Subsequent studies have demonstrated that a pre-trained Transformer-based model (PTM) [35] can achieve state-of-the-art performance on various tasks. Consequently, the Transformer has emerged as the preferred architecture for NLP tasks. Besides the NLP domain, the Transformer has gained significant adoption in areas such as computer vision, audio processing, etc. [36]. The Non-local Network [46] was the first to employ the self-attentive mechanism in the field of computer vision, achieving successful target detection. Several frameworks have been proposed in recent years [6,31,32]to enhance the Transformer and optimize it from various perspectives. Visual transformers [50] incorporate the Transformer into a CNN, enhancing the CNN network by allocating semantic information of the input image to different channels and closely correlating them through encoder blocks (referred to as VT blocks). VT blocks are employed as an alternative to partial convolution to improve the semantic modeling capacity of CNN networks. SWIN-T [6]introduced Shifted Window based Self-Attention, significantly reducing the computational complexity of the transformer when processing images. Funnel Transformer [42] employs a funnel-like encoder architecture that incorporates pooling along the sequence dimension to progressively decrease the length of the hidden sequence and then employs upsampling for reconstruction, effectively reducing FLOP and memory consumption. When employing the transformer in the computer vision (CV) domain, the feature space resolution is constrained, and the network encounters challenges in convergence during training. To address these issues, Zhu et al. [20] introduced Deformable DETR, which accelerates model convergence by directing the attention module to concentrate on a subset of key sampling points surrounding the reference.

Attention mechanism. Attention plays a crucial role in human perception of external information, as humans selectively concentrate on the most salient parts when processing information in a scene

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

to enhance the capture of relevant information [38]. RAM [47] integrates deep neural networks with an attention mechanism, enabling end-to-end updating of the entire network by iteratively predicting significant regions. This marks the first implementation of an attention mechanism in CNN networks. Numerous subsequent works have adopted comparable attention strategies. STN [48] predicts the spatial transformation by incorporating a sub-network that identifies significant regions in the input. SE-Net [8] introduces a compact module that enhances inter-channel relationships by utilizing global average pooling to compute attention across channels. GSoP-Net [45]addresses the limitation of using global average pooling alone in SENet for collecting contextual information, which restricts the modeling capacity of the attention mechanism. To overcome this, GSoP-Net proposes the global second-order pooling (GSoP) block to capture higher-order statistics while incorporating global contextual information. CBAM [9] incorporates global maximum pooling in addition to global average pooling, boosting the attention mechanism's response to maximum gradient feedback. Furthermore, the combination of spatial attention and channel attention demonstrates superior performance compared to channel attention alone. And adding spatial attention to channel attention verifies that using both is better than using channel attention alone. In our Feature-wise Attention, we introduce soft pooling [14] as a novel contextual information to provide distinct gradient feedback for individual features. This approach assigns different attention weights to different features, thereby enhancing the preservation of texture information in the image instances.

3. Materials and Methods

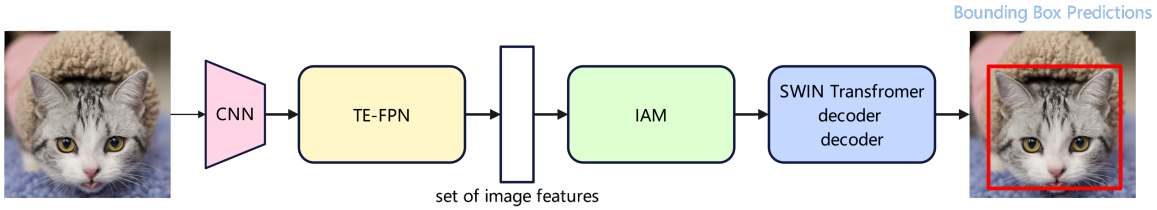

Figure 1. TIG-DETR comprises a backbone network, a new pyramidal structure known as Texture-Enhanced FPN (TE-FPN), and an enhanced DETR detector.

The proposed Texturized Instance Guidance DETR (TIG-DETR) architecture comprises a backbone, an FPN network, and a DETR-based detector. In order to enhance the model's localization capability, we introduce a bottom-up path in the FPN that retains the texture information of the feature maps and combine it with the pyramidal features produced in the top-down path of the FPN. This fusion results in a feature map that contains both abundant semantic and texture information. The SRS module is employed to integrate the features produced at each level of the feature pyramid, and Feature-wise Attention is utilized to enhance the features of the resulting output feature map. Within the DETR-based detection model, we employ local self-attention to improve the recognition performance of small object instances and facilitate faster model convergence. Additionally, we introduce a module to address the limitations of local attention in perceiving large object instances, thereby enhancing the model's capability to detect small object instances without compromising its performance in detecting large object instances.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

3.1. TE-FPN

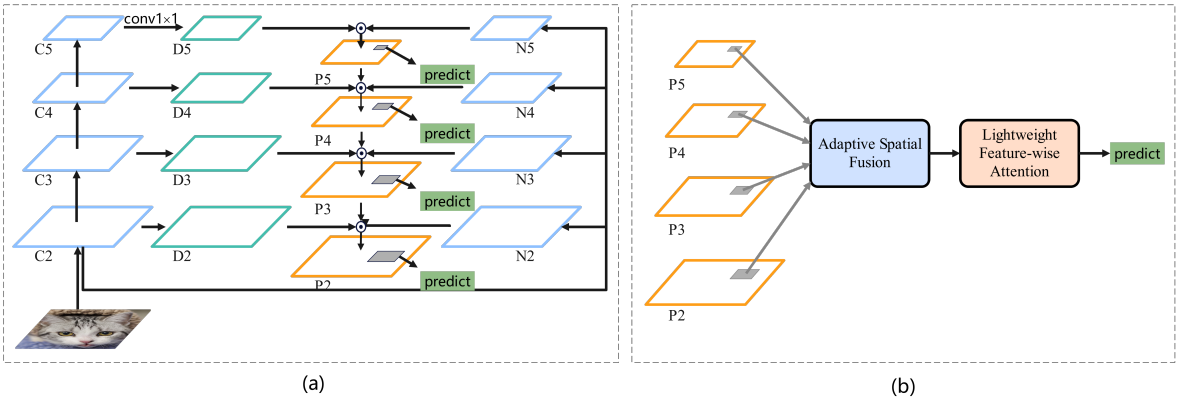

Figure 2. Overview of TE-FPN. The proposed approach introduces an architecture called Enhancing texture information with a bottom-up architecture (ETA), which allows the input of low-level features to each level of the feature hierarchy. The lightweight Feature-wise Attention (LFA) is employed to extract channel weights using the channel attention module, which are then used to generate the final integrated features. The Feature-wise Attention (FWA) leverages multiple contextual information to acquire channel weights and enhance the features of the final output feature map. (a) Schematic diagram illustrating the architecture of Enhancing texture information with a bottom-up approach. (b) Schematic diagram illustrating the enhanced features using Feature-wise Attention.

The top-down propagation of robust semantic information by FPN enhances the model's ability to accurately classify features at all levels of the pyramid. The accurate localization of instances in the model relies on their high response to instance parts or edges, whereas the bottom-up path approach effectively propagates robust TEXTURE information, thereby enhancing the model's ability to localize features at all levels of the feature pyramid. In this paper, we propose a new pyramid structure Texture-Enhanced FPN (TE-FPN), which contains a bottom-up path leading from the low level of the backbone network, so that the fused feature map has both strong semantic and texture information. Additionally, we introduce a novel channel attention mechanism to the Soft RoI Selection process, aiming to further enhance the fused features.

Enhancing texture information with a bottom-up architecture. FPN acquires features from the backbone, and a large amount of texture information is inevitably lost when the backbone is downsampled, a situation that may affect the accuracy of the detection network in obtaining information about the location of instances in the image. To address this limitation, we incorporate an 'Enhancing texture information with a bottom-up architecture' (ETA) into FPN, aiming to enhance the texture information in the feature map at each level. Following the definition of FPN [10], feature layers of the same size are generated in each network phase, and different feature layers correspond to different phases of the network. The Resnet-50 [33] serves as the backbone network, and $\{P2, P3, P4, P5\}$ represent the feature layers generated by the FPN. $\{C2, C3, C4, C5\}$ represent the feature layers at different stages in the backbone network, and $\{D2, D3, D4, D5\}$ represent the feature layers of $\{C2, C3, C4, C5\}$ after dimensionality reduction using convolution. From C2 to C5, P2 to P5, and D2 to D5 spatial sizes are gradually downsampled with a downsampling factor of 2. $\{N2, N3, N4, N5\}$ represent the feature maps newly generated by the bottom-up path, corresponding to $\{C2, C3, C4, C5\}$.

Specifically, the first step is to reduce N2 to C2 using convolution with a channel dimension of 256. This channel dimension aligns with the feature map in FPN and enables effective fusion between the two feature maps. Subsequently, the downsampled feature map is further downsampled with

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

sampling coefficients of 2, 4, and 8 to generate $\{N3, N4, N5\}$, preserving more texture information compared to conventional convolutional downsampling. The process is described as follows:

$$N_i = pool_{2^{i-2}}(N_2) \tag{1}$$

Where $pool_\gamma$denotes the downsampling of the sampling factor ofγ

Finally, depicted in Figure 3(a), we up-sample Pi using a sampling factor of 2 and merge the up-sampled feature map with Ni+1 and Di+1, which have the same size, to generate Pi+1. Notably, P5 is obtained by merging only D5 and N5. The resulting fused feature map contains a combination of rich semantic and texture information. To mitigate the confounding effect after feature fusion, we employ a Lightweight Feature-wise Attention (LFA) module, as shown in Figure 3(b). In this module, we implement a lightweight attention mechanism using FC layers instead of the more complex shared MLP layers, and combine the output feature vectors through element-wise summation with a sigmoid function. The process can be summarized as follows:

$$\begin{aligned} LFA(F) = \sigma(fc1(Avgpool(x)) + fc2(Maxpool(x)) \\ + fc3(Softpool(x))) \end{aligned} \tag{2}$$

Where,LFA denotes the lightweight channel attention function,σ denotes the sigmoid function, $Avgpool$,$Maxpool$,$Softpool$ denotes the global average pooling, global maximum pooling and global soft pooling. Respectively, lightweight Feature-wise Attention is used to mitigate the confounding effect after feature fusion, rather than enhancing the features themselves.

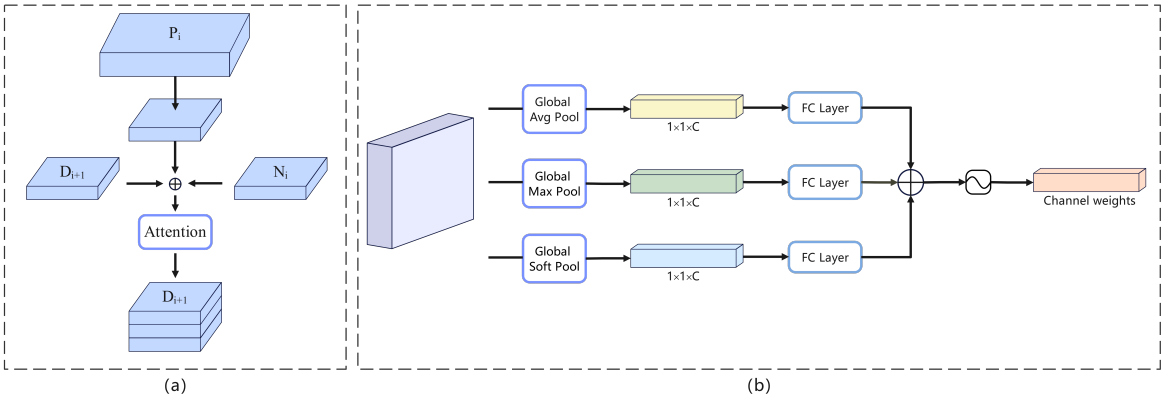

Figure 3. (a) The schematic diagram illustrates our proposed bottom-up architecture for the feature fusion module, enhancing texture information. (b) The diagram illustrates the schematic of the Lightweight Feature-wise Attention.

Feature-wise Attention. During the target detection task, the detector heavily relies on the edge and texture information of the instances in the image to accurately delineate the instances. However, the multi-scale feature fusion can introduce a blending effect, leading to discontinuity in the fused feature map features. This can result in the detector acquiring incorrect edge and texture information of the instances, ultimately affecting the accuracy of instance localization and detection tasks.To mitigate the influence of the blending effect on the model, we introduce a novel attention module called 'Feature-wise Attention' in its lightweight version, specifically designed to address the impact of the blending effect on model performance. In this work, we replace the ROI module with the Soft RoI Selection (SRS) module for feature fusion across different scales in the feature pyramid. Additionally, we incorporate the standard Feature-wise Attention to enhance the features in the final output.

The Feature-wise Attention (FWA) is illustrated in Figure 4. In this mechanism, we first utilize global average pooling, global soft pooling, and global maximum pooling to obtain three different spatial contexts. These contexts capture various aspects of the feature map. Next, each context is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

processed through an MLP layer with shared parameters. Finally, the resulting feature vectors are combined using element-wise summation followed by a sigmoid function. The channel attention in FWA focuses on identifying significant features within the graph. Global average pooling provides feedback for every pixel point on the feature map, while global maximum pooling focuses on gradients by considering only the areas with the highest response. On the other hand, soft pooling [14] produces diverse gradient feedback for different pixel points during gradient backpropagation. Global average pooling tends to capture overall image features, global maximum pooling emphasizes instance edge information, and global soft pooling captures the overall texture information of the instance. By incorporating the global soft pooling contextual information and assigning higher weights to each pixel point of the instance, we enhance the texture information of the instance in the image. The Feature-wise Attention mechanism can be summarized as follows:

$$FWA(F) = \sigma(mlp(Avgpool(x)) + mlp(Maxpool(x)) + mlp(Softpool(x))) \quad (3)$$

where denotes the FWA attention mechanism and σdenotes the sigmoid function. By adding FWA, the texture information of the instances in the image is enhanced to obtain better localization.

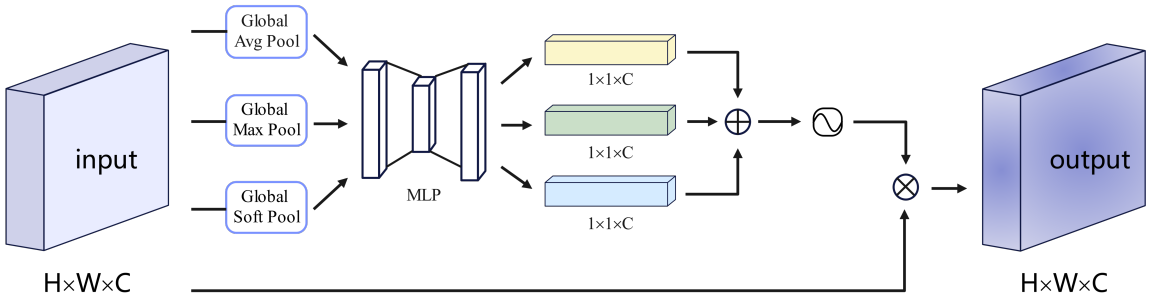

Figure 4. Schematic representation of Feature-wise Attention.

3.2. Instance Based Advanced Guidance Module

The transformer used in our TIG-DETR detector follows the structure of Shifted Window based Self-Attention in SWIN-T. It replaces the multi-headed self-attentive module in DETR with W-MSA and SW-MSA in an alternating manner. The main goal is to reduce the computational complexity of the Transformer part in DETR, enhance the detection performance of small object instances, and speed up the model convergence. However, the limitation of interaction between windows in Shifted Window based Self-Attention affects the detection performance of large object instances. To address this, we introduce a new module called Instance Based Advanced Guidance Module (IAM) before the encoder. This module allows the model to perceive the instances in the image before performing local self-attention, compensating for the degraded detection performance of large object instances caused by the window interaction limitation.

Specifically, as shown in Figure 5, the images from different stages in the backbone undergo a scale-invariant downsampling process to achieve a consistent channel dimension. They are then resized to the final output size. Afterwards, they are fused with the output image at multiple scales, enabling the fused feature map to combine information from different scales and enhance the texture information of the image.After undergoing an LFA, the fused feature map, originally of size $w \times h \times C$, is divided into M^2patches. In Figure 5, we use M=2 as an example. These patches are fused together through concatenation, resulting in a patch of size $(h/M) \times (w/M) \times CM^2$. The different colors within the fused patches represent the channel information of the patches at different locations. Each pixel within the fused patch contains positional information from the patches at different locations, allowing the model to extract global features, contextual relationships, and better perceive objects of varying

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

sizes.The fused patch is then passed through the multi-headed self-attentive module, and the output patch is used to reconstruct the original feature map.We believe that this method is advantageous for improving Shifted Window based Self-Attention, as it enables the feature map to capture instances before window attention. Despite a slight increase in computational complexity compared to Shifted Window based Self-Attention, we have successfully implemented this approach. The details are as follows:

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

$$\Omega(W-MSA) = 4hwC^2 + 2N^2hwC \tag{5}$$

$$\Omega(IAM) = 4hwC^2M^2 + 2(h/M)^2(w/M)^2C \tag{6}$$

In the equation, N represents the edge length of the shifted window. It is observed that the computational complexity of the global MSA increases quadratically with hw. When N remains constant, the computational complexity of W-MSA becomes linear. In our IAM, with M fixed, the computational complexity of $2(h/M)^2(w/M)^2$Cin this part is much smaller than $2(hw)^2C$. But $4hwC^2M^2$ for $4hwC^2$ is only a linear increase, much smaller than the difference between the computation of $2(h/M)^2(w/M)^2C$ and $2(hw)^2C$. Consequently, the computational complexity of IAM is still significantly lower compared to the global MSA.

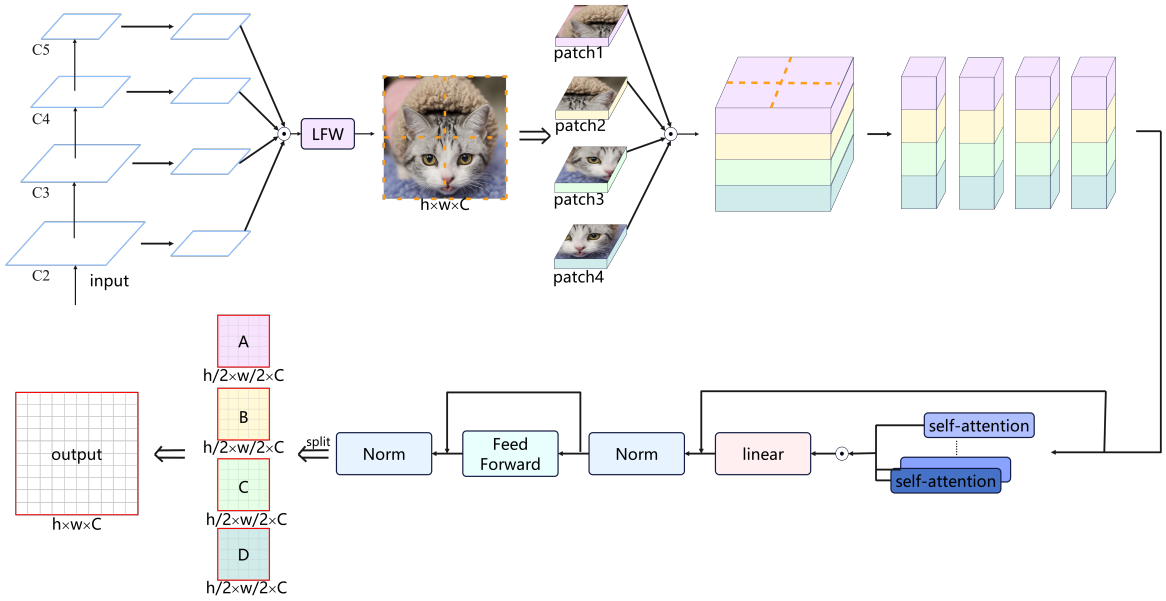

Figure 5. Schematic diagram of Instance Based Advanced Guidance Module.

By performing the mentioned operations, we enhance the texture information of the image and establish associations among individual pixel points within the image prior to applying movable window attention to the entire image. This allows the model to perceive instances in the image before the finer self-attention operation, thereby improving its ability to detect large object instances. A feedforward network (FFN) is employed after the IAM module.

We present the generalized Instance Based Advanced Guidance Module, which can be applied to various backbone networks without the need for FPNs. In the case of a backbone with FPNs, we utilize FPNs to replace the multiscale fusion component of the model.

 doi:10.20944/preprints202306.1084.v1

4. Results

We conduct target detection experiments using TIG-DETR on the COCO dataset. We compare TIG-DETR and its individual components with other techniques. Additionally, we conduct instance segmentation comparison experiments between TE-FPN and other techniques on the CitysPaces dataset.

4.1. COCO and Evaluation Metrics

We compare our method with other techniques on the challenging COCO [18] dataset. The COCO dataset consists of 80 classes and includes 115,000 training images (train2017), 5,000 validation images (val2017), and 20,000 images for test-dev (the labels of test-dev are not publicly available). We train our model on the train-2017 subset and evaluate the ablation study on val2017, as well as report the final results on test-dev. The evaluation metrics used are coco's average precision (AP), including AP, AP_{50},AP_{75}, AP_S, AP_M, AP_L. The latter three metrics measure the performance for targets of different sizes.

4.2. Implementation Details

By default, we train the TIG-DETR model using AdamW [24] optimizer on 8 GPUs for 50 epochs. The initial learning rate is set to 2×10^{-4}, and after the 40th epoch, the learning rate is reduced by a factor of 0.1. For the TE-FPN model, we train it on 8 GPUs for 15 epochs. During training, we extract 16 images from one image to generate training samples. The initial learning rate is set to 0.02, and it is reduced by a factor of 0.1 after the 10th and 14th epochs, respectively.

4.3. Main Results

We evaluated TIG-DETR and its components on the COCO test development set and compared them with advanced two-stage detectors. The final results are presented in Table 1. We compared TIG-DETR with DETR, and when using only ResNet-50 as the backbone without FPN, our model achieved an AP score that was only 0.6 lower than DETR. TIG-DETR, which utilizes a local attention mechanism, significantly improves the model's convergence speed, which is only one-tenth of DETR. By introducing IAM to alleviate the limitations of local attention in detecting large object instances, the model's accuracy in detecting large objects decreases slightly, and after adding TE-FPN, the AP reaches 45.9. This demonstrates that IAM has a significant impact on improving the performance of local attention in detecting large object instances. We further adjusted IAM by removing the multiscale fusion and incorporating TE-FPN, resulting in a final AP of 44.1 for TIG-DETR. We also applied IAM to other DETR detectors based on the local self-attention mechanism [20], as shown in Table 1, demonstrating its effectiveness across different models. Notably, IAM shows remarkable improvements in detecting large object instances, highlighting its robustness and versatility. Visualization results are presented in Figure 6.

By replacing FPN with TE-FPN, we achieved an AP of 38.4 for Faster R-CNN using ResNet-50 as the backbone, which is 2.0 points higher than Faster R-CNN based on ResNet-50-FPN. Moreover, TE-FPN also performs well with more powerful backbone networks. For instance, when using ResNext-101-32x4d, our approach improves the AP by an additional 1.2 points. Table 1 demonstrates the varying degrees of performance improvement achieved by TE-FPN across different backbones, detectors, and tasks, highlighting its robustness and generalization capability. Visualization results are presented in Figure 7.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

Table 1. Comparison with baseline and state-of-the-art COCO test development methods

Method	Backbone	Schedule	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN∗	ResNet-50-FPN	×1	36.4	58.1	39.1	21.3	40.5	44.6
Faster R-CNN∗	ResNet-101-FPN	×1	38.6	60	42.1	22.2	42.5	47.1
Faster R-CNN∗	ResNet-101-FPN	×2	39.4	61.1	43.2	22.6	42.7	50.1
Faster R-CNN∗	ResNext-101-32x4d-FPN	×1	40.3	62.6	43.6	24.5	42.9	49.9
Faster R-CNN∗	ResNext-101-64x4d-FPN	×1	41.7	**64.9**	44.4	24.7	45.8	51.3
Mask R-CNN∗	ResNet-50-FPN	×1	37.1	58.9	40.3	22.3	40.5	45.5
Mask R-CNN∗	ResNet-101-FPN	×1	39.1	61.2	42.2	22.8	42.3	49.2
Mask R-CNN∗	ResNet-101-FPN	×2	40	61.8	43.7	22.7	43.4	52.1
RetinaNet∗	ResNet-50-FPN	×1	35.8	55.7	38.7	19.4	39.7	44.9
RetinaNet∗	MobileNet-v2-FPN	×1	32.9	52.1	34.9	17.9	34.8	42.6
DETR	ResNet-50	×1	42	62.4	44.2	20.5	45.8	**61.1**
Deformable DETR	ResNet-50	×1	43.8	62.6	47.7	26.4	47.1	58
Deformable DETR+IAM	ResNet-50	×1	**44.3**	62.9	**48.3**	**26.3**	**47.6**	60.3
Faster R-CNN*(ours)	ResNet-50-TE-FPN	×1	38.4	61	41.9	23.1	41.7	47.5
Faster R-CNN(ours)	ResNet-101-TE-FPN	×1	40.2	62.6	43.6	23.5	43.5	50.9
Faster R-CNN (ours)	ResNet-101-TE-FPN	×2	41.1	63.4	44.3	23.6	44.1	52.7
Faster R-CNN (ours)	ResNext-101-32x4d-TE-FPN	×1	41.5	63.8	45.1	24.8	45.1	52.3
Faster R-CNN(ours)	ResNext-101-64x4d-TE-FPN	×1	42.7	65.4	46	25.9	45.9	53.5
Mask R-CNN(ours)	ResNet-50-TE-FPN	×1	38.9	61.1	42.4	23.2	42.2	49
Mask R-CNN(ours)	ResNet-101-TE-FPN	×1	40.4	63	44.2	23.7	43.3	51.4
Mask R-CNN(ours)	ResNet-101-TE-FPN	×2	41.5	63.6	45.7	24.1	44.2	53.2
RetinaNet(ours)	ResNet-50-TE-FPN	×1	36.9	57.9	39.6	20.8	40.1	46.4
RetinaNet(ours)	MobileNet-v2-TE-FPN	×1	33.9	53.7	35.8	18.5	35.7	43.9
TIG-DETR	ResNet-50	×1	43.1	62.1	46.2	24.7	46.8	60.5
TIG-DETR	ResNet-50-TE-FPN	×1	**44.1**	**62.8**	**48.4**	**25.6**	**47.9**	**62.4**

*The symbol '∗'means our re-implemented results through mmdetection.The bolded part of the font indicates the largest indicator in the comparison experiment.

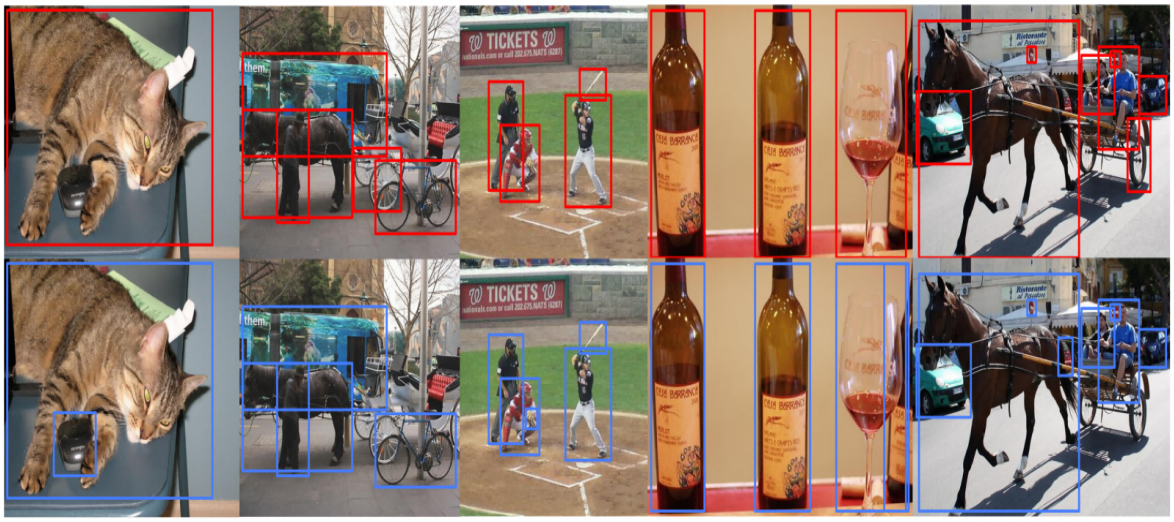

Figure 6. TIG-DETR vs DETR. red bounding box shows the detection result of DETR, while blue bounding box shows the detection result of TIG-DETR.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

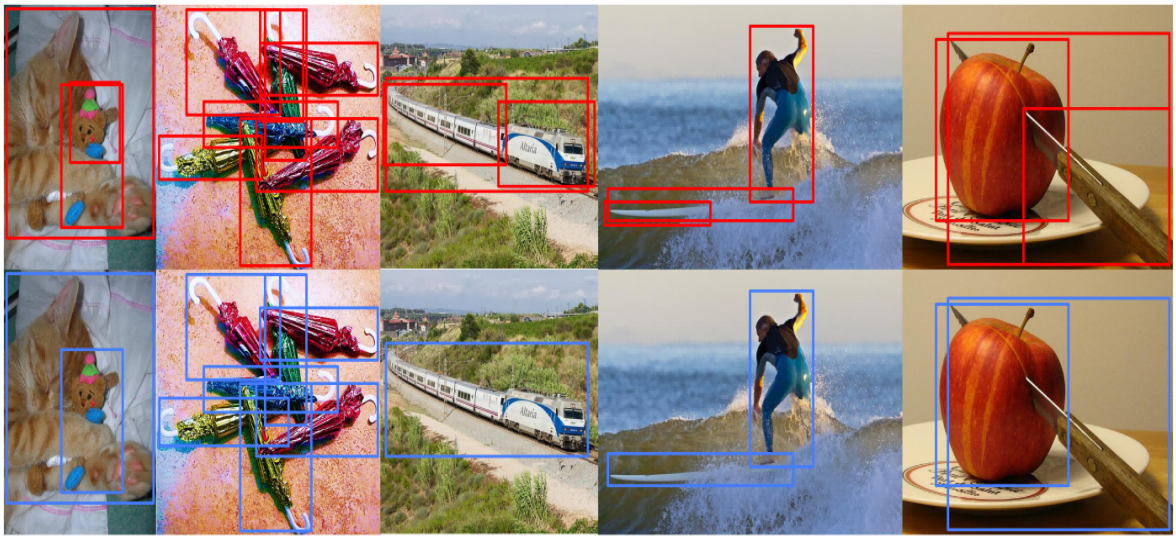

Figure 7. TE-FPN vs FPN. red bounding box shows the detection result of FPN, while blue bounding box shows the detection result of TE-FPN.

4.4. Ablation Study

In this section, we conduct ablation experiments to analyze the impact of each component in our proposed TIG-DETR and TE-FPN modules

4.4.1. TIG-DETR

To analyze the significance of each component in TIG-DETR, we systematically incorporated TE-FPN and IAM into the model to assess the influence of each component on the model's performance. The results of all the experiments are presented in Table 2.

Table 2. TIG-DETR ablation experiments on COCO.

IAM	S-IAM	TE-FPN	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
			40.3	60.5	42.9	22.2	44.5	57.4
√			43.1	62.1	46.2	24.7	46.8	60.5
	√		40.7	60.6	44.1	22.1	44.4	59.1
		√	43.7	62.4	47.6	26.7	47.6	60.7
	√	√	**44.1**	**62.8**	**48.4**	**26.6**	**47.9**	**62.4**

*IAM denotes Instance Based Advanced Guidance Module, S-IAM denotes Instance Based Advanced Guidance Module after removal of multiscale fusion, TE-FPN denotes Texture-Enhanced FPN

According to Table 2, the inclusion of the Instance Based Advanced Guidance Module enhances the model's accuracy by 2.8 APs. Additionally, the introduction of multiscale fusion leads to improved detection performance for instances of various sizes, particularly for large object instances with a notable improvement of 3.1 APs.

The inclusion of the Instance Based Advanced Guidance Module, which excludes multiscale fusion, solely enhances the detection performance of large object instances by 1.7 APs. The overall improvement for the model is 0.4 APs.

The introduction of Texture-Enhanced FPN results in a remarkable improvement of 3.4 APs in the model, highlighting the substantial performance enhancement brought by Texture-Enhanced FPN to TIG-DETR.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 June 2023 doi:10.20944/preprints202306.1084.v1

4.4.2. TE-FPN

In order to analyze the significance of each component in TE-FPN, we incrementally incorporate the bottom-up path, LFA, and FWA into the model to evaluate the effectiveness of each component. The results also demonstrate the synergistic effect of combining different components, highlighting their complementary nature. The baseline model for this ablation study is a Faster R-CNN with Resnet-50 as the backbone. The detailed results are presented in Table 3.

Table 3. TE-FPN ablation experiments on COCO.

SRS	ETA	LFA	SRS+FWA	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
				36.2	56.1	38.6	20.0	39.6	47.5
√				36.8	59.1	39.8	20.7	40.2	48.3
	√			37.0	56.7	39.9	20.8	40.3	48.1
		√		36.8	56.5	39.3	20.6	40.2	48.0
			√	37.5	57.4	40.1	21.5	40.7	49.0
	√	√		37.5	57.5	40.2	21.4	41.0	49.4
	√		√	37.6	58.0	40.1	21.5	41.3	49.6
		√	√	37.8	57.9	40.4	21.6	41.2	49.8
	√	√	√	**38.2**	**58.8**	**40.9**	**21.9**	**42.3**	**50.4**

*We use Resnet-50+FPN+Faster-R-CNN as our baseline method and gradually add Enhancing texture information with a bottom-up architecture (ETA), Lightweight Feature- wise Attention (LFA), Feature-wise Attention (FWA).SRS is an abbreviation for the Soft RoI Selection method mentioned in the paper.

According to Table 3, the incorporation of bottom-up paths into TE-FPN improves the baseline approach by 0.8 AP. This demonstrates the significant impact of enhancing texture information in the feature map on enhancing model performance.

By incorporating LFA into the baseline method, the AP improves from 36.2 to 36.8. This indicates that the lightweight Feature-wise Attention has a significant impact on reducing the confounding effect caused by cross-scale feature fusion.

4.5. Cityscapes

By replacing the ROI of the baseline method with SRS, the AP improves by 0.7. Furthermore, the addition of FWA to the model results in an additional improvement of 0.6 AP. This highlights the significant enhancement effect of FWA on the features.

We conducted additional experiments on TE-FPN using the Cityscapes [51] dataset to assess its effectiveness in performing instance segmentation tasks. The Cityscapes dataset consists of street scenes captured by in-vehicle cameras, containing numerous overlapping and blurred instances. We utilized 2.9K images for training, 0.5K images for validation, and 1.5K images with fine annotations for testing. Additionally, 20K images with coarse annotations were included but excluded from training. We present the results on the validation and secret test subsets, evaluating the model performance based on AP and AP_{50} metrics.

The instance segmentation task focuses on 8 object classes: person, rider, car, truck, bus, train, motorcycle, and bicycle. We trained the model using 8 GPUs, with 8 images randomly sampled from each training image. The initial learning rate was set to 0.01, and it was reduced to 0.001 after 18k iterations. The test performance results are presented in Table 4.

Table 4. The effectiveness of TE-FPN on Cityscapes.

Method	AP[val]	AP	AP_{50}	person	rider	car	truck	bus	train	motorcycle	bicycle
Mask R-CNN [fine-only]	31.5	26.2	49.9	30.5	23.7	46.9	22.8	32.2	18.6	19.1	16.0
Mask R-CNN [COCO]	36.4	32.0	58.1	34.8	27.0	49.1	30.1	40.9	30.9	24.1	18.7
TE-FPN[fine-only]	34.2	29.5	54.8	34.0	27.8	52.7	25.6	35.2	23.0	21.1	19.1
TE-FPN[COCO]	**39.6**	**34.9**	**61.2**	**39.1**	**31.1**	**54.3**	**31.5**	**43.9**	**31.1**	**26.2**	**22.4**

*Results on Cityscapes val subset, denoted as AP [val], and on Cityscapes test subset, denoted as AP..The bolded part of the font indicates the largest indicator in the comparison experiment.

We utilized Mask R-CNN with ResNet-50 as the baseline model and replaced the FPN with TE-FPN. By pre-training our TE-FPN model on COCO, we achieved a performance improvement of 2.7 APs over Mask R-CNN for "fine-only" data. As shown in Table 3, TE-FPN consistently demonstrates notable results for the instance segmentation task, highlighting the model's strong generalization capability and its effectiveness across different tasks.

5. Conclusion

We present our target detection method, TIG-DETR, in this paper. TIG-DETR incorporates several simple yet effective components to enhance the model's neck part, and introduces an Instance Based Advanced Guidance Module to improve the model's ability to perceive large object instances through local attention. By utilizing the low-level feature map from the backbone network for downsampling and fusing it with features at all levels in the FPN, we ensure that the feature map contains both semantic and texture information. In Shifted Window based Self-Attention, we downsample and fuse the feature map generated by each stage in the backbone with the output features. We then slice the fused feature map, perform a concatenation operation on the slices, and pass it through a multi-headed self-attention module. This process enhances the texture information of the feature map and improves the model's ability to perceive large object instances. Our approach has shown promising results. Our future work will focus on extending our approach to video data.

Author Contributions: : Conceptualization, K.W.; methodology, Z.Y.and K.W; software, Z.Y.; validation, Y.W.; formal analysis, C.L.; data curation, K.W.; writing—original draft preparation, Z.Y.and K.W; writing—review and editing, Z.Y.and K.W; visualization, Z.Y.and K.W; supervision, G.L.;

Funding: This research was funded by the Fund of Jilin Provincial Science and Technology Department :Research and development of junior high school physical intelligence experimental platform in mobile environment(20200401087GX).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.
2. Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
3. Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.
4. Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.
5. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

6. Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10012-10022.
7. Liu S, Huang D, Wang Y. Learning spatial fusion for single-shot object detection[J]. arXiv preprint arXiv:1911.09516, 2019.
8. Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
9. Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.
10. Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.
11. Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 8759-8768.
12. Guo C, Fan B, Zhang Q, et al. Augfpn: Improving multi-scale feature learning for object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 12595-12604.
13. Luo Y, Cao X, Zhang J, et al. CE-FPN: enhancing channel information for object detection[J]. Multimedia Tools and Applications, 2022, 81(21): 30685-30704.
14. Stergiou A, Poppe R, Kalliatakis G. Refining activation downsampling with SoftPool[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10357-10366.
15. He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.
16. Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[J]. Advances in neural information processing systems, 2014, 27.
17. Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104: 154-171.
18. Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer International Publishing, 2014: 740-755.
19. Loshchilov I, Hutter F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.
20. Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767 (2018).
21. Zhu X, Su W, Lu L, et al. Deformable detr: Deformable transformers for end-to-end object detection[J]. arXiv preprint arXiv:2010.04159, 2020.
22. Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers[C]//Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. Springer International Publishing, 2020: 213-229.
23. Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer International Publishing, 2014: 740-755.
24. Loshchilov I, Hutter F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.
25. Fu C Y, Liu W, Ranga A, et al. Dssd: Deconvolutional single shot detector[J]. arXiv preprint arXiv:1701.06659, 2017.
26. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.
27. Chen T, Saxena S, Li L, et al. Pix2seq: A language modeling framework for object detection[J]. arXiv preprint arXiv:2109.10852, 2021.
28. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
29. Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.
30. Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.

31. Wang W, Xie E, Li X, et al. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 568-578.
32. Han K, Xiao A, Wu E, et al. Transformer in transformer[J]. Advances in Neural Information Processing Systems, 2021, 34: 15908-15919.
33. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
34. Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.
35. Zheng M, Gao P, Zhang R, et al. End-to-end object detection with adaptive clustering transformer[J]. arXiv preprint arXiv:2011.09315, 2020.
36. Qiu X, Sun T, Xu Y, et al. Pre-trained models for natural language processing: A survey[J]. Science China Technological Sciences, 2020, 63(10): 1872-1897.
37. Lin T, Wang Y, Liu X, et al. A survey of transformers[J]. AI Open, 2022.
38. Mnih, Volodymyr, Nicolas Heess, and Alex Graves. "Recurrent models of visual attention." Advances in neural information processing systems 27 (2014).
39. Guo, Meng-Hao, et al. "Attention mechanisms in computer vision: A survey." Computational Visual Media 8.3 (2022): 331-368.
40. Wang, Wenhai, et al. "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions." Proceedings of the IEEE/CVF international conference on computer vision. 2021.
41. Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88: 303-338.
42. Dai, Zihang, et al. "Funnel-transformer: Filtering out sequential redundancy for efficient language processing." Advances in neural information processing systems 33 (2020): 4271-4282.
43. Roy, Aurko, et al. "Efficient content-based sparse attention with routing transformers." Transactions of the Association for Computational Linguistics 9 (2021): 53-68.
44. Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
45. Gao Z, Xie J, Wang Q, et al. Global second-order pooling convolutional networks[C]//Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. 2019: 3024-3033.
46. Wang X, Girshick R, Gupta A, et al. Non-local neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7794-7803.
47. Mnih V, Heess N, Graves A. Recurrent models of visual attention[J]. Advances in neural information processing systems, 2014, 27.
48. Jaderberg M, Simonyan K, Zisserman A. Spatial transformer networks[J]. Advances in neural information processing systems, 2015, 28.
49. Hearst M A, Dumais S T, Osuna E, et al. Support vector machines[J]. IEEE Intelligent Systems and their applications, 1998, 13(4): 18-28.
50. Wu, Bichen, et al. "Visual transformers: Token-based image representation and processing for computer vision." arXiv preprint arXiv:2006.03677 (2020).
51. Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3213-3223.