AttentionShift: Iteratively Estimated Part-based Attention Map for Pointly Supervised Instance Segmentation

Mingxiang Liao[1*] Zonghao Guo[1 *] Yuze Wang[2] Peng Yuan[2] Bailan Feng[2]
Fang Wan[1†]
[1]University of Chinese Academy of Sciences, [2]Huawei Noah's Ark Lab
{liaomingxiang20, guozonghao19}@mails.ucas.ac.cn, wanfang@ucas.ac.cn
wangyuze1@hisilicon.com, {fengbailan, yuanpeng12}@huawei.com

Abstract

Pointly supervised instance segmentation (PSIS) learns to segment objects using a single point within the object extent as supervision. Challenged by the non-negligible semantic variance between object parts, however, the single supervision point causes semantic bias and false segmentation. In this study, we propose an AttentionShift method, to solve the semantic bias issue by iteratively decomposing the instance attention map to parts and estimating fine-grained semantics of each part. AttentionShift consists of two modules plugged on the vision transformer backbone: (i) token querying for pointly supervised attention map generation, and (ii) key-point shift, which re-estimates part-based attention maps by key-point filtering in the feature space. These two steps are iteratively performed so that the part-based attention maps are optimized spatially as well as in the feature space to cover full object extent. Experiments on PASCAL VOC and MS COCO 2017 datasets show that AttentionShift respectively improves the state-of-the-art of by 7.7% and 4.8% under mAP@0.5, setting a solid PSIS baseline using vision transformer.

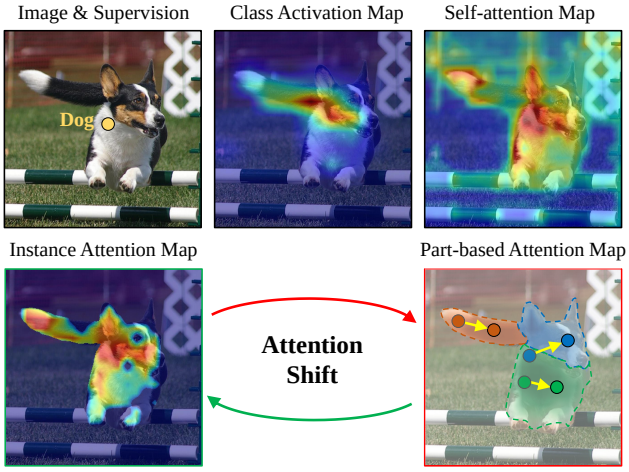

Figure 1. Comparison of existing methods with the proposed AttentionShift. **Upper:** The class activation map (CAM) method [51] suffers partial activation for the lack of spatial constraint. The self-attention map generated by vision transformer [16] encounters false and missed object parts. **Lower:** AttentionShift literately optimizes part-based attention maps (indicated by key-points) by shifting the key-points in the feature space to precisely localize the full object extent. (Best viewed in color)

1. Introduction

Instance segmentation is one of the most important vision tasks with a wide range of applications in medical image processing [21, 23, 36], human machine interface [3, 7, 24] and advanced driver assistance system [14, 37, 41]. Nevertheless, this task requires great human efforts to annotate instance masks, particular in the era of big data. For example, it takes more than four years to create the ground-truth instance masks in the MS COCO dataset by a human annotator [29]. The large annotation cost hinders the deployment of instance segmentation to real-world applications.

Pointly supervised instance segmentation (PSIS) [25, 26], where each instance is indicated by a single point, has been a promising approach to solve the annotation cost issue. Compared with the precise mask annotation, PSIS requires only about 20% annotation cost, which is comparable with the weakly supervised method, while the performance is far beyond the later [2].

Existing methods [25, 26] typically estimate a single pseudo mask for each instance and refine the estimated mask by training a segmentation model. However, such methods ignore the fact that the semantic variance between object parts is non-negligible. For example, a "dog head"

*Equal Contribution (Liao: Idea, Experiment; Guo: Detection Baseline, Writing).
†Corresponding Author.

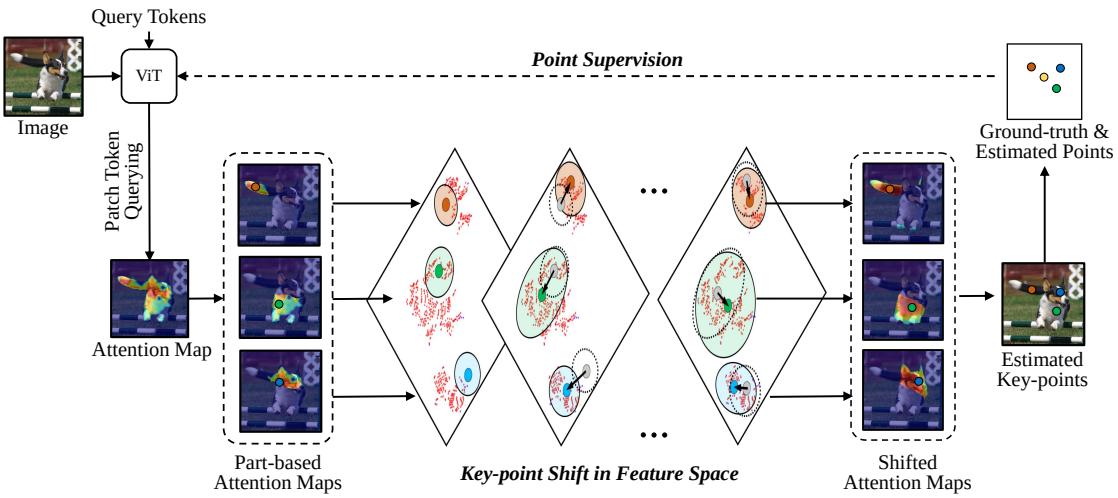

Figure 2. Flowchart of AttenttionShift. During training, the model first learns the fine-grained semantics by decomposing the instance attention map to parts (indicated by key-points) and performing key-point shift the feature space. It then takes the estimated key-points (each represents a part) as supervisions and querying their locations using the vision transformer.

and a "dog tail" belongs to the same semantic of "dog", but their appearances are quite different, Fig. 1 upper. With a single supervision point, these methods could estimate the semantic of "dog" as that of the most discriminative part ("dog tail" in middle of Fig. 1 upper) or that of the regions around the supervision point ("dog head" in right of Fig. 1 upper), which is termed as *semantic bias*. We make a statistical analysis on ViT and found only 33% of ViT attention maps can cover over 50% of foreground pixels. Despite its ability to model long-range feature dependencies, ViT appears to remain vulnerable to the issue of semantic bias.

In this study, we propose AttentionShift to solve the semantic bias problem by estimating fine-grained semantics of multiple instance parts and learning an instance segmentor under the supervision of estimated fine-grained semantics, Fig. 1(lower). Considering that instance parts are unavailable during training, AttentionShift adopts an iterative optimization procedure, which spatially decomposes each instance to parts based on key-points defined on mean feature vectors and adaptively updates the key-points in a way like mean shift [11] in the feature space, Fig. 2.

Using the vision transformer (ViT) as the backbone, AttentionShift consists of two steps: (i) token querying of point supervision for instance attention map generation. It takes advantage of the ViT to generate an instance attention map by matching the semantics and locations of patch tokens with those of the supervision point. (ii) key-point shift which re-estimates the part-based attention map by key-point initialization and filtering in the feature space. These two steps are iteratively performed so that the part-based attention map, indicated by key-points, is optimized spatially as well as in the feature space to cover the full object extent.

We conduct experiments on commonly used PASCAL VOC and the challenging MS COCO 2017 datasets. AttentionShift respectively improves the state-of-the-art of by 7.7% and 4.8% under mAP@0.5, demonstrating the potential to fill the performance gap between pointly-supervised instance segmentation and fully supervised instance segmentation methods.

The contributions of this paper are concluded as follows:

- We propose a part-based attention map estimation approach for PSIS, which estimates fine-grained semantics of instances to alleviate the semantic bias problem in a systematic fashion.
- We represent object parts using key-points and leverage AttentionShift to operate key-points in the feature space, providing a simple-yet-effective fashion to optimize part-based attention maps.
- AttentionShift achieves state-of-the-art performance, setting a solid PSIS baseline using vision transformer.

2. Related Work

Object Localization using Key-points. Key-point features received extensive attention in the past decades. In the era of hand-crafted features, SIFT [32] used orientation-encoded feature channels or bit cyclic shift to find out the scale-invariant key-point locations. When using deep learning models for object detection, key-points including corners [52], centers [52], deformable grids [43], and edge points [39] were used to categorize objects and regress bounding boxes. Key-points have been widely used in human pose estimation, $e.g.$, Convolutional pose machines

(CPM) [40] took advantage of spatial context to predict and refine part confidence maps. Openpose [3] delivered a great real-time pose estimation method by proposing a part affinity field (PAF) to group human parts. ExtremeNet [53] learned the activation maps for extreme points, $e.g.$, top and left ones, to locate objects. RepPoint-V2 [8] introduced foreground/background and corner activation maps to discriminate objects from others. The success of key-point approaches, particularly for part-based feature extraction and localization, inspires us to define attention maps as key-points from a totally new perspective.

Weakly Supervised Detection and Segmentation. It pursues localization ability and discrimination despite being trained on image-level labels. The class activation map (CAM) [51] defined a global average pooling layer and enables the localization ability by mapping the class score is mapped back to the previous convolutional layer. IR-Net [1] converted CAM [51] to a refined instance-aware one that can correctly segment individuals from each other. BESTIE [25] transferred a heat map from semantic knowledge in semantic segmentation results. Nevertheless, when objects come together, the weakly supervised segmentation problem becomes ill-posed as the solution is not unique. For example, when localizing discriminative object parts and/or multiple neighboring objects with a single box/mask, the training objective (image classification) remains being met. Recent methods have introduced adversarial training [10, 34], spatial regularization [5, 6, 33, 34, 42, 45, 49, 50], divergent activation [42, 46] or continuation optimization [38, 44]. However, the ill-posed problem remains, which causes a large performance gap between weakly and fully supervised methods.

Pointly Supervised Detection and Segmentation. The difference between pointly supervised and weakly supervised methods lies in the former uses an additional point within the object extent to indicate the coarse location. Pointly supervised methods eliminate the ambiguity of localization supervision signals, while only increasing the annotation cost by 10% [26]. Through replacing supervision points using those of minimal local responses, BESTIE [25] implemented a promising PSIS approach with center points. Supervised by point annotations, WISE-Net [26] locates a region of each object and segments objects by calculating the similarity between foreground regions. The main reasons for these improvements come from the supervision of relatively accurate instance-level labels. However, the semantic bias caused by the mismatch between the single supervision point and multiple object parts remains unsolved. PointRend [9] proposed to segment instances under the supervision of multiple points manually annotated in bounding boxes. While achieving promising performances, it requires additional bounding-box supervision, which is expensive to acquire. The problem of using a single supervision point to precisely locate all object parts while alleviating semantic bias remains unsolved.

3. AttentionShift

As shown in Fig. 3, AttentionShift consists of two steps: (1) token querying which converts each single point supervision to an instance attention map; (2) Key-point shift which re-estimates the attention map in the feature space and generates a set of points as supervision.

3.1. Token Querying: Pointly Supervised Attention Map Generation

Point Prediction using Query Tokens. To generate attention map for instance segmentation, we define a point prediction task by introducing learnable query tokens. As shown in Fig. 3, an input image is first converted to $W \times H$ image patches (termed patch tokens). These patch tokens are concatenated with the query tokens and fed to the vision transformer (ViT) to extract the token features. The features of query tokens are then passed through two multi-layer perceptron (MLP) branches ("Loc. MLP" and "Cls. MLP") to predict points with class probabilities and point coordinates. The prediction points are matched with and supervised by the GT supervision points using the bipartite matching loss [4]. Note that each GT supervision point can only be matched with a single prediction point, the unmatched prediction points are regarded as background.

Attention Map Generation using Matched Tokens. The query tokens which are matched with the supervision points are thought to be aware of objects. We then used these tokens to generate the attention map for each instance.

To activate accurate object extent, we utilize the self-attention operation of ViT following TS-CAM [16, 27] and produce a self-attention map $A^S \in \mathbb{R}^{W \times H}$ for each instance using the matched query tokens. $A^S_{i,j}$ denotes the feature similarity between the patch token $\phi_{i,j}$ and the matched query token, where the features of patch tokens are denoted as $\Phi = \{\phi_{i,j} \in \mathbb{R}^{1 \times D}, i = 1, 2, ..., W,\ j = 1, 2, ..., H\}$) and D is the dimension of features.

Considering that A^S is learned under the supervision of a single point, it suffers from the semantic bias problem and therefore contains background noise and misses object parts, Fig. 3. We simply use it as the guidance to partition the patch tokens to high confidence foreground Φ^+, high confidence background Φ^- or ignored tokens, by setting thresholds for foreground and background on the A^S. The

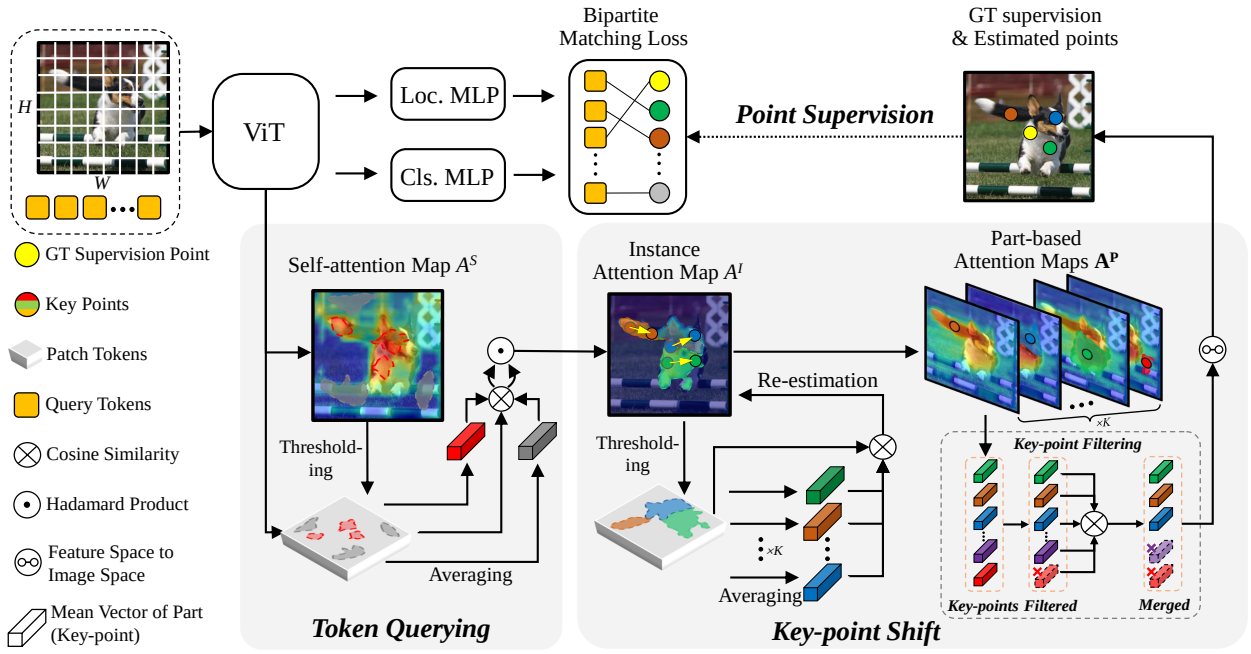

Figure 3. AttentionShift implementation. For each supervision point, token querying produces an instance attention map, which is decomposed to parts through multiple key-points. By key-point shift in the feature space, fine-grained semantics are estimated and the shifted key-points, together with the GT point, are used to supervise the model to update the instance attention map.

instance attention map $A^I \in \mathbb{R}^{W\times H}$ is estimated as

$$A^I_{i,j} = \frac{d(\phi_{i,j}, \psi^+)}{\sum\limits_{\phi_{i,j}\in\Phi^+} d(\phi_{i,j}, \psi^+)} \times \left(1 - \frac{d(\phi_{i,j}, \psi^-)}{\sum\limits_{\phi_{i,j}\in\Phi^-} d(\phi_{i,j}, \psi^-)}\right). \tag{1}$$

where $d(a, b)$ denotes the cosine similarity between vector a and b. $\psi^+ = \frac{1}{|\Phi^+|}\sum_{\phi_{i,j}\in\Phi^+}\phi_{i,j}$ and $\psi^- = \frac{1}{|\Phi^-|}\sum_{\phi_{i,j}\in\Phi^-}\phi_{i,j}$ denote the average feature of patch tokens in Φ^+ and Φ^-, respectively. The first term in Eq. 1 refines the foreground in A^S, while the second term reduces the background noise.

3.2. Key-point Shift: Part-based Attention Map Re-estimation

The instance attention map A^I still suffers from the semantic bias problem as semantics variance among object parts is non-negligibly. With a single point on the head of a dog as supervision, ViT tends to learn the semantic of "dog head" to represent "dog". To alleviate this problem, we propose to partition the instance attention map into parts, representing the part-based attention map by key-points, and estimating the fine-grained semantics by key-point shift.

Part-based Attention Map by Key-points. We propose to decompose the attention map A^I to K part-based attention maps $\mathbf{A}^P = \{A^{P_k} \in \mathbb{R}^{W\times H}, k = 1, ..., K\}$. Nevertheless, precise object parts are unavailable under the point supervision setting. For objects of complex and irregular shapes, it remains challenging to directly learn object parts. Inspired by the assumption that the patch tokens (feature vectors) within an object part share the fine-grained semantics and therefore are dense in the feature space, stable and extreme points within the dense feature vectors are employed to indicate object parts.

To fulfill this purpose, we randomly initialize and adaptively shift K key-points towards stable and extreme points within the feature space. Denote the key-points as $\mathbf{Q} = \{Q_k, k = 1, 2, ..., K\}$, where $Q_k \in \mathbb{R}^2$ are the location indexes of key-points. $Q_k = (i, j)$ denotes that the k-th key-point is closest to the patch token $\phi_{i,j}$ in the feature space. Feature vectors of the key-points are denoted as $\Psi = \{\psi_k \in \mathbb{R}^{1\times D}, k = 1, 2, ..., K\}$. Specially, given an instance attention map A^I, we uniformly sample K key-points $\mathbf{Q}$ on A^I within the high confidence area (as shown in Fig. 3). The key-point feature ψ_k is initialized as ϕ_{Q_k}. Based on the initialized key-points, the instance attention map A^I is decomposed into part-based ones, as

$$A^{P_k}_{i,j} = \frac{d(\phi_{i,j}, \psi_k)}{\sum\limits_{i,j} d(\phi_{i,j}, \psi_k)}, \tag{2}$$

where $A^{P_k}_{i,j}$ denotes the feature similarity between the patch token $\phi_{i,j}$ and the key-point ψ_k.

Key-point Shift. As these key-points are randomly initialized, they can not represent object parts well. We borrow the idea of mean shift [11] and perform key-point shift in the feature space to find the stable and extreme points, which is formulated as

$$\psi_k = \frac{1}{\sum_{i,j} \delta(A_{i,j}^{P_k})} \sum_{\phi_{i,j} \in \Phi^+} \delta(A_{i,j}^{P_k}) \phi_{i,j}, \tag{3}$$

where $\delta(A_{i,j}^{P_k})$ is 1 if $(\arg\max_n A_{i,j}^{P_n}) = k$, and is 0 otherwise. The location of the k-th key-point is computed as

$$Q_k = \arg\max_{(i,j)} d(\phi_{i,j}, \psi_k), \tag{4}$$

Decomposing the attention map using Eq. 2 does not consider feature density. To solve, we introduce a density weight w to the feature of patch token $\phi_{i,j}$ as

$$w(\phi_{i,j}, \psi_k) = \frac{e^{d(\phi_{i,j}, \psi_k)/(\alpha\beta_k)}}{\sum_{i,j} \delta(A_{i,j}^{P_k}) e^{d(\phi_{i,j}, \psi_k)/(\alpha\beta_k)}}, \tag{5}$$

where β_k is an adaptive bandwidth defined as

$$\beta_k = 1 - \frac{1}{\sum_{i,j} \delta(A_{i,j}^{P_k})} \sum_{i,j} \delta(A_{i,j}^{P_k}) d(\phi_{i,j}, \psi_k). \tag{6}$$

β_k is larger when the feature distribution around ψ_k is denser in the feature space. α is a temperature parameter. Weighted by $w(\cdot)$, the patch token $\phi_{i,j}$ which is closer to the key-point ψ_k contributes more to push the key-point ψ_k shifting to the denser sample region in the feature space. Based on the defined weight, Eq. 3 is rewritten as

$$\psi_k = \sum_{\phi_{i,j} \in \Phi^+} \delta(A_{i,j}^{P_k}) w(\phi_{i,j}, \psi_k) \phi_{i,j}. \tag{7}$$

Eq. 7 enables each key-point to move to a sample-dense area in the feature space. Based on shifted key-points, part attention maps are updated by Eq. 2. Iteratively performing Eq. 2 and Eq. 7 N times (empirically, N=10), the key-points independently shift to the mean of dense feature vectors so that the indicated parts have fine-grained semantics.

Key-point Filtering. Despite the effectiveness to localize fine-grained semantics, key-point shift faces noises when falsely shifting to the background area or other objects. We thereby propose to filter noise key-points while merging similar key-points, as detailed in Alg. 1. Alg. 1 consists of two steps. The first step removes key-points that are not highly overlapped with the high confidence area on the attention A^I, which is indicated by $\delta(A_{i,j}^I)$. $\delta(A_{i,j}^I)$ is 1 if $\phi_{i,j} \in \Phi^+$, and 0 otherwise. The second step merges the key-points which are similar with each other in a non-maximum suppression (NMS) fashion.

Algorithm 1 Key-point Filtering.

Input: Key-point features $\Psi = \{\psi_k\}_{k=0}^K$, instance attention map A^I, foreground threshold T_f, and merge threshold T_m

Output: Features of filtered key-points $\Psi' = \{\psi_k\}_{k=0}^{K'}$.

Step 1: Filtering.
for $\psi_k \in \Psi$ **do**
  if $\frac{\sum_{i,j} \delta(A_{i,j}^{P_k}) \delta(A_{i,j}^I)}{\sum_{i,j} \delta(A_{i,j}^{P_k})} < T_f$ **then**
    $\Psi \leftarrow \Psi \backslash \psi_k$
  end if
end for
Step 2: Merging.
$\Psi' \leftarrow \emptyset$
while $\Psi \neq \emptyset$ **do**
  $\mathcal{M} \leftarrow \emptyset$ /*set of key-points to be merged*/
  $m \leftarrow \text{argmax}_k \sum_{i,j} \delta(A_{i,j}^{P_k})$
  $\Psi \leftarrow \Psi \backslash \psi_m$
  $\mathcal{M} \leftarrow \mathcal{M} \cup \psi_m$
  for $\psi_k \in \Psi$ **do**
    if $d(\psi_k, \psi_m) > T_m$ **then**
      $\mathcal{M} \leftarrow \mathcal{M} \cup \psi_k$
      $\Psi \leftarrow \Psi \backslash \psi_k$
    end if
  end for
  $\Psi' \leftarrow \Psi' \cup \text{Mean}(\mathcal{M})$ /*mean feature vector*/
end while
return Ψ'

Attention Map Re-estimation. After performing key-point filtering by Alg. 1, the spatial locations of each key-point are estimated by Eq. 4. Each key-point is employed as a pseudo supervision point, which shares the same class label with its corresponding GT supervision point. With key-point supervisions, features of patch tokens are aware of fine-grained semantics within full object extent, so that the instance attention map is re-estimated according to Eq. 1.

4. Pointly Supervised Instance Segmentation

AttentionShift is implemented based on the Mask RCNN framework [20]. In addition to the point prediction branch illustrated in Fig. 3, it contains a bounding-box detection branch and an instance segmentation branch. Fig. 4 shows the overall framework of AttentionShift.

Point Prediction. This branch has been detailed in Sec. 3. The bipartite matching loss is defined as $\mathcal{L}_P = l_{L1} + l_{cls}$ upon query tokens, where l_{L1} is L1 loss [28] for matched query tokens, and l_{cls} is focal loss [28] for all query tokens.

Bounding-box Detection. The detection branch consists of a region proposal network initialized by a convolutional

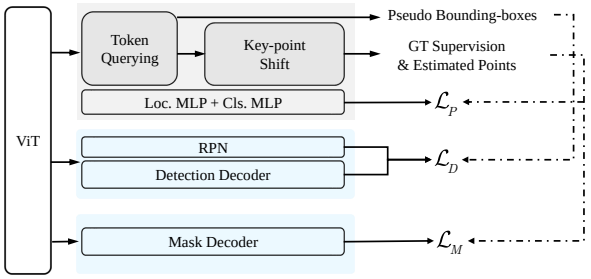

Figure 4. Flowchart of the detailed implementation of the proposed AttentionShift.

and two FC layers, a detection head initialized using the pre-trained transformer decoder [47] and a detection loss with pseudo bounding boxes. To produce the pseudo bounding boxes, we binarize the attention maps A^I [27] and generate tight boxes to enclose the maximum connected area on them. The detection loss is defined as $\mathcal{L}_D = l_{rpn} + l_{det}$, where l_{rpn} is the loss of RPN, and l_{det} the detector head.

Instance Segmentation. The segmentation branch is initialized by a mask decoder following [47], which segments pixels within the object bounding box to foreground and background. The key-points $\mathbf{Q}$ are used as supervision for the foreground. The supervision for the background is generated by sampling pixels in regions where $A^I_{i,j}$ is small. The segmentation loss is defined as $\mathcal{L}_M = l_{ce}$, where l_{ce} denotes cross-entropy loss [20].

Training and Inference. The loss of the proposed method can be defined as $\mathcal{L} = \mathcal{L}_P + \mathcal{L}_D + \mathcal{L}_M$. During training, network parameters are updated with AdamW algorithm. Duing inference, only the detector, and the mask decoder are carried out.

5. Experiment

In this section, we first describe the experimental setting. We then report the performance of AttentionShift and compare it with the state-of-the-art methods. We finally present visualization analysis and ablation studies.

5.1. Setting

Protocols. We evaluate AttentionShift on the augmented Pascal VOC 2012 [35] and MS-COCO [30]. The augmented Pascal VOC 2012, a combination of the original Pascal VOC 2012 and the SBD [18], contains 20 category objects with 10,582 training images and 1,449 *val* images. MS-COCO contains 118k images for training, 5k images for validation, and 20k images for testing. It has 80 object categories collected in natural scenes with object occlusion, clutter backgrounds, and object scale variation, indicating more challenging for instance segmentation. On the augmented VOC 2012 we use the mAP@0.5 [15] metric. Following [1, 25], we also report the results under mAP@0.25 and mAP@0.75. When evaluating on the MS-COCO, the standard AP [29] metric is applied. We also conduct MRCNN refinement following [1, 25].

Table 1. Performance on the Pascal VOC 2012 *val* set. $\mathcal{F}$, $\mathcal{I}$, and $\mathcal{P}$ indicate full mask, image label, and point supervision, respectively. For the method using extra proposals, we use $\mathcal{M}, \mathcal{W}$, and $\mathcal{R}$ to indicate segment proposals, weakly supervised semantic segmentor, and region proposals. "Sup." denotes supervision fashions. † indicates applying MRCNN refinement

Method	Sup.	Extra	mAP_{25}	mAP_{50}	mAP_{75}
Mask R-CNN [20]	$\mathcal{F}$	-	76.7	67.9	44.9
Mask R-CNN(ViT) [47]	$\mathcal{F}$	-	77.2	68.3	46.0
Label-Penet [17]	$\mathcal{I}$	$\mathcal{R}$	49.2	30.2	12.9
CL [22]	$\mathcal{I}$	$\mathcal{M}, \mathcal{R}$	56.6	38.1	12.3
BESTIE [25]	$\mathcal{I}$	$\mathcal{W}$	53.5	41.8	24.2
IRNet [1]	$\mathcal{I}$	-	-	46.7	23.5
WISE-Net [26]	$\mathcal{P}$	$\mathcal{M}$	53.5	43.0	25.9
BESTIE [25]	$\mathcal{P}$	$\mathcal{W}$	58.6	46.7	26.3
AttnShift(ours)	$\mathcal{P}$	-	**68.3**	**54.4**	25.4
BESTIE† [25]	$\mathcal{P}$	$\mathcal{W}$	66.4	56.1	30.2
AttnShift†(ours)	$\mathcal{P}$	-	**70.3**	**57.1**	**30.4**

Implementation Details. AttentionShift follows the training setting of imTED [48]. During Training, random horizontal flips and auto-augmentation on multi-scale ranges are used for data augmentation. AttentionShift is trained with AdamW optimizer with batch size 16 on eight Tesla A100 GPUs. The weight decay and training epoch are 0.05 and 12 respectively. The learning rate is initialized as 0.0001, and reduced by a magnitude after 8 and 11 epochs. We adopt ViT-S [13] pre-trained on ImageNet-1K [12] with MAE method [19] as the backbone network.

5.2. Performance

In Table 1, AttentionShift is compared with SOTA methods on the VOC 2012 *val* set. AttentionShift outperforms the SOTA BESTIE [25] by a significant margin 7.7% (54.4% vs 46.7%) in mAP@0.5, demonstrating that iterative estimation of part-based attention maps facilities activating precise and full object extent. Particularly upon mAP@0.25 metric, AttentionShift achieves 68.3%, nearly 10% better than that of BESTIE. Upon the challenging mAP@0.75 metric, AttentionShift achieves a comparable performance to BESTIE without any extra proposals.

In Table 2, AttentionShift is compared with SOTA methods on MS-COCO *val* and *test-dev*. With ViT-S, AttentionShift ("AttnShift-S") outperforms BESTIE by 1.4% AP (19.1% vs 17.7%). Upon AP@0.5 metric, Attention-

Table 2. Performance on MS-COCO 2017 val and $test\text{-}dev$ set. $\mathcal{S}_{\mathcal{I}}$ indicates training with proposals generated by the salient object detector. "Sup." denotes supervision fashions.

Method	Sup.	Extra	AP	AP50	AP75
COCO val2017					
Mask R-CNN [20]	$\mathcal{F}$	-	35.4	77.3	37.5
Mask R-CNN(ViT) [47]	$\mathcal{F}$	-	38.8	61.2	41.3
BESTIE [25]	$\mathcal{I}$	$\mathcal{W}$	14.3	28.0	13.2
WISE-Net [26]	$\mathcal{P}$	$\mathcal{M}$	7.8	18.2	8.8
BESTIE [25]	$\mathcal{P}$	$\mathcal{W}$	17.7	34.0	16.4
AttnShift-S(ours)	$\mathcal{P}$	-	**19.1**	**38.8**	**17.4**
AttnShift-B(ours)	$\mathcal{P}$	-	**21.2**	**42.0**	**19.4**
COCO test-dev					
Mask R-CNN [20]	$\mathcal{F}$	-	35.4	77.3	37.5
Mask R-CNN(ViT) [47]	$\mathcal{F}$	-	38.9	61.5	41.7
LIID [31]	$\mathcal{I}$	$\mathcal{M}, \mathcal{S}_{\mathcal{I}}$	16.0	27.1	16.5
BESTIE [25]	$\mathcal{I}$	$\mathcal{W}$	14.4	28.0	13.5
BESTIE [25]	$\mathcal{P}$	$\mathcal{W}$	17.8	34.1	16.7
AttnShift-S(ours)	$\mathcal{P}$	-	**19.1**	**38.9**	**17.1**
AttnShift-B(ours)	$\mathcal{P}$	-	**21.9**	**43.5**	**20.1**

Shift significantly outperforms BESTIE by 4.8% (38.8% vs 34.0%). Upon more challenging AP@0.75 metric AttentionShift still outperforms BESTIE by 1.0% (17.4% vs 16.4%). These results shows the superiority of AttentionShift by addressing the semantic bias problem. With ViT-B backbone ("AttnShift-B") further achieves 21.2% upon AP metric, establishing a new SOTA PSIS benchmark.

5.3. Visualization and Analysis

As shown in Fig. 5(second column), the self-attention maps produced by ViT falsely activate backgrounds and/or object parts. By introducing token querying and AttentionShift, the instance activation maps more precisely activate object extent, Fig. 5(third column), which finally produce precise instance segmentation, Fig. 5(the last column).

The progress of key-point shift is visualized in Fig. 6. At the first shift, instance attention map is decomposed to part-based attention maps. But the initial part-based attention maps deviate from real object parts. Key-point shift then makes sure that key-points are consistent with object parts.

With AtenttionShift, the part-based attention maps are refined to represent object parts. The key-point filtering is applied to remove key-points that are shifted to the background area to avoid false semantics. And redundant key-points are merged to prevent the model from focusing on the redundant object parts. By iteratively performing token querying and key-point shift, the part-based attention maps are optimized and the fine-grained semantics are estimated.

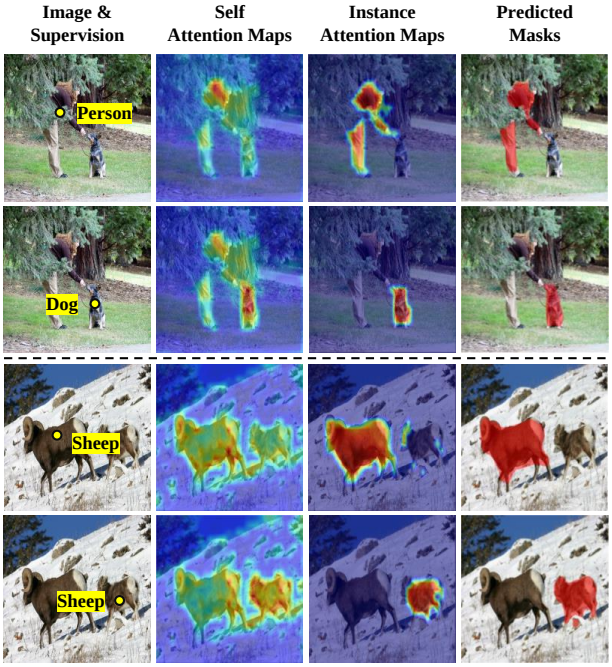

Figure 5. Visualization of the attention maps and instance masks.

5.4. Ablation Study

In this section, we first introduce the baseline. The we conduct experiments to analyze the effect of attention map generation, key-point shift, and key-point filtering. All results in this section are evaluated on the VOC 2012 val set.

Baseline. Following the previous WSOD [16, 27], we leverage the self-attention maps in ViT to generate pseudo-bounding boxes and use them as the supervision of the detection branch. The baseline generate key-points by randomly selecting points in the high-confidence area of the self-attention map without shifting and filtering. As shown in Table 3, the baseline method achieves 38.0% mAP.

Attention Map Generation. We replace the self-attention map with the instance attention map. As shown in Table 3, the mAP is significantly improved by 8.8% (46.8% vs 38.0%), which indicates the effectiveness of instance attention map generalization with noise reduction.

Key-point Shift. We introduce key-point shift to decompose the generated attention map into part-based ones. As shown in Table 3, introducing key-point shift further improves the mAP by 2.4% (49.2% vs 46.8%), indicating the semantic bias problem can be reduced by estimating fine-grained semantics of object parts.

We further conduct experiments with different hyper-parameters α and N in Table 4. When $N = 10$, $\alpha = 1.0$

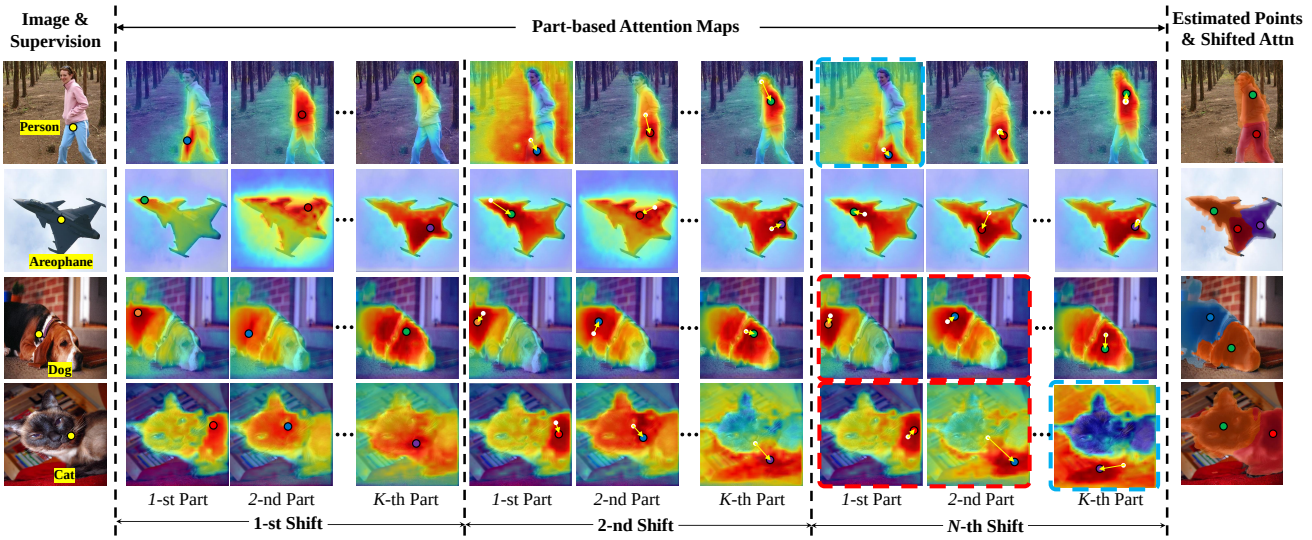

Figure 6. Visualization of iterative estimation of part-based attention maps using the proposed token querying and key-point shift procedure. (Best viewed in color)

Table 3. Ablation studies of AttentionShift modules.

Baseline	Attention Map Generation	Key-point Shift	Key-point Filtering	mAP
✓				38.0
	✓			46.8
	✓	✓		49.2
	✓	✓	✓	53.2

Table 4. Performance $w.r.t.$ hyper-parameters α and N in key-point shift.

α	0.1	1.0	10	Inf	1.0	1.0	1.0	1.0
N	10	10	10	10	0	5	10	20
mAP	51.4	53.2	49.1	49.8	50.3	51.8	53.2	52.5

Table 5. Performance $w.r.t$, threshold T_f and T_m in key-point filtering.

T_f	0.85	0.85	0.85	0.80	0.85	0.90
T_m	0.80	0.85	0.90	0.85	0.85	0.85
mAP	51.1	53.2	51.3	53.0	53.2	52.3

reports the best mAP (53.2%). When $\alpha = \text{Inf}$, the proposed key-point shift (defined by Eq. 2 and Eq. 7) degrades into vanilla key-point shift (defined by Eq. 2 and Eq. 3), which reduces the mAP by 3.4% (49.8% vs 53.2%). With $\alpha = 1.0$, mAP increase with N becomes larger, and the best mAP is achieved with $N = 10$. Note that more iterations of Eq. 2 and Eq. 7 ($N = 20$) do not further improve the mAP.

Key-point Filtering. In Table 3, Key-point filtering further improves the mAP by 4.0% (53.2% vs 49.2%) , by removing the noisy and redundant key-points.

Table 5 shows the impact of thresholds T_f and T_m. AttentionShift is robust to T_f when $T_f \leq 0.85$. However, when T_f becomes too large (T_f=0.9), foreground parts could be falsely filtered, which hurts the performance by $\sim$1%. In comparison, the merging threshold T_m has a larger impact. The best mAP (53.2%) is achieved when $T_m = 0.85$. Larger or smaller T_m causes $\sim$2% mAP drop.

6. Conclusion

In this study, we proposed an AttentionShift approach, to solve the semantic bias issue of pointly supervised instance segmentation. Our approach used key-points to represent attention maps, as well as leveraging the mean shift algorithm in the feature space to perform part-based attention segmentation. It iteratively decomposed the instance attention map to parts through key-points and estimated fine-trained semantics of each part, so that segmentation is optimized spatially and in the feature space. Extensive experiments on large-scale datasets validated the performance of AttentionShift, with striking contrast to the state-of-the-art methods. AttentionShift built a solid baseline for pointly supervised instance segmentation with vision transformer.

Acknowledgement:This work was supported by National Natural Science Foundation of China (NSFC) under Grant 62006216, 61836012, and 62225208, the Fundamental Research Funds for the Central Universities, and the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDA27000000.

References

[1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *IEEE CVPR*, pages 2209–2218, 2019. 3, 6

[2] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, pages 549–565, 2016. 1

[3] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *arXiv:1812.08008*, 2018. 1, 3

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, volume 12346, pages 213–229, 2020. 3

[5] Nenglun Chen, Xingjia Pan, Runnan Chen, Lei Yang, Zhiwen Lin, Yuqiang Ren, Haolei Yuan, Xiaowei Guo, Feiyue Huang, and Wenping Wang. Distributed attention for grounded image captioning. *arXiv preprint arXiv:2108.01056*, 2021. 3

[6] Pengfei Chen, Xuehui Yu, and Xumeng Han *et al*. Point-to-box network for accurate object detection via single point supervision. In *ECCV*, 2022. 3

[7] Xingyu Chen, Yufeng Liu, Yajiao Dong, Xiong Zhang, Chongyang Ma, Yanmin Xiong, Yuan Zhang, and Xiaoyan Guo. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In *IEEE CVPR*, pages 20512–20522, 2022. 1

[8] Yihong Chen, Zheng Zhang, Yue Cao, Liwei Wang, Stephen Lin, and Han Hu. Reppoints v2: Verification meets regression for object detection. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020. 3

[9] Bowen Cheng, Omkar Parkhi, and Alexander Kirillov. Pointly-supervised instance segmentation. In *IEEE CVPR*, pages 2607–2616, 2022. 3

[10] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *IEEE CVPR*, pages 2219–2228, 2019. 3

[11] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI*, 24:603–619, 2002. 2, 5

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *IEEE CVPR*, pages 248–255. IEEE CS, 2009. 6

[13] 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 *ICLR*, 2021. 6

[14] Francis Engelmann, Martin Bokeloh, Alireza Fathi, Bastian Leibe, and Matthias Nießner. 3d-mpa: Multi-proposal aggregation for 3d semantic instance segmentation. In *IEEE CVPR*, pages 9028–9037, 2020. 1

[15] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.*, pages 303–338, 2010. 6

[16] Wei Gao, Fang Wan, Xingjia Pan, Zhiliang Peng, Qi Tian, Zhenjun Han, Bolei Zhou, and Qixiang Ye. TS-CAM: token semantic coupled attention map for weakly supervised object localization. In *IEEE ICCV*, pages 2886–2895, 2021. 1, 3, 7

[17] Weifeng Ge, Weilin Huang, Sheng Guo, and Matthew R. Scott. Label-penet: Sequential label propagation and enhancement networks for weakly supervised instance segmentation. In *IEEE ICCV*, pages 3344–3353, 2019. 6

[18] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998, 2011. 6

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 6

[20] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *IEEE ICCV*, 2017. 5, 6, 7

[21] Joy Hsu, Wah Chiu, and Serena Yeung. DARCNN: domain adaptive region-based convolutional neural network for unsupervised instance segmentation in biomedical images. In *IEEE CVPR*, pages 1003–1012, 2021. 1

[22] Jaedong Hwang, Seohyun Kim, Jeany Son, and Bohyung Han. Weakly supervised instance segmentation by deep community learning. In *IEEE WACV*, pages 1019–1028, 2021. 6

[23] Wei Ji, Shuang Yu, Junde Wu, Kai Ma, Cheng Bian, Qi Bi, Jingjing Li, Hanruo Liu, Li Cheng, and Yefeng Zheng. Learning calibrated medical image segmentation via multi-rater agreement modeling. In *IEEE CVPR*, pages 12341–12351, 2021. 1

[24] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J. Kim. HOTR: end-to-end human-object interaction detection with transformers. In *IEEE CVPR*, pages 74–83, 2021. 1

[25] Beomyoung Kim, Youngjoon Yoo, Chaeeun Rhee, and Junmo Kim. Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement. In *IEEE CVPR*, pages 4268–4277, 2022. 1, 3, 6, 7

[26] Issam H. Laradji, Negar Rostamzadeh, Pedro O. Pinheiro, David Vázquez, and Mark Schmidt. Proposal-based instance segmentation with point supervision. In *IEEE ICIP*, pages 2126–2130, 2020. 1, 3, 6, 7

[27] Mingxiang Liao, Fang Wan, Yuan Yao, Zhenjun Han, Jialing Zou, Yuze Wang, Bailan Feng, Peng Yuan, and Qixiang Ye. End-to-end weakly supervised object detection with sparse proposal evolution. In *ECCV*, 2022. 3, 6, 7

[28] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE ICCV*, pages 2999–3007, 2017. 5

[29] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, pages 740–755, 2014. 1, 6

[30] 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 *ECCV*, pages 740–755, 2014. 6

[31] Yun Liu, Yu-Huan Wu, Peisong Wen, Yujun Shi, Yu Qiu, and Ming-Ming Cheng. Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation. *IEEE TPAMI*, 44:1415–1428, 2022. 7

[32] David G. Lowe. Object recognition from local scale-invariant features. In *IEEE ICCV*, pages 1150–1157, 1999. 2

[33] Weizeng Lu, Xi Jia, Weicheng Xie, Linlin Shen, Yicong Zhou, and Jinming Duan. Geometry constrained weakly supervised object localization. *arXiv preprint arXiv:2007.09727*, 2020. 3

[34] Jinjie Mai, Meng Yang, and Wenfeng Luo. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *IEEE CVPR*, pages 8766–8775, 2020. 3

[35] Everingham Mark, Van Gool Luc, Williams Christopher KI, Winn John, and Zisserman Andrew. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 6

[36] Simon Reiß, Constantin Seibold, Alexander Freytag, Erik Rodner, and Rainer Stiefelhagen. Every annotation counts: Multi-label deep supervision for medical image segmentation. In *IEEE CVPR*, pages 9532–9542, 2021. 1

[37] Thang Vu, Kookhoi Kim, Tung Minh Luu, Thanh Nguyen, and Chang D. Yoo. Softgroup for 3d instance segmentation on point clouds. In *IEEE CVPR*, pages 2698–2707, 2022. 1

[38] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. C-mil: Continuation multiple instance learning for weakly supervised object detection. In *IEEE CVPR*, pages 2199–2208, 2019. 3

[39] F. Wei, X. Sun, H. Li, J. Wang, and S. Lin. Point-set anchors for object detection, instance segmentation and pose estimation. In *ECCV*, 2020. 2

[40] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *IEEE CVPR*, pages 4724–4732, 2016. 3

[41] Xinli Xu, Shaocong Dong, Lihe Ding, Jie Wang, Tingfa Xu, and Jianan Li. Fusionrcnn: Lidar-camera fusion for two-stage 3d object detection. *arXiv preprint arXiv:2209.10733*, 2022. 1

[42] Haolan Xue, Chang Liu, Fang Wan, Jianbin Jiao, Xiangyang Ji, and Qixiang Ye. Danet: Divergent activation for weakly supervised object localization. In *IEEE ICCV*, pages 6589–6598, 2019. 3

[43] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. Reppoints: Point set representation for object detection. In *IEEE ICCV*, pages 9656–9665, 2019. 2

[44] Qixiang Ye, Fang Wan, Chang Liu, Qingming Huang, and Xiangyang Ji. Continuation multiple instance learning for weakly and fully supervised object detection. *IEEE TNNLS*, pages 1–15, 2021. 3

[45] Xuehui Yu, Pengfei Chen, and Di Wu *et al*. Object localization under single coarse point supervision. In *CVPR*, 2022. 3

[46] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *IEEE ICCV*, pages 6023–6032, 2019. 3

[47] Xiaosong Zhang, Feng Liu, Zhiliang Peng, Zonghao Guo, Fang Wan, Xiangyang Ji, and Qixiang Ye. Integral migrating pre-trained transformer encoder-decoders for visual object detection. *arXiv:2205.09613*, 2022. 6, 7

[48] Xiaosong Zhang, Feng Liu, Zhiliang Peng, Zonghao Guo, Fang Wan, Xiangyang Ji, and Qixiang Ye. Integral migrating pre-trained transformer encoder-decoders for visual object detection. *arXiv preprint arXiv:2205.09613*, 2022. 6

[49] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *IEEE CVPR*, pages 1325–1334, 2018. 3

[50] Xiaolin Zhang, Yunchao Wei, Guoliang Kang, Yi Yang, and Thomas Huang. Self-produced guidance for weakly-supervised object localization. In *ECCV*, pages 597–613, 2018. 3

[51] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE CVPR*, pages 2921–2929, 2016. 1, 3

[52] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 2

[53] Xingyi Zhou, Jiacheng Zhuo, and Philipp Krähenbühl. Bottom-up object detection by grouping extreme and center points. In *IEEE CVPR*, pages 850–859, 2019. 3