Training Object Detectors from Scratch: An Empirical Study in the Era of Vision Transformer

Weixiang Hong, Jiangwei Lao, Wang Ren, Jian Wang, Jingdong Chen, Wei Chu
Ant Group
hwx229374@antgroup.com

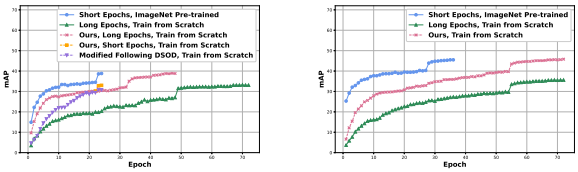
Abstract

Modeling in computer vision has long been dominated by convolutional neural networks (CNNs). Recently, in light of the excellent performances of self-attention mechanism in the language field, transformers tailored for visual data have drawn numerous attention and triumphed CNNs in various vision tasks. These vision transformers heavily rely on large-scale pre-training to achieve competitive accuracy, which not only hinders the freedom of architectural design in downstream tasks like object detection, but also causes learning bias and domain mismatch in the fine-tuning stages. To this end, we aim to get rid of the "pre-train & fine-tune" paradigm of vision transformer and train transformer based object detector from scratch. Some earlier work in the CNNs era have successfully trained CNNs based detectors without pre-training, unfortunately, their findings do not generalize well when the backbone is switched from CNNs to vision transformer.

Instead of proposing a specific vision transformer based detector, in this work, our goal is to reveal the insights of training vision transformer based detectors from scratch. In particular, we expect those insights can help other researchers and practitioners, and inspire more interesting research in other fields, such as semantic segmentation, visual-linguistic pre-training, etc. *One of the key findings is that both architectural changes and more epochs play critical roles in training vision transformer based detectors from scratch. Experiments on MS COCO datasets demonstrate that vision transformer based detectors trained from scratch can also achieve similar performances to their counterparts with ImageNet pre-training.*

1. Introduction

The extraordinary performance of AlexNet [25] on the ImageNet image classification challenge has sparked the passion in convolutional neural networks (CNNs), and led to a variety of powerful CNN backbones through greater scale [17], more extensive connections [24], and more sophisticated forms of convolution [6]. Consequently, modeling in computer vision has long been dominated by CNNs, until the Transformer architecture [8] is recently adapted from natural language processing (NLP) to vision community. A group of transformers tailored for visual data have triumphed numerous CNN-based methods in many vision tasks (*e.g.* , image classification [9], object detection [2], semantic segmentation [5], *etc*). Among them, object detection is one of the fastest moving areas due to its wide applications in surveillance, autonomous driving, *etc*.



(a) Swin-T based FCOS [43]. (b) Swin-T based Faster R-CNN [36].

Figure 1. We train and evaluate Swin-T [30] based detectors (FCOS [43] and Faster R-CNN [36]) on the COCO dataset. We observe that: 1). Swin-T based detectors trained from scratch do not achieve comparable mAP to their ImageNet pre-trained counterpart, even if more epochs of training are conducted following He *et al*. [14]. 2). The results of Swin-T based FCOS will increase if its architecture is modified following DSOD [39], which is originally proposed to boost the proposal-free CNNs based detector when pre-training is unavailable. However, the performance of "Swin-T + FCOS +DSOD" detector trained from scratch is still not as good as the ImageNet pre-trained one. 3). With suitable architectural changes and sufficient training epochs, the proposed vision transformer based detectors without pre-training demonstrate competitive mAP to their ImageNet pre-trained counterparts.

Most of the advanced object detectors require initialization from large-scale pre-training to achieve good performances, no matter whether their backbones are CNNs or vision transformers [30, 36]. Typically, these methods first pre-train the backbone model on ImageNet [38] dataset, then fine-tune the pre-trained weights on the specific object detection task. Fine-tuning from pre-trained models has at

Figure 2. **Qualitative comparisons between naively trained-from-scratch Faster R-CNN, and ours.**

least two advantages. First, it is convenient to reuse various state-of-the-art deep models that are publicly available. Second, fine-tuning can quickly generate the final model and requires much fewer annotated training samples than the classification task. The fine-tuning process can also be viewed as an instance of transfer learning [33].

However, there are also critical limitations when adopting the pre-trained networks in object detection: 1). **Limited structure design space** [39]. The pre-trained models are usually cumbersome (containing a huge number of parameters) for performing well on the ImageNet classification task. Existing object detectors directly adopt the pre-trained networks, resulting in little flexibility to control/adjust the network structures. The requirement of computing resources is also boosted by the complex network structures. 2). **Learning bias** [48]. Both the loss functions and category distributions differ between classification and detection tasks, leading to different searching/optimization spaces. Thus, learning may be biased towards a local minimum for detection tasks. 3). **Domain mismatch** [12]. Though fine-tuning can mitigate the gap of different target category distributions, it is still a severe problem when the source domain (ImageNet) has a huge mismatch with the target domain such as depth images, medical images, *etc*.

Some earlier work have studied on training CNNs based object detection networks from scratch [14, 39]. Specifically, DSOD [39] argues that only proposal-free detectors can be trained from scratch, though proposal-based methods like Faster R-CNN [36] often have superior performances than proposal-free ones. In detail, DSOD [39] augments the original detector by deep supervision, stem block and dense prediction, *etc*., to achieve ideal detection performances. In contrast, He *et al*. [14] points out that no architectural change is required for training from scratch. As long as sufficient training iterations are executed, detectors trained from scratch can converge to similar accuracy to their ImageNet pre-training counterparts.

Given the fact that vision transformers have outperformed CNNs in numerous computer vision tasks, we are motivated to raise the following two questions: 1). Do the findings [14, 39] obtained on CNNs based detectors remain effective, in the era of vision transformer? 2). If not, is it still possible to train vision transformer based object detectors from scratch?

In this work, we experimentally answer the two questions above in Section 3 and Section 4. Specifically, we first show that naively applying the experiences from [14, 39] to vision transformer is not enough. As illustrated in Figure 1, if either architectural changes or more training epochs are solely applied, vision transformer based detectors that are trained from scratch will achieve inferior results compared to their pre-trained counterparts. Then, instead of proposing a specific vision transformer based detector, we aim to reveal the insights of training vision transformer based detector from scratch. In particular, we find that both architectural changes and more epochs are important in train vision transformer based detectors from scratch. Together with several other techniques, we manage to train transformer based detectors from scratch and achieve competitive results to the ImageNet pre-trained counterpart. We expect those insights can help other researchers and practitioners, and inspire more interesting research in other fields, such as semantic segmentation [21], visual-linguistic pre-training [19], *etc*.

Our main findings are summarized as follows:

1. **From RoIPooling to RoIAlign**. We observe that proposal-based and proposal-free detectors exhibit distinct behavior when trained from scratch, that is, proposal-free detectors degrade less than proposal-based ones compared to their pre-trained counterparts. We find out this phenomenon is essentially caused by RoIPooling, *i.e*., it hinders the gradient from being smoothly back-propagated to backbone layers. We address this problem by replacing RoIPool-

ing with RoIAlign, and achieve consistencies between proposal-based and proposal-free detectors.

2. **From T-T-T-T to C-C-T-T**. Recent studies have revealed that large-scale pre-training essentially makes lower attention layers to learn inductive bias and "act like convolutions" [35]. Thus, we replace the first two stages of vision transformers with convolution blocks, namely, from T-T-T-T to C-C-T-T, where T and C stand for transformer and convolution block, respectively. Such a replacement directly introduces the inductive prior of convolution into the backbone model, making it less dependent on ImageNet pre-training.

3. **Gradient Calibration**. In C-C-T-T architecture, we observe that the convolution and self-attention layers exhibit significant differences in terms of the scale of gradient. Since it is better to adjust all of the layers a little rather than to adjust just a few layers a large amount [49], we propose to calibrate the gradients of our model, and achieve better convergence property.

4. **More Training Epochs**. As argued by He *et al*. [14], it is unrealistic and unfair to expect models trained from random initialization to converge as fast as those initialized from ImageNet pre-training. Typical ImageNet pre-training can learn not only semantic information, but also low-level features (*e.g*., edges, textures) that do not need to be re-learned during fine-tuning. Therefore, models trained from scratch must be trained for more epochs than typical fine-tuning schedules.

2. Related Work

2.1. Vision Transformer

Inspired by the recent success of self-attention mechanism [45] in natural language field, there are growing interests in exploiting transformer architecture for vision tasks. The pioneering work ViT [9] directly applies a Transformer architecture on non-overlapping image patches for image classification. It achieves an impressive speed-accuracy trade-off on image classification compared to CNNs. Later work such as [13, 30, 44] have made significant progress in modifying the ViT architecture for better performances. Particularly, Swin Tranformer [30] achieves state-of-the-art results on various tasks, including object detection, semantic segmentation, *etc*. Our analysis on training vision transformer based detector will be based on Swin Transformer.

2.2. Combining Vision Transformer & Convolution

Generally speaking, convolutional layers tend to have faster converging rate thanks to their strong prior of inductive bias, while attention layers exhibit higher model capacity that can benefit from large-scale pre-training [35]. To achieve the balance of inductive prior and model capacity, some pioneer work have attempted to combine convolutional and attention layers. For example, Conformer [34] proposes a feature coupling unit to fuse the features extracted by convolutional and self-attention layers, ConViT [7] introduces gated positional self-attention to equip vision transformer with a "soft" convolutional inductive bias. CvT [46] designs a hierarchy of transformers containing a convolutional token embedding, and a convolutional self-attention block leveraging a convolutional projection.

2.3. Train Object Detection from Scratch

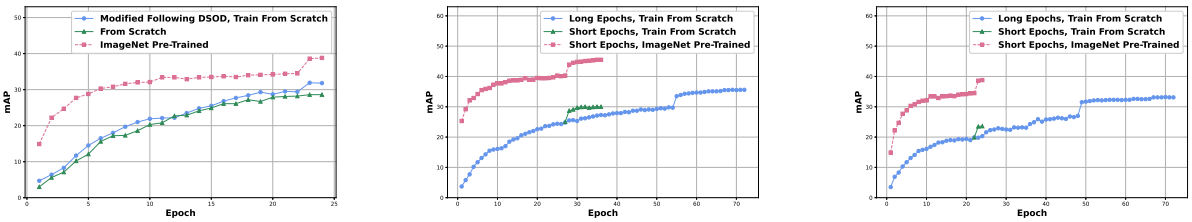
Earlier object detection methods were trained with no pre-training [32, 37, 41]. Given the success of pre-training in the R-CNN [11], the "pre-training and fine-tuning" paradigm becomes a conventional wisdom in modern CNNs based detectors. Nevertheless, due to the limitations caused by pre-training, research efforts have been continuously devoted to train CNNs based detector from scratch [14, 26, 27, 39]. Specifically, DetNet [27] and CornerNet [26] concentrate on designing detection-specific architectures, which is not the focus of this work. DSOD [39] contributes a set of principles for enabling detectors to train from scratch, but it only works for proposal-free methods. He *et al*. [14] does not require any specific architectural change, instead, they advocate that training from scratch only requires more iterations to sufficiently converge.

3. Do the findings obtained on CNNs based detectors remain effective?

In this section, we experimentally investigate whether DSOD [39] and He *et al*. [14] generalize well to vision transformer based detectors.

Backbone. Without loss of generality, we choose the representative work Swin Transformer [30] to investigate the generality of [14, 39] to vision transformer. To be specific, we use Swin-T, an instance of Swin Transformer, as the backbone for all detectors in this section. The complexity of Swin-T is similar to that of ResNet-50 [30].

Detectors. Modern object detectors can be roughly classified into two categories: proposal-based and proposal-free, depending on whether object proposals are utilized as intermediate results. We choose Faster R-CNN [36] and FCOS [43] as representative of proposal-based and proposal-free detectors. Faster R-CNN [36] is the seminal work that innovatively addresses object detection in an end-to-end manner. It first generates a set of region proposals based on pre-defined anchors, then classifies and refines those proposals to obtain final bounding boxes. Thus, Faster R-CNN is also regarded as a two-stage detector. In contrast, FCOS [43] is a one-stage proposal-free method, which contributes a significantly simplified detection framework. The



(a) Train FCOS [43] following DSOD [39]. (b) Train Faster R-CNN following [14]. (c) Train FCOS following [14].

Figure 3. **Train Swin-T based detectors on COCO dataset following [14, 39].** In Figure 3a, the orange, blue and green curves stand for vanilla architecture without pre-training, vanilla architecture with pre-training and modified architecture following DSOD [39]. The modifications do improve the detection performances when trained from scratch, but the gap to the pre-trained baseline is still significant. In Figure 3b and 3c, we train Faster R-CNN [36] and FCOS [43] for more epochs following He *et al*. [14]. The extended training epochs significantly boost the detector trained from scratch, unluckily, the final detection mAP is still inferior to the ImageNet pre-trained counterpart.

bounding boxes are directly regressed from the feature map, without involving anchors and proposals.

Dataset. All experiments are conducted on the challenging MS COCO [28] dataset that includes 80 object classes. Following the common practice [29], all $115K$ images in the trainval $35k$ split are used for training, and all $5K$ images in the minival split are used as validation for analysis study.

Training and Inference. During training, we resize the input images to keep their shorter side being 800 and their longer side less or equal to $1,333$. The whole network is initialized with He initialization [16] and trained using the AdamW [31] optimizer with batch size as 16. During the inference phase, we resize the input image in the same way as in the training phase, and forward it through the whole network to output the predicted bounding boxes with predicted classes. Then, the Non-Maximum Suppression (NMS) [11] is applied with the IoU threshold 0.6 per class to generate final top 100 confident detections per image.

3.1. Train FCOS Following [39]

DSOD [39] advocates 4 principles for training detectors from scratch, *i.e*., 1). Proposal-free; 2). Deep supervision; 3). Stem block; 4). Dense prediction. According to these principles, we made the following modifications to our Swin-T + FCOS detector: 1). FCOS is naturally proposal-free; 2). We add dense connections between stages of Swin-T following [39]; 3). We change the patchify stem to Inception [42] style. Note that [47] has also emphasized the importance of stem block in vision transformer; 4). For each scale, we only learn half of new feature maps and reuse the remaining half of the previous ones. Besides, we also train the vanilla Swin-T + FCOS, with and without initialization from ImageNet pre-training, so as to provide comparison baselines. All three models are trained for 24 epochs, with the learning rate decay once at the 22nd epoch following [3].

The experimental results are shown in Figure 3a. The blue and orange curves denote the vanilla Swin-T + FCOS, with/without ImageNet pre-training. As expected, the one with pre-training significantly outperforms the counterpart that is trained from scratch, in terms of both convergence rate and final detection mAP. Also, as shown by the green curve in Figure 3a, the variant modified according to DSOD [39] demonstrated improved performances than the vanilla Swin-T + FCOS architecture. Unfortunately, it still has a large gap to the pre-trained version.

3.2. Train FCOS and Faster-RCNN Following [14]

Different from DSOD [39], He *et al*. [14] argues that training from scratch on target dataset is feasible without architectural changes, and the resulting detection performance is no worse than its ImageNet pre-training counterparts. Since there is no constraint on proposal-based or proposal-free detectors in [14], we extend the training iterations of both FCOS and Faster R-CNN, with Swin-T as their backbones.

Specifically, we train both two detectors with the initial learning rate and monitor the validation set mAP at each epoch. When the mAP reaches saturation, we decay the learning rate and continue to train it until convergence. The experimental results are shown in Figure 3b and 3c. The extended training epochs significantly boost the detector trained from scratch, unluckily, the final detection mAP is still inferior to the ImageNet pre-trained counterpart. Also, one can compare the gaps of final mAP between ImageNet pre-trained version and train-from-scratch one, and observe that Faster R-CNN degrades more than FCOS.

3.3. Discussion

The results in Figure 3 indicate that the findings in CNNs era, either architectural changes [39] or long epochs [14], do not generalize well enough on vision transformer based detectors. However, given the improvement by solely applying [39] or [14], it is natural to consider combining the

		mAP	mAP_S	mAP_M	mAP_L
Train from Scratch	RoIPooling	26.6	13.0	29.0	37.7
	RoIAlign	30.3	15.5	33.0	41.6
Pre-train & Fine-tune	RoIPooling	42.1	21.3	40.2	49.2
	RoIAlign	42.5	21.6	40.6	49.7

Table 1. **From RoIPooling to RoIAlign.** RoIAlign enables smooth gradient back-propagation and boosts detection mAP by 3.7 points in "Train from Scratch" setting. When it moves to the "Pre-train & Fine-tune" case where the weights are properly initialized, the improvement of RoIAlign is not so significant.

best of two worlds, as is elaborated in the next section.

4. Method

In this section, we present the step-by-step modifications to FCOS and Faster R-CNN, so as to train both proposal-based and proposal-free detectors from scratch.

4.1. From RoIPooling to RoIAlign

We first investigate the distinct behaviors of proposal-free and proposal-based detectors observed in Section 3.2, *i.e.*, Faster R-CNN degrades more than FCOS when switched from "Pre-train & Fine-tune" to "Train from Scratch". We find that the unsatisfactory performances of Faster R-CNN [36] are essentially caused by the internal information loss in RoIPooling [10]. Specifically, RoIPooling involves max pooling on a region of feature maps. It requires to execute quantization or padding, if the coordinates of RoI are float, or the size of region cannot be exactly divided by the size of RoIPooling operator. The quantization or padding inevitably causes information distortion [15], hence hinders the gradients from being smoothly back-propagated from region-level to backbone. The proposal-based methods work well with pre-trained network models because they are well initialized by pre-trained weights, while this is not true for training from scratch.

We empirically find that Faster R-CNN [36] can also converge well if we replace RoIPooing [10] with RoIAlign [15], in which any quantizations of the RoI boundaries or bins are avoided. Instead, bilinear interpolation is exploited to compute the exact values of the output features. We train Swin-T based Faster R-CNN on the COCO dataset, and show the experimental results in Table 1. In the case of "Train from Scratch", RoIAlign achieves 3.7 points higher mAP than RoIPooling. While in the "Pre-Train & Fine-tune" setting, the improvement is relatively tiny, which validates our interpretations above.

4.2. From T-T-T-T to C-C-T-T

The convolution operations inherently have the inductive bias towards local processing, which is replaced in vision transformers by global processing performed by multi-head self-attention [45]. Intuitively, it seems not so necessary to conduct long-range attention modeling in pixel-level or lower stages of backbones. Recent studies have also revealed that large-scale pre-training essentially makes lower attention layers to learn inductive bias and "act like convolutions" [35]. Therefore, a natural idea is to replace early self-attention layers with convolution, so as to directly introduce the inductive prior of convolution into the model and mitigate the dependence on large-scale pre-training.

		mAP	#param.	FLOPs	Memory
Faster R-CNN	T2-T2-T6-T2	30.3	68.93M	246.3G	15.1G
	C2-C2-T6-T2	26.6	44.08M	188.31G	10.6G
	C2-C2-T12-T4	37.9	72.63M	250.68G	15.8G
FCOS	T2-T2-T6-T2	23.6	35.73M	211.56G	14.2G
	C2-C2-T6-T2	18.8	24.67M	187.16G	9.8G
	C2-C2-T12-T4	29.5	39.51M	227.15G	14.4G

Table 2. **From T-T-T-T to C-C-T-T.** All experiments are trained from scratch in this table, the results of "Pre-Train & Fine-tune" setting are presented in ablation studies (Table 3). The C-C-T-T architecture significantly boosts the mAP of both Faster R-CNN and FCOS without consuming more resources.

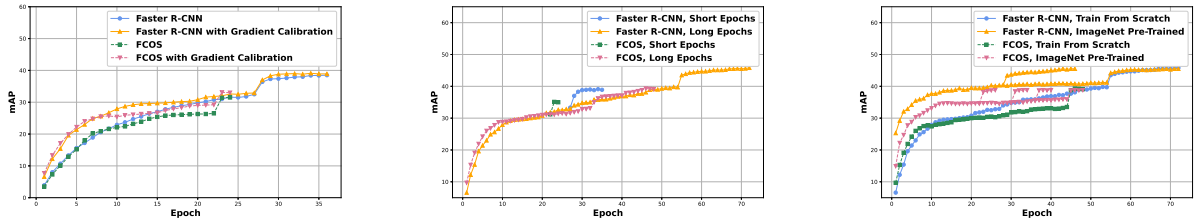
Similar to ResNet [17], Swin Transformer also has four stages, each of which consists of multiple stacked transformer blocks. We dub such an architecture as T-T-T-T, where T stands for transformer. We replace the first two T block with residual convolutional blocks (termed as C) to introduce the inductive prior of convolution into the model. As shown in Table 2, though such replacement is efficient in resource (*e.g.*, parameters, FLOPs and memories), the resulting detection mAPs of both Faster R-CNN and FCOS degrade, possibly due to the reduced model capacity.

Fortunately, thanks to the removal of high-resolution self-attention operators in lower layers, we are feasible to enhance the model capacity by heuristically stacking more self-attention blocks to the latter two T blocks. As shown in Table 2, the C-C-T-T architecture significantly boosts the detection mAP of both Faster R-CNN and FCOS when trained from scratch, without consuming more resources than the vanilla T-T-T-T architecture [1]. More variants of architectures such as C-C-C-C and C-T-T-T are ablated in Section 5.1.1.

4.3. Gradient Calibration

The heterogeneous C-C-T-T architecture introduces the hybrid of convolution and self-attention layer. We observe that they exhibit significant differences in terms of the norm of layer gradient (defined below). The norm of layer gra-

[1] Strictly speaking, the C-C-T-T based detector cannot be called a vision transformer based detector. However, for the simplicity of presentation, we do not explicitly distinguish C-C-T-T and T-T-T-T architectures in concept, and still refer the process of training both of them as training vision transformer based detectors.



(a) Gradient Calibration. (b) More epochs. (c) More epochs with pre-training.

Figure 4. (a). Gradient calibration not only improves the final detection mAP, but also accelerates the convergence rate. (b). Detectors trained from scratch require more epochs than those with pre-trained weights to reach convergence. (c). Under the long epochs training schedules, C-C-T-T architecture trained from scratch converges to a solution that is no worse than the pre-trained T-T-T-T counterpart.

dient of self-attention layers can be up to *10* times as that of convolution layers. Existing research has found that it is better to adjust all of the layers a little rather than to adjust just a few layers a large amount [49]. Therefore, we propose to calibrate the gradients of our model, so as to achieve better convergence property.

Definition 1 (Norm of gradient.)**.** Given a N-layer neural network, we define $C_{i,j}$ to be the expected norm of the gradient w.r.t. weights $W_i(j)$ of layer i:

$$C_{i,j} = \mathbb{E}_{z_0 \in D}\left[\left(z_{i-1}(j) y_i(j)\right)^2\right], \tag{1}$$

where D is the set of training data, z_{i-1} is the activation of layer $i-1$, and y_i is the backpropagated error of layer i.

Definition 2 (Norm of layer gradient.)**.** Given norm of gradient, the norm of layer gradient is defined as:

$$C_i = \mathbb{E}_j\left[C_{i,j}\right]. \tag{2}$$

The proposed gradient calibration works by adjusting the scale of weights in each layer in initialization, so that they are all equal to their geometry average. Specifically, we first compose a batch with randomly selected samples from the training set. Next, we forward and backward this batch through our model to obtain the norm of layer gradient. Then, we compute the geometry average of all norms of layer gradient, and find out the scale correction multiplier of each layer. Finally, we multiply the weights with the scale correction multiplier so that they have the same norm of layer gradient. The entire process is summarized in Algorithm 1, where α in Line 4 is a hyper-parameter (0.25 in this work) against oscillatory behavior. As illustrated in Figure 4a, the training curves of Faster R-CNN and FCOS, with and without gradient calibration. The proposed gradient calibration not only improves the final detection mAP, but also accelerates the convergence rate.

Here we also present another perspective to intuitively interpret the benefits of gradient calibration. Typically, Transformer models require a small learning rate to converge, for example, 0.0005 in BERT [8], 0.001 in Swin Transformer [30]. In contrast, the learning rate for CNNs is much larger, *i.e.*, 0.1 for ResNet [17]. Though the optimizers for vision transformer and CNNs are usually different (*e.g.*, AdamW *vs.* SGD), the significant gap in learning rate suggests that it might be sub-optimal to naively train a hybrid model of convolution and self-attention without any adjustment.

Algorithm 1 Gradient calibration.

1. Draw a batch of samples from training set
2. Compute the norm of layer gradient $C_i = \mathbb{E}_j\left[C_{i,j}\right]$
3. Compute the average ratio $\bar{C} = \left(\prod_i C_i\right)^{\frac{1}{N}}$
4. Compute scale calibration multiplier $r_k = (C_k/\bar{C})^{\alpha}$
5. Calibrate the weights of each layer as $W_k \leftarrow r_k W_k$

4.4. More Training Epochs

Though gradient calibration accelerates the convergence and improves final mAP, it is still unrealistic and unfair to expect models trained from random initialization like [16] to converge as well as those initialized from ImageNet pre-training. Typical ImageNet pre-training can learn not only semantic information, but also low-level features (*e.g.*, edges, textures) that do not need to be re-learned during fine-tuning.

Similar to the settings in Section 3.2, we train our detectors with the initial learning rate and monitor the validation set mAP at each epoch. When the mAP reaches saturation, we decay the learning rate and continue to train it until convergence. In consideration of the scale of the COCO and ImageNet dataset, the iterations of "more training epochs" setting are still much less than the "pre-train & fine-tune" pipeline (See Figure 2 of [14]). The experimental results are shown in Figure 4b. As expected, detectors trained from scratch require more epochs than those with pre-trained weights to reach convergence. Particularly, the final mAP of both Faster R-CNN and FCOS are 45.8 and 38.9, which is superior or similar to their ImageNet pre-trained counterpart, *i.e.*, 42.5 and 38.8 as shown in Table 1 and Figure 3a, respectively. The qualitative comparisons of

several samples are shown in Figure 2.

Moreover, we train T-T-T-T models initialized by ImageNet pre-trained weights for long epochs, and explore different training schedules by varying the epochs at which the learning rate is reduced (where the mAP leaps). As illustrated in Figure 4c, the C-C-T-T model trained from random initialization needs more iterations to converge, but the final mAP is no worse than of the fine-tuning counterpart.

5. Experiments

We conduct experiments on the MS COCO datasets and measure detection performances by mean Average Precision (mAP). All codes and trained models will be made public in future.

5.1. Ablation Studies

We first investigate the effectiveness of different architecture designs and different scale of backbone model. All ablation studies are based on the Faster R-CNN detector [36]. The performance achieved by different variants and backbones settings are reported in the following.

5.1.1 On the design of architecture

Based on the findings that lower attention layers to learn inductive bias and "act like convolutions" [35], we propose 4 variants with increasingly more Transformer stages, *i.e.*, C-C-C-C, C-C-C-T, C-C-T-T and C-T-T-T, where C and T represent Convolution and Transformer respectively. For the purpose of conducting fair comparisons of the 4 designs, we will heuristically adjust the number of layers in each stage, to make each of them consumes similar GPU memory to that of T2-T2-T6-T2 (roughly 16G in this ablation study).

To systematically study the design choices, we evaluate their performances in two different settings, *i.e.*, "Pre-train & Fine-tune", "Train from Scratch". For "pre-train & fine-tune", we pre-train the model on ImageNet dataset, and fine-tune the weights on the COCO dataset for object detection, following the setting of Swin Transformer [30]; For "train from scratch", we conducted the training following our proposed methods in Section 4.

The experimental results are shown in Table 3. On one hand, under the "Pre-train & Fine-tune" paradigm, the full transformer architecture T-T-T-T, which is exactly Swin-T [30], achieves the highest mAP at 45.4. Also, we can observe that the performance monotonically grows during the change from C-C-C-C to T-T-T-T, even if the total number of layers is decreasing. Such a phenomenon demonstrates the great modeling capacity of the self-attention operator. On the other hand, when it moves to "Train from Scratch" setting, C-C-T-T architecture shows the best detection performance at 45.8, which reveals the good trade-off between model capacity and inductive prior. Notably, under the same consumption of memory, the C-C-T-T architecture trained from scratch achieves 0.3 point higher mAP than the T-T-T-T variant initialized from ImageNet pre-trained weights. Several qualitative results are shown in Figure 5.

	Pre-train & Fine-tune	Train from Scratch
C2-C2-C16-C6	43.5	42.3
C2-C2-C14-T5	44.1	43.8
C2-C2-T12-T4	45.3	**45.8**
C2-T2-T8-T3	45.4	44.8
T2-T2-T6-T2	**45.5**	43.4

Table 3. **Different designs of backbone.** The experimental results validate the rationale of our choice of C-C-T-T architecture.

	PT	mAP	mAP_S	mAP_M	mAP_L
T2-T2-T6-T2 (Swin-T)	✓	45.5	30.0	49.0	58.7
C2-C2-T12-T4		45.8	30.5	49.2	59.3
T2-T2-T18-T2, #Channel=96 (Swin-S)	✓	48.2	32.9	52.2	62.2
C2-C2-T36-T4 (#Channel=96)		48.6	33.4	52.9	62.8
T2-T2-T18-T2, #Channel=128 (Swin-B)	✓	51.0	35.5	54.8	64.4
C2-C2-T36-T4 (#Channel=128)		51.2	35.8	55.1	64.8
T2-T2-T18-T2, #Channel=192 (Swin-L)	✓	52.9	37.0	56.7	66.5
C2-C2-T36-T4 (#Channel=192)		53.0	37.2	56.9	66.8

Table 4. **Different scales of backbones.** The proposed method works well for various instances of Swin Transformer [30]. PT stands for Pre-Training, the detectors initialized with ImageNet pre-training are labeled by checkmarks.

5.1.2 On the variants of Swin Transformer

We study the generability of the proposed method to other Swin Transformer variants, namely, Swin-T, Swin-S, Swin-B and Swin-L. Similar to previous settings, we adjust the number of layers in the latter two stages, to make T-T-T-T and C-C-T-T architectures consume similar resources.

The experimental results are shown in Table 4. The #Channels denotes the channel number of the hidden layers in the first stage for T-T-T-T architecture, and the channel number of the residual unit in the first stage for C-C-T-T architecture. The proposed method, trained from scratch, consistently performs favorably against the vanilla Swin Transformer counterpart that is initialized with ImageNet pre-training, validating the efficacy of our work.

5.2. Working with State-of-the-Art Detectors

In this section, we train several state-of-the-art detectors from scratch, including Cascade Mask R-CNN [1], ATSS [55], RepPointsV2 [4] and Sparse R-CNN [40]. We adopt the implementation provided by mmdetection [3]. There are 3 types of backbones for each of the 4 detectors, *i.e.*, 1). The C-C-C-C architecture, which is essentially ResNet-50 [17]; 2). The T-T-T-T architecture, which is es-

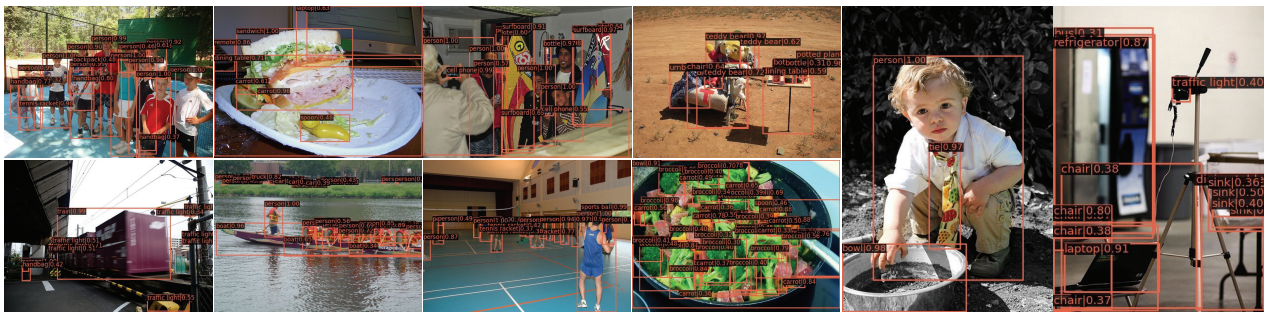


Figure 5. **Results obtained by our vision transformer based detector, trained from scratch on COCO dataset.**

Method	Backbone	PT	mAP	mAP_{50}	mAP_{75}	#params	FLOPs	FPS
Cascade Mask R-CNN	C3-C4-C6-C3 (ResNet-50)	✓	46.3	64.3	50.5	82M	739G	18.0
	T2-T2-T6-T2 (Swin-T)	✓	50.5	69.3	54.9	86M	745G	15.3
	C2-C2-T12-T4		51.0	69.8	55.3	90M	725G	19.2
ATSS	C3-C4-C6-C3 (ResNet-50)	✓	43.5	61.9	47.0	32M	205G	28.3
	T2-T2-T6-T2 (Swin-T)	✓	47.2	66.5	51.3	36M	215G	22.3
	C2-C2-T12-T4		47.5	66.7	51.6	37M	217G	26.1
RepPointsV2	C3-C4-C6-C3 (ResNet-50)	✓	46.5	64.6	50.3	42M	274G	13.6
	T2-T2-T6-T2 (Swin-T)	✓	50.0	68.5	54.2	45M	283G	12.0
	C2-C2-T12-T4		50.4	68.9	54.5	48M	279G	14.6
Sparse R-CNN	C3-C4-C6-C3 (ResNet-50)	✓	44.5	63.4	48.2	106M	166G	21.0
	T2-T2-T6-T2 (Swin-T)	✓	47.9	67.3	52.3	110M	172G	18.4
	C2-C2-T12-T4		48.2	67.4	52.6	113M	170G	22.3

Table 5. **Working with sota detectors.** The proposed methods demonstrate promising results than both C-C-C-C and T-T-T-T architectures. PT stands for Pre-Training, the detectors initialized with ImageNet pre-training are labeled by checkmarks.

sentially Swin-T [30]; 3). The proposed C-C-T-T architecture with gradient calibration. Note that the C-C-C-C and T-T-T-T models are pre-trained on ImageNet, while ours is randomly initialized. All combinations are trained on the COCO dataset with multi-scale learning (resizing the input such that the shorter side is between 480 and 800 while the longer side is at most 1333), AdamW [31] optimizer (initial learning rate of 0.0001, weight decay of 0.05, and batch size of 16) and sufficiently long training epochs. The results are shown in Table 5, the proposed C-C-T-T design with gradient calibration demonstrates competitive performances in all experiments.

6. Conclusion

The domination of convolutional neural networks (CNNs) in various vision tasks has recently been challenged by transformer models, which heavily depend on large-scale pre-training to achieve competitive accuracy. The dependence on pre-training not only hinders the freedom of architectural design in downstream tasks like object detection, but also causes learning bias and domain mismatch in the fine-tuning stages. In this work, we first show that naively applying the experiences from training CNNs based detectors to vision transformer based ones results in unsatisfactory performances. Then, we demonstrate the feasibility of training vision transformer based detectors from scratch, and contribute a set of principles for realizing this goal. Particularly, the purpose of this work is not to propose a specific vision transformer based detector. Instead, we aim to reveal the insights of training vision transformer based detector from scratch, and expect those insights can help other researchers and practitioners, and inspire more interesting research in other fields, such as semantic segmentation [18], image search [20, 22, 23, 50–54], *etc*. By introducing a series of effective modifications such as C-C-T-T and gradient calibration, the proposed detectors demonstrate competitive mAP to their pre-trained variants, under the same long training epochs schedule. Extensive experiments demonstrate the merits and advantages of the proposed method.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 7

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *The European Conference on Computer Vision (ECCV)*, 2020. 1

[3] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Mmdetection: Open mmlab detection toolbox and benchmark, 2019. 4, 7

[4] Yihong Chen, Zheng Zhang, Yue Cao, Liwei Wang, Stephen Lin, and Han Hu. Reppoints v2: Verification meets regression for object detection. In *Advances in Neural Information Processing Systems (NIPS)*, 2020. 7

[5] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Advances in Neural Information Processing Systems (NIPS)*, 2021. 1

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[7] Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning (ICML)*, 2021. 3

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT*, 2019. 1, 6

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 1, 3

[10] Ross Girshick. Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. 5

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3, 4

[12] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[13] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. In *Advances in Neural Information Processing Systems (NIPS)*, 2021. 3

[14] Kaiming He, Ross Girshick, and Piotr Dollar. Rethinking imagenet pre-training. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 3, 4, 6

[15] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 5

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. 4, 6

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 5, 6, 7

[18] Weixiang Hong, Qingpei Guo, Wei Zhang, Jingdong Chen, and Wei Chu. Lpsnet: A lightweight solution for fast panoptic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 8

[19] Weixiang Hong, Kaixiang Ji, Jiajia Liu, Jian Wang, Jingdong Chen, and Wei Chu. Gilbert: Generative vision-language pre-training for image-text retrieval. In *ACM Special Interest Group on Information Retrieval (SIGIR)*, 2021. 2

[20] Weixiang Hong, Xueyan Tang, Jingjing Meng, and Junsong Yuan. Asymmetric mapping quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2019. 8

[21] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan. Conditional generative adversarial network for structured domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[22] Weixiang Hong and Junsong Yuan. Fried binary embedding: From high-dimensional visual features to high-dimensional binary codes. *IEEE Transactions on Image Processing (T-IP)*, 2018. 8

[23] Weixiang Hong, Junsong Yuan, and Sreyasee Das Bhattacharjee. Fried binary embedding for high-dimensional visual features. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8

[24] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[25] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 1

[26] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *The European Conference on Computer Vision (ECCV)*, 2018. 3

[27] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Detnet: A backbone network for object detection. In *The European Conference on Computer Vision (ECCV)*, 2018. 3

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *The European Conference on Computer Vision (ECCV)*, 2014. 4

[29] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *The European Conference on Computer Vision (ECCV)*, 2016. 4

[30] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *The IEEE International Conference on Computer Vision (ICCV)*, 2021. 1, 3, 6, 7, 8

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. 4, 8

[32] Ofer Matan, Christopher J. C. Burges, Yann LeCun, and John Denker. Multi-digit recognition using a space displacement neural network. In *Advances in Neural Information Processing Systems (NIPS)*, 1992. 3

[33] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE TKDE*, 2010. 2

[34] Zhiliang Peng, Wei Huang, Shanzhi Gu, Lingxi Xie, Yaowei Wang, Jianbin Jiao, and Qixiang Ye. Conformer: Local features coupling global representations for visual recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[35] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks?, 2021. 3, 5, 7

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 1, 2, 3, 4, 5, 7

[37] Henry Rowley, Shumeet Baluja, and Takeo Kanade. Human face detection in visual scenes. In *Advances in Neural Information Processing Systems (NIPS)*, 1996. 3

[38] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 2015. 1

[39] Zhiqiang Shen, Zhuang Liu, Jianguo Li, Yu-Gang Jiang, Yurong Chen, and Xiangyang Xue. DSOD: Learning deeply supervised object detectors from scratch. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 3, 4

[40] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, and Ping Luo. Sparse r-cnn: End-to-end object detection with learnable proposals. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 7

[41] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 3

[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[43] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 3, 4

[44] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning (ICML)*, 2021. 3

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 3, 5

[46] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *The IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[47] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, and Ross Girshick. Early convolutions help transformers see better. In *Advances in Neural Information Processing Systems (NIPS)*, 2021. 4

[48] Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. Raise a child in large language model: Towards effective and generalizable fine-tuning. In *EMNLP*, 2021. 2

[49] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 3, 6

[50] Tan Yu, Jingjing Meng, Chen Fang, Hailin Jin, and Junsong Yuan. Product quantization network for fast visual search. *International Journal of Computer Vision (IJCV)*, 2020. 8

[51] Tan Yu, Jingjing Meng, and Junsong Yuan. Is my object in this video? reconstruction-based object search in videos. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, 2017. 8

[52] Tan Yu, Zhenzhen Wang, and Junsong Yuan. Compressive quantization for fast object instance search in videos. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 8

[53] Tan Yu, Yuwei Wu, and Junsong Yuan. Hope: Hierarchical object prototype encoding for efficient object instance search in videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8

[54] Tan Yu, Yi Yang, Yi Li, Lin Liu, Hongliang Fei, and Ping Li. Heterogeneous attention network for effective and efficient cross-modal retrieval. In *ACM Special Interest Group on Information Retrieval (SIGIR)*, 2021. 8

[55] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 7