

Meta-Point Learning and Refining for Category-Agnostic Pose Estimation

Junjie Chen¹ Jiebin Yan^{1*} Yuming Fang¹ Li Niu²

¹ Jiangxi University of Finance and Economics,

² Shanghai Jiao Tong University

{chenjunjie, yanjiebin, fangyuming}@jxufe.edu.cn, ustcnewly@sjtu.edu.cn

Abstract

Category-agnostic pose estimation (CAPE) aims to predict keypoints for arbitrary classes given a few support images annotated with keypoints. Existing methods only rely on the features extracted at support keypoints to predict or refine the keypoints on query image, but a few support feature vectors are local and inadequate for CAPE. Considering that human can quickly perceive potential keypoints of arbitrary objects, we propose a novel framework for CAPE based on such potential keypoints (named as meta-points). Specifically, we maintain learnable embeddings to capture inherent information of various keypoints, which interact with image feature maps to produce meta-points without any support. The produced meta-points could serve as meaningful potential keypoints for CAPE. Due to the inevitable gap between inherency and annotation, we finally utilize the identities and details offered by support keypoints to assign and refine meta-points to desired keypoints in query image. In addition, we propose a progressive deformable point decoder and a slacked regression loss for better prediction and supervision. Our novel framework not only reveals the inherency of keypoints but also outperforms existing methods of CAPE. Comprehensive experiments and in-depth studies on large-scale MP-100 dataset demonstrate the effectiveness of our framework. Code is available at <https://github.com/chenbys/MetaPoint>

1. Introduction

Pose estimation is a fundamental and significant computer vision task, which aims to produce the locations of pre-defined semantic part of object instance in 2D image. Recently, it has received increasing attention in the computer vision community due to its wide applications in virtual reality, augmented reality, human-computer interaction, robot and automation. However, most pose estimation methods are trained with category-specific data and thus cannot be

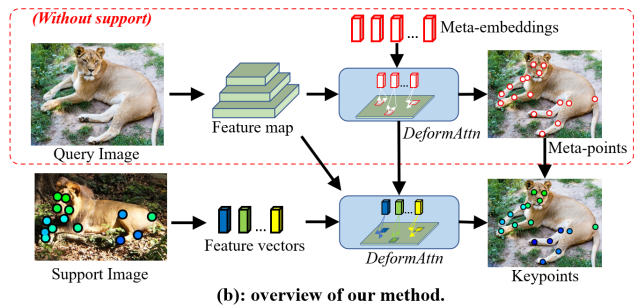
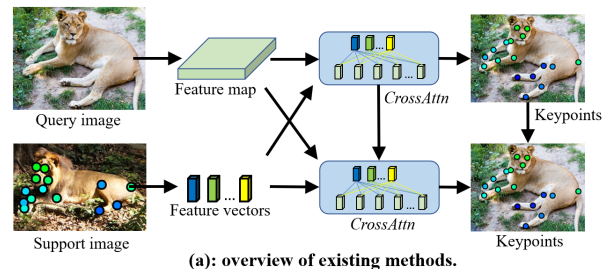


Figure 1. Overview of existing methods and our method. (a): Existing methods only rely on the local features at support keypoints to predict or refine the keypoints. (b): Our method employs learnable embeddings to capture inherent information and produces meta-points without support. Then our method assigns and refines meta-points according to support keypoints.

applied for novel classes, especially when they have different keypoint classes. Therefore, category-agnostic pose estimation (CAPE) [41, 48] is recently proposed to localize desired keypoints for arbitrary classes given one or few support images annotated with keypoints.

Unlike the standard pose estimation task where the keypoint classes are pre-defined, the desired keypoints in CAPE are defined by support keypoints. Therefore, existing methods [41, 48] intuitively extract features at support keypoints and then rely on them to predict or refine the keypoints on query image, as shown in Fig. 1 (a). Although such straightforward pipeline has achieved considerable success, the information provided by a few feature vectors are local

*Corresponding author

[32, 45] and inadequate, especially when support keypoints are partially occluded or blurred. Furthermore, above methods focus on learning pixel-level correspondences and neglect to mine inherent and complementary information of keypoints, which is a fundamental issue of CAPE.

We conjecture that there are some inherent or universal points in each object, based on the insight that human can quickly grasp the essentials of keypoints of arbitrary objects without any reference or support image. We refer to such class-agnostic potential points as meta-points, which may seem similar to object proposal [1, 22, 57] or mask proposal [23, 25, 37] but actually have significant characteristics. For single object, meta-points could provide fine-grained and structural information with a sparse representation. For two objects, meta-points of the same semantic part should have consistent contact. For example, a lion and a bird may have consistent meta-points (*e.g.*, eye) and individual meta-points (*e.g.*, wing). Therefore, meta-point learning is a fundamental yet challenging issue.

In this paper, we propose a novel pipeline for CAPE, which first predicts meta-points without support and then assign and refine them to desired keypoints according to support, as illustrated in Fig. 1 (b). Specifically, we maintain learnable meta-embeddings to store the inherent information of various keypoints for producing meta-points. Given a query image, these meta-embeddings will interact with its feature maps via a transformer decoder to mine the inherent information and thus could produce meta-points without support. To bridge the gap between inherent meta-points and desired keypoints, we first employ bipartite matching-based assignment to assign optimal meta-points for desired keypoints according to support. Afterwards, we employ the support feature vectors together with the mined inherent information to refine assigned meta-points via another transformer decoder. Thanks to the meta-points and mined inherent information, our framework could produce more precise keypoints for CAPE. In addition, we propose a progressive deformable point decoder, which progressively mines fine-grained features to decode points with last points as reference based on deformable attention [56]. We also propose a slacked regression loss to reduce the immature gradients of auxiliary decoder layers.

We evaluate our framework on the largest dataset for CAPE, *i.e.*, Multi-category Pose (MP-100) dataset [48]. The in-depth studies demonstrate that our method could learn meaningful meta-points for arbitrary classes without support. And the comprehensive experiments indicate that our meta-points could dramatically facilitate CAPE and lead to better performance against existing methods. Our contributions could be summarized as: **1)** To the best of our knowledge, we are the first to learn class-agnostic potential keypoints for CAPE. **2)** We propose a novel framework for CAPE based on meta-point learning and refining. **3)** Two

minor contributions are progressive deformable point decoder and slacked regression loss for better prediction and supervision. **4)** Our method not only produces meaningful meta-points but also predicts more precise keypoints against existing methods in CAPE.

2. Related Works

2.1. Category-Specific Pose Estimation

Pose estimation is a fundamental task in computer vision, which aims to localize the pre-defined keypoints of objects in the input image. Most pose estimation methods are class-specific, *e.g.*, estimating pose for human [2, 27], animals [4, 26], and vehicles [38, 43]. Technically, existing pose estimation methods could be roughly categorized into regression-based methods [19, 28, 36], heatmap-based methods [11, 13, 54], and query-based methods [34, 40, 50]. To name a few, Li *et al.* [28] proposed a novel regression paradigm with Residual Log-likelihood Estimation to learn the change of the distribution to facilitate the training process. Geng *et al.* [19] proposed to employ multi-branch structure for disentangled keypoint regression, which could be able to attend to the keypoint regions and thus improve the performances. Luo *et al.* [33] proposed to estimate scale-adaptive heatmaps, which adaptively adjusts the standard deviation for each keypoint and thus is tolerant of various scales and ambiguities. Shi *et al.* [40] proposed to learn multiple pose queries for reasoning a set of full-body poses and then use a joint decoder to refine by exploring kinematic relations. Although above works have achieved great success for estimating pose of specific classes, they are inapplicable for novel classes, especially when the novel classes have different number and kinds of keypoints. In this work, we focus on estimating pose for arbitrary classes with a few support images annotated with desired keypoints.

2.2. Category-Agnostic Pose Estimation

Few-shot learning [46] focuses on learning novel classes using only a few samples and has been applied in extensive visual tasks, *e.g.*, classification [6, 12], object detection [16, 55], segmentation [9, 29, 53]. For keypoint estimation, previous few-shot methods mostly focus on specific domains, *e.g.*, facial images [3, 47], clothing images [18], or X-ray images [10, 52]. For wider classes, Lu *et al.* [30] explored a flexible few-shot scenario including novel/base classes and novel/base keypoints. Later, Lu *et al.* [31] proposed a saliency-guided transformer for above scenario and transductive scenario. He *et al.* [21] explored to estimate keypoints using a few examples combined with unlabeled images. Recently, POMNet [48] elaborated a large-scale dataset including 100 classes and set the task of category-agnostic pose estimation. Technically, POMNet [48] proposed a keypoint matching framework, which extracts the

features at support keypoints and employs their matching similarities to every pixels to retrieve the desired keypoints. Afterwards, CapeFormer [41] enhanced the similarity modeling in above matching pipeline, and proposed to further refine each keypoint via a elaborate transformer decoder. Although above methods [41, 48] have great promoted the development of CAPE, they only rely on the local features at support keypoints and neglect to mine inherent and universal information of keypoints. In contrast, we propose to mine the inherent or universal points in objects without any support, and propose a progressive deformable point decoder to progressively mines fine-grained features to decode points more precisely.

2.3. Category-Agnostic Proposal Learning

Category-agnostic proposal learning is a long-standing research topic in computer vision and has extensive downstream tasks. Early methods rely on low-level features (*e.g.*, edges, super-pixels, or saliency) to produce bounding box proposals [1, 57] or mask proposals [7, 8, 23, 25]. Recent methods employ various network to learn box proposals [24, 39] or mask proposals [15, 49]. To name a few, one of the most impressive works is Faster R-CNN [39], which first learns class-agnostic box proposals and then refines them to object bounding boxes. Pinheiro *et al.* [37] proposed to learn the class-agnostic segmentation masks and their object likelihood scores. Therefore, proposal learning is a significant and foundational task and could greatly facilitate downstream tasks in the second stage. Nevertheless, there are few works on learning class-agnostic keypoints, and we propose to learn potential keypoints and name our learned points as meta-points. Basically, our meta-point could provide fine-grained and structural information with a sparse representation. Furthermore, our meta-points have consistent contact for the same semantic part of different objects. Therefore, such meta-points not only reveals the inherency of keypoints but also serves as meaningful proposals for downstream tasks. Although the coarse keypoints in [41] are named as “proposals”, their “proposals” are predicted according to support keypoints and thus do not have the class-agnostic properties.

3. Method

Class-agnostic pose estimation (CAPE) aims to estimate desired keypoints on query image given a few support images annotated with keypoints for novel classes. For N -shot setting, CAPE could be formulated as a function:

$$\mathbf{P}_q = \mathcal{F}_{cape}(\mathbf{x}_q, \{\mathbf{x}_{s_n}\}_{n=1}^N, \{\mathbf{P}_{s_n}^*\}_{n=1}^N) \quad (1)$$

where N support images $\{\mathbf{x}_{s_n}\}_{n=1}^N$ and their ground-truth (GT) keypoints $\{\mathbf{P}_{s_n}^*\}_{n=1}^N$ are given, and the task is to estimate the desired keypoints \mathbf{P}_q on query image \mathbf{x}_q . To

evaluate the generalization capacity for arbitrary classes, all classes in benchmark are split into non-overlapping base classes and novel classes, and query and support images are sampled from only base (*resp.*, novel) classes for training (*resp.*, evaluating). For simplicity, we first introduce our method in 1-shot setting, and then clarify the intuitive extension with more support images. Specifically, the function is simplified as $\mathbf{P}_q = \mathcal{F}_{cape}(\mathbf{x}_q, \mathbf{x}_s, \mathbf{P}_s^*)$, where $\mathbf{x}_q, \mathbf{x}_s \in \mathbb{R}^{3 \times H \times W}$ and $\mathbf{P}_q, \mathbf{P}_s^* \in \mathbb{R}^{K \times 2}$. For brevity of description, we employ subscript to indicate the variable source and use square bracket to show the index in variable, *e.g.*, $\mathbf{P}_q[k]$ for the k -th keypoint on query image. To tackle such problem, we propose a novel framework based on meta-point learning and refining, which not only reveals the keypoint inherency but also produces precise keypoints. The detailed framework is illustrated in Fig. 2, we first introduce the pipelines of two-stages and then clarify the architecture of our point decoder used in pipelines.

3.1. Predicting Meta-Point without Support

Considering that human is able to quickly abstract keypoints for arbitrary classes without support, we propose to model such fundamental ability to facilitate CAPE. As shown in the upper part of Fig. 2, we maintains M trainable meta-embeddings $\tilde{\mathbf{E}} \in \mathbb{R}^{M \times D}$ in our model (*e.g.*, $M = 100$), which are shared by all samples and store the inherent and universal information of keypoints. For the input sample \mathbf{x}_q , our model first employs ResNet-50 [20] backbone to extract pyramid feature maps $\{\mathbf{F}_q^r\}_{r=1}^R$, where $\mathbf{F}_q^r \in \mathbb{R}^{D^r \times H^r \times W^r}$ corresponds to the r -th residual layer in deep-to-shallow order ($R = 3$ by default). Afterwards, the meta-embeddings interact with image feature maps to mine inherent information, which could be summarized as:

$$\left\{ \left[\tilde{\mathbf{P}}_q^l; \tilde{\mathbf{E}}_q^l \right] \right\}_{l=1}^L = \mathcal{D}_{meta}^L(\tilde{\mathbf{E}}, \tilde{\mathbf{P}}_{init}, \{\mathbf{F}_q^r\}_{r=1}^R). \quad (2)$$

Specifically, \mathcal{D}_{meta} is implemented with our Progressive Deformable Point Decoder, which employs DeformAttn[56] to progressively decode L layers of points with last points as reference points and will be clearly introduced in Sec. 3.3. $\tilde{\mathbf{P}}_{init} \in \mathbb{R}^{M \times 2}$ are the initial points, and we adopt uniform grid points to launch the decoding for meta-points. From the l -th decoder layer in \mathcal{D}_{meta} , we obtain the query meta-points $\tilde{\mathbf{P}}_q^l \in \mathbb{R}^{M \times 2}$ and query meta-embeddings $\tilde{\mathbf{E}}_q^l \in \mathbb{R}^{M \times D}$. We use the predictions from the last layer as the proposals for following processes. Because not all meta-points are visible for arbitrary objects, we employ a lightweight binary classifier to predict the visibilities $\tilde{\mathbf{v}}_q \in \mathbb{R}^M$ according to the last query meta-embeddings.

In this way, we are able to produce meta-points $\tilde{\mathbf{P}}_q^L$ and their visibilities $\tilde{\mathbf{v}}_q$ without any support, which reveal the inherency of keypoints and serve as meaningful proposals to facilitate downstream tasks.

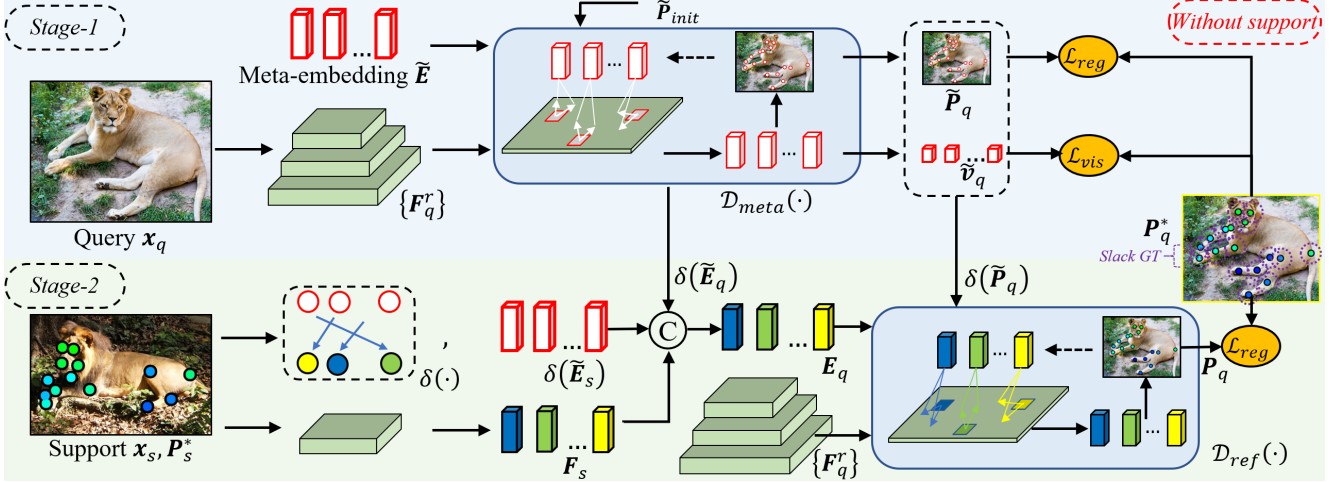


Figure 2. Our framework employs two stages to predict meta-points and desired keypoints. In the first stage, the learnable meta-embeddings interact with query feature maps via our progressive deformable point decoder to mine inherent information to predict meta-points and their visibilities. In the second stage, the meta-points are assigned with identities according to the given support keypoints. After that, the assigned meta-points are refined to desired keypoints based the support features and mined inherent information via another point decoder.

3.2. Refining Meta-Point with Support

Due to the inevitable gap between keypoint inherency and annotation, we have to refine above meta-points to desired keypoints P_q with GT support keypoints P_s^* . The support information includes two aspects, *i.e.*, desired identity and desired detail, which will be described as follows.

3.2.1 Identity Assignment

Each annotated keypoint has its identity/index, and thus we need to determine the index mapping $\delta(\cdot)$ between meta-points and desired keypoints, *i.e.*, the $\delta(k)$ meta-point corresponds to the k -th keypoint. Specifically, we employ the bipartite matching-based assignment as in [5, 14] to solve the mapping according to a cost matrix $\mathcal{C} \in \mathbb{R}^{M \times K}$, where each entry $\mathcal{C}[m, k]$ indicates the cost for assigning the m -th meta-point to the k -th annotated keypoint. To solve the assignment according to support, we obtain the meta-points $\{\tilde{P}_s^l\}_{l=1}^L$ and visibilities \tilde{v}_s on support image as the same pipeline in Sec. 3.1. Then, we compute the cost by:

$$\mathcal{C}_s[m, k] = \sum_{l=1}^L \mathcal{L}_{sl_1}(\tilde{P}_s^l[m], P_s^*[k], l) - \alpha \log(\tilde{v}_s[m]), \quad (3)$$

where \mathcal{L}_{sl_1} measures the l_1 deviation with slacks on auxiliary decoder layers and will be introduced in Sec. 3.4. The second term considers the classification of meta-point visibility, with a trade-off hyper-parameter $\alpha = 0.5$. Finally, we use \mathcal{C}_s to obtain the optimal assignment $\delta(\cdot)$. In this way, each desired keypoint is assigned with a meta-point, *i.e.*, the k -th keypoint is assigned with $\tilde{P}[\delta(k)]$. Be-

cause we set $K \leq M$, and the unassigned meta-points are treated as ignore. For brevity, we employ $\delta(\tilde{P}) \in \mathbb{R}^{K \times 2}$ and $\delta(\tilde{E}) \in \mathbb{R}^{K \times D}$ to denote the assigned meta-points and meta-embeddings re-ordered in desired sequence.

3.2.2 Details Enhancement

Besides the desired identities, we refine meta-points to desired coordinates according to the support information. By default, we employ the prediction of the last decoder layer in \mathcal{D}_{meta} to refine and omit the decoder layer index for brevity, *i.e.*, we use assigned meta-points $\delta(\tilde{P}_q)$ and their embeddings $\delta(\tilde{E}_q)$ to launch the refinement. Basically, we derive support keypoint features F_s via “soft” ROI pooling as in previous works [41, 48], which contain information of local details in support image. We also collect the meta-embeddings assigned to support keypoints $\delta(\tilde{E}_s)$, which have absorbed the global inherent information in support image. Afterwards, we squeeze the concatenation of above embeddings to obtain comprehensive embeddings $E_q \in \mathbb{R}^{K \times D}$ for keypoints, *i.e.*, $E_q = f_{sqz}([\delta(\tilde{E}_q); \delta(\tilde{E}_s); F_s])$. Then, the assigned meta-points $\delta(\tilde{P}_q)$ are progressively refined according to E_q and image feature maps, as:

$$\{[P_q^l; E_q^l]\}_{l=1}^L = \mathcal{D}_{ref}^L(E_q, \delta(\tilde{P}_q), \{F_q^r\}_{r=1}^R). \quad (4)$$

Specifically, we also employ our Progressive Deformable Point Decoder to implement \mathcal{D}_{ref} (same architecture but different parameters with \mathcal{D}_{meta}) and leave the details to Sec. 3.3. In this way, our decoder \mathcal{D}_{ref} progressively refines assigned meta-points to desired keypoints.

3.3. Progressive Deformable Point Decoder

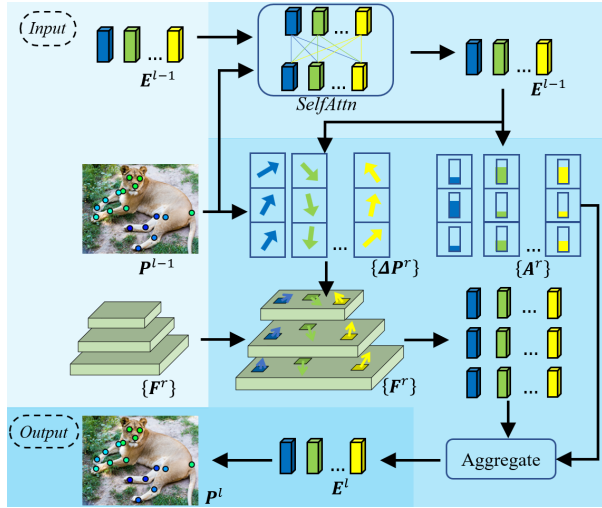


Figure 3. Illustration of decoder layer in our Progressive Deformable Point Decoder. The input embeddings first interact with each other via a self-attention module. After that, the offsets and weights are predicted to mine fine-grained features on input feature maps. Finally, a deformable attention module uses input points as reference to refine the embeddings and points.

Predicting precise keypoints requires fine-grained and high-resolution features, and thus we propose a progressive deformable point decoder, which makes full use of deformable attention module [56] to precisely refine keypoints. Our decoder consists of L decoder layers ($L = 3$ by default), and the architecture of decoder layer is shown in Fig. 3. Given points $P^{l-1} \in \mathbb{R}^{K \times 2}$ and their embeddings $E^{l-1} \in \mathbb{R}^{K \times D}$, each decoder layer performs self attention [44] and deformable attention [56] to update points and embeddings according to feature maps $\{F^r\}_{r=1}^R$, as:

$$[P^l; E^l] = \mathcal{D}(E^{l-1}, P^{l-1}, \{F^r\}_{r=1}^R). \quad (5)$$

By stacking multiple such layers, our decoder could update the points in a progressive and precise manner. Specifically, each decoder layer first employs a self attention module to render each point aware of other points' positions and contents as in [41] (also with identity embeddings and position embeddings). Afterwards, each embedding predicts the offsets $\{\Delta P^r\}_{r=1}^R$ and weights $\{A^r\}_{r=1}^R$ for paying attention to each feature map. By using the points P^{l-1} as reference points, the deformable attention module aggregates sampled features according to weights and obtain updated embeddings E^l . Then, the coordinates of points are refined by:

$$P^l = \sigma(\sigma^{-1}(P^{l-1}) + \mathcal{F}_{mlp}(E^l)), \quad (6)$$

where \mathcal{F}_{mlp} is a lightweight point head and σ indicates Sigmoid function. In this way, input points and their embed-

dings could be progressively refined according to the nearby fine-grained features.

3.4. Training and Inference

Our full training objective consists of keypoint regression loss and visibility classification loss. For the regression loss, strict supervision on auxiliary decoder layers will trigger immature gradients for fitting keypoints with inadequate capacities. And thus we adopt a slacked L_1 loss:

$$\mathcal{L}_{sl_1}(P, P^*, l) = \sum_k \varepsilon(|P[k] - P^*[k]|_1 - \lambda \cdot (L - l)), \quad (7)$$

where ε indicates ReLU function and $\lambda = 0.1$ is a slack hyper-parameter. And our total regression loss is

$$\mathcal{L}_{reg} = \sum_{l=1}^L \mathcal{L}_{sl_1}(\delta(\tilde{P}_q^l), P_q^*, l) + \mathcal{L}_{sl_1}(P_q^l, P_q^*, l), \quad (8)$$

where the first part supervises the assigned meta-points and the second part supervises the refined meta-points towards to GT keypoints P_q^* . For the visibility classification loss, we employ binary cross-entropy loss:

$$\mathcal{L}_{vis} = BCE(\tilde{v}_q, \mathbf{1}_\delta), \quad (9)$$

where $\mathbf{1}_\delta \in \mathbb{R}^M$ is a vector indicating assigned meta-points with 1, otherwise 0. Therefore, our full training objective is calculated as $\mathcal{L}_{full} = \mathcal{L}_{reg} + \alpha \cdot \mathcal{L}_{vis}$, where we reuse the balance hyper-parameter α in cost computation Eqn. 3.

In the inference stage, we employ the result of last decoder layer P_q^L as the estimated keypoints. As for N -shot setting, we average the support keypoint features from different shots as previous works [41, 48]. Besides, we average their cost matrices to obtain better assignment $\delta(\cdot)$, i.e., $C_{\bar{s}} = \frac{1}{N} \sum_n C_{s_n}$. We also average their assigned meta-embeddings, i.e., $\delta(\tilde{E}_{\bar{s}}) = \frac{1}{N} \sum_n \delta(\tilde{E}_{s_n})$. In this way, we could obtain enhanced information to facilitate prediction.

4. Experiments

4.1. Dataset, Metric, and Implementation Details

Overall, we follow the experimental setting of previous work [41]. Specifically, we conduct experiments on MP-100 dataset [48], which covers 100 classes and 8 super-classes and is the largest benchmark dataset for CAPE. MP-100 contains over 18K images and 20K annotations, and the keypoint numbers range from 8 to 68 across different classes. The collected 100 classes are split into non-overlapping train/val/test sets with the ratio of 70 : 10 : 20. There are five random splits to reduce the impact of randomness. We use the Probability of Correct Keypoint (PCK) [51] as the quantitative metric. Apart from the PCK of threshold 0.2 as in [41, 48], we also report the mPCK (of [0.05, 0.1, 0.15, 0.2]) for a more comprehensive evaluation.

Table 1. Results of 1-shot and 5-shot setting on MP-100 dataset. We summarize PCK@0.2 results and report mPCK results in Tab. 2.

Method	1-shot setting						5-shot setting					
	Split1	Split2	Split3	Split4	Split5	Average	Split1	Split2	Split3	Split4	Split5	Average
ProtoNet [42]	46.05	40.84	49.13	43.34	44.54	44.78	60.31	53.51	61.92	58.44	58.61	58.56
MAML [17]	68.14	54.72	64.19	63.24	57.20	61.50	70.03	55.98	63.21	64.79	58.47	62.50
Fine-tune [35]	70.60	57.04	66.06	65.00	59.20	63.58	71.67	57.84	66.76	66.53	60.24	64.61
POMNet [48]	84.23	78.25	78.17	78.68	79.17	79.70	84.72	79.61	78.00	80.38	80.85	80.71
CapeFormer [41]	89.45	84.88	83.59	83.53	85.09	85.31	91.94	88.92	89.40	88.01	88.25	89.30
MetaPoint	83.45	79.88	77.59	76.53	79.09	79.31	84.13	80.30	78.02	77.12	79.58	79.83
MetaPoint+	90.43	85.59	84.52	84.34	85.96	86.17	92.58	89.63	89.98	88.70	89.20	90.02

Table 2. The evaluation results using mPCK metric.

(a) in 1-shot and 5-shot settings.

		Split1	Split2	Split3	Split4	Split5
1-shot	CapeFormer	75.13	69.30	68.59	68.50	71.38
	MetaPoint+	77.11	71.07	70.32	69.93	72.73
5-shot	CapeFormer	78.05	74.43	74.87	73.80	76.11
	MetaPoint+	79.22	75.51	76.20	75.92	77.65

(b) in cross super-category pose estimation.

		Human Body	Human Face	Vehicle	Furniture
CapeFormer		60.21	58.04	27.34	35.60
MetaPoint+		62.42	60.72	30.78	38.52

4.2. Comparison with Prior Works

Following previous works [41], our comparison baselines include ProtoNet [42], MAML [17], Fine-tune [35], POMNet [48], and CapeFormer [41]. Specifically, ProtoNet, MAML, and Fine-tune are popular few-shot learning methods adapted in [48]. POMNet [48] adopts a keypoint matching framework to match support keypoints to the pixels of query image. CapeFormer [41] enhances above similarity modeling and further refines keypoints via transformer modules. We use two versions of our model for comparison, *i.e.*, MetaPoint and MetaPoint+. MetaPoint directly employs the predicted meta-points as results, where the support keypoints are only used to determine the identities for evaluation. MetaPoint+ is the full-fledged version of our model and employs the refined meta-points as results.

We summarize the results of PCK@0.2 in 1-shot setting and 5-shot setting in Tab. 1. Firstly, we could see that our MetaPoint achieves satisfactory performances without any support and refinement, which indicates the mined meta-points could reveal the keypoints inherency and serve as high-quality proposals. Our MetaPoint+ achieves the optimal performances against all baselines, demonstrating the effectiveness of our framework on learning and refining meta-points. We could also see that our methods have relatively robust performances on various dataset splits and different number of support images.

Table 3. Results of cross super-category pose estimation. We summarize the PCK@0.2 results and leave mPCK results in Tab. 2.

Method	Human Body	Human Face	Vehicle	Furniture
ProtoNet	37.61	57.80	28.35	42.64
MAML	51.93	25.72	17.68	20.09
Fine-tune	52.11	25.53	17.46	20.76
POMNet	73.82	79.63	34.92	47.27
CapeFormer	83.44	80.96	45.40	52.49
MetaPoint+	84.32	82.21	46.51	53.67

In addition, the performances of various methods approach to saturation, probably because PCK@0.2 employs a coarse threshold. We also report the results of mPCK (with the thresholds in [0.05, 0.1, 0.15, 0.2]) against the most competitive baseline (*i.e.*, CapeFormer) for a more comprehensive evaluation in Tab. 2. From the perspective of mPCK, we can see that our MetaPoint+ outperforms dramatically against the most competitive baseline, which more clearly demonstrates the effectiveness of our novel framework based on meta-point learning and refining.

4.3. Cross Super-Category Pose Estimation

In the standard setting of CAPE, although the training classes, validation classes, and test classes have no overlap, some classes may have common characteristics and thus easy to transfer, *e.g.*, the body of various kinds of animals. We follow the experiments of cross super-category evaluation in previous works [41, 48] to investigate the generalization ability on significantly different classes of our method. Specifically, we respectively employ one of four super-categories (*i.e.*, human face, human body, vehicle, and furniture) in MP-100 dataset as the test categories, while the other categories are used as the training categories. As shown in Tab. 3, our method outperforms than all baselines, demonstrating that the our method could mine more class-agnostic information via learning meta-points, and thus are more robust against the distribution shift of categories. We also include the evaluation results using mPCK metric in Tab. 2. The performances vary greatly across super-classes, probably because different super-classes have diverse kinds

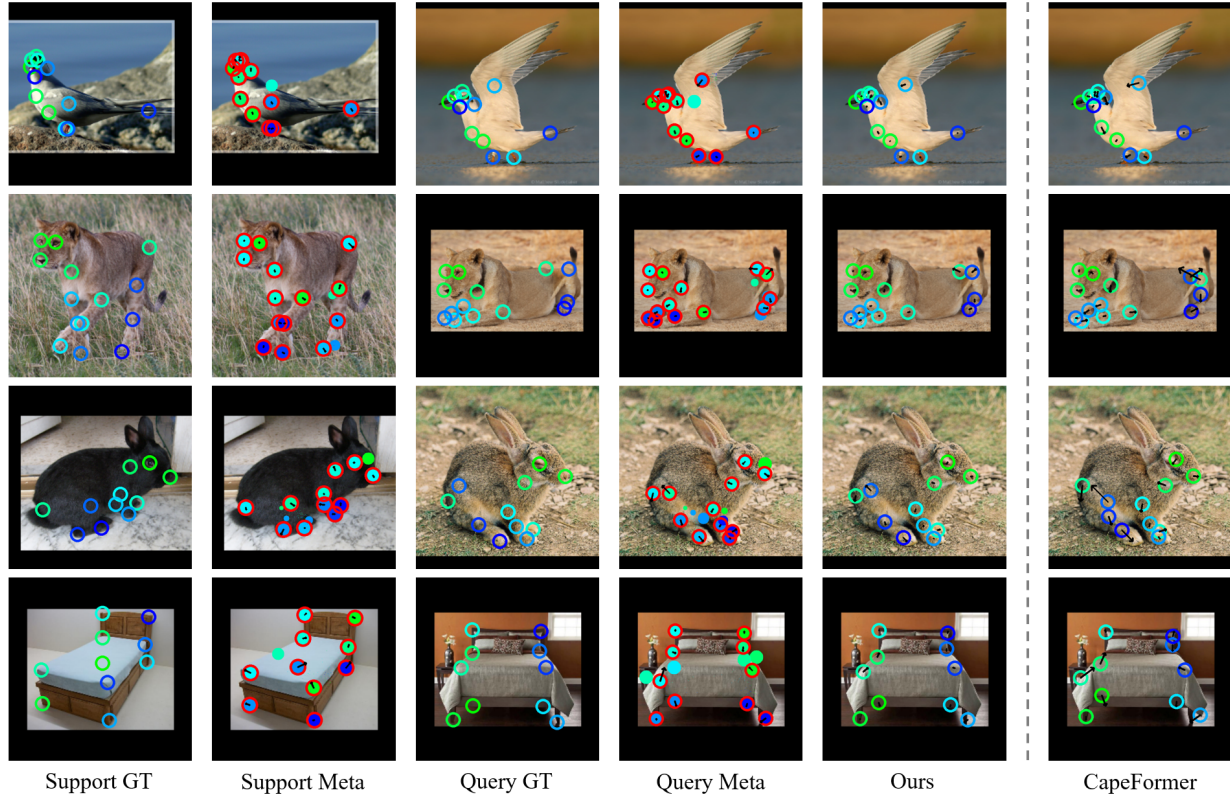


Figure 4. In each row, we show the GT keypoints and predictions of a pair of support image and query image. The left two columns show the GT keypoints and estimated meta-points on support image. The middle two columns show the GT keypoints and estimated meta-points on query image. The right two columns show the final keypoints predicted by ours and CapeFormer. We employ the small black arrows to indicate the deviations to GT. The radii of drew meta-points indicate their visibilities, and the assigned meta-points are encircled with red.

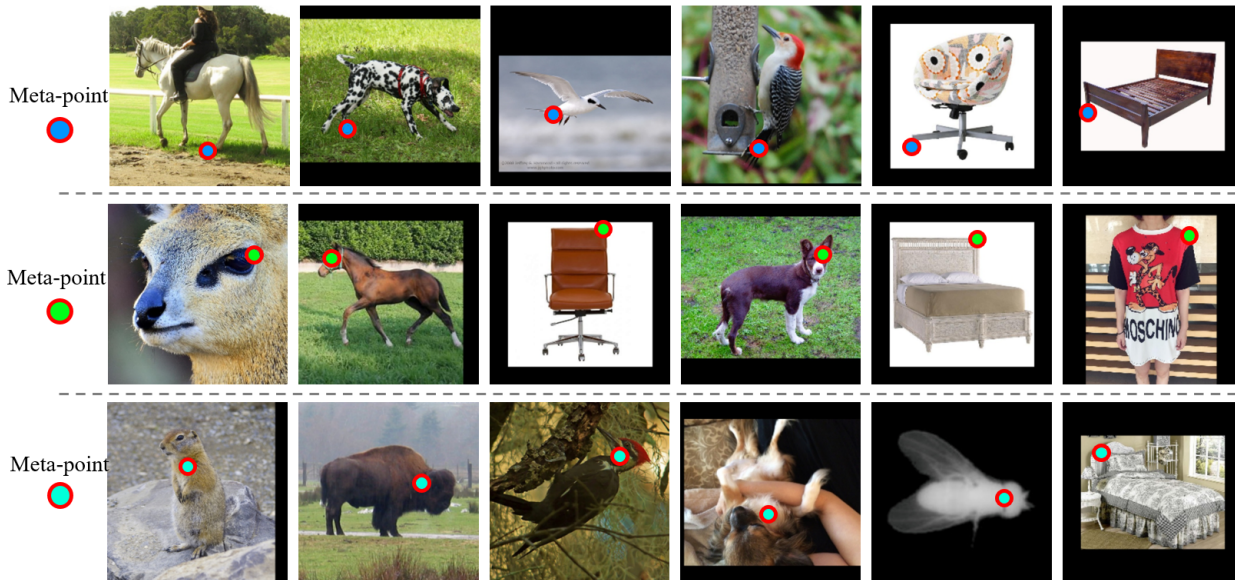


Figure 5. In each row, we track one meta-embedding and visualize its meta-point on various samples if the visibilities are greater than 0.5.

Table 4. Results (mPCK) of various component combinations.

	$+\tilde{v}$	$+F_s$	$+\delta(\tilde{E}_s)$	$+\mathcal{L}_{sl_1}$	Split1
#1	-	-	-	-	68.18
#2	✓	-	-	-	70.37
#3	✓	✓	-	-	74.70
#4	✓	✓	✓	-	76.43
#5	✓	✓	✓	✓	77.11

of keypoints as discussed in [41].

4.4. Qualitative Analyses

For a more intuitive understanding of our method, we visualize the predicted meta-points and keypoints in 1-shot setting for novel classes in Fig. 4. As shown in the second column and the fourth column, our model could predict meaningful meta-points without any support, which have satisfactory performances compared with assigned GT. Furthermore, the meta-points correspond to consistent semantic parts in support images and query images. Therefore, the predicted meta-points not only reveal the inherency of keypoints but also could serve as favourable proposals for subsequent refinement. As shown in the first row, the support keypoints of bird feet are hard to provide clear support information. In such case, previous works may be confused by the ambiguous support, while our method could predict precise keypoints based on the inherent and complementary information mined via meta-points. Similar cases could be found in other visualizations.

To further explore the properties of learned meta-points, we track one meta-embedding and visualize its meta-point on various samples if the visibilities are greater than 0.5 in Fig. 5. As shown in the first row, we could find that the meta-point could correspond to consistent semantic parts of various classes, even if the parts have just high-level or potential relation. Note that there is no annotation binding the points across classes, which demonstrates that our model could mine and extract the inherent and universal information of keypoints. In the second row, the meta-point may correspond to the egoistic left-upper point of objects. Similar phenomena could be found in other cases.

4.5. Ablation Study

In this section, we investigate the performances of component combinations and various configurations of our method. Firstly, we gradually enable various components in our model and report the mPCKs in Tab. 4. The row #1 is a plain version to directly predict and refine meta-points without extra information, which shows the basic performance of learned meta-points. In row #2, we further enable visibility \tilde{v} , which avoids the wrong but accidental assignment to non-existent meta-points (e.g., occluded) and thus

Table 5. Results (mPCK) of various configurations of our model.

	Version	Split1	Split2	Split3	Split4
#1	Standard	77.11	71.07	70.32	69.93
#2	Dual-Asm	73.21	66.90	65.78	65.61
#3	$R = 2$	76.73	70.66	70.02	69.64
#4	$R = 4$	77.18	71.12	70.40	69.97
#5	$L = 2$	76.30	70.22	69.38	69.21
#6	$L = 4$	77.20	71.15	70.44	69.98

prompts the performance. In row #3 and row #4, we enable support features F_s and support meta-embeddings $\delta(\tilde{E}_s)$ in refinement, which provide critical and complementary information for refining meta-points to desired keypoints. Row #5 indicating the contribution of our slacked regression loss \mathcal{L}_{sl_1} , which could slack the supervision on immature predictions. In addition, we explore the impact of various configurations in Tab. 5. Specifically, Row #1 indicates the standard performance of our model. Row #2 solves the assignment $\delta(\cdot)$ based on the sum of cost matrices of query and support images but obtain worse results, probably because only the cost matrix of support image is available in inference. Row #3 and row #4 explore the impact of feature map number R , and our default configuration (i.e., $R = 3$) could mine effective features from pyramid feature maps via our point decoder, but too shallow feature map only improves negligibly (i.e., $R = 4$) probably because the features are too low-level. The last two rows investigate the impact of decoder layer number, and our point decoder could progressively refine keypoints via multiple decoder layers. The performances are gradually saturated, and our default choice (i.e., $L = 3$) is appropriate and enough.

5. Conclusion

In this paper, we have proposed a novel framework for CAPE based on meta-point learning and refining. Such meta-points not only reveal the inherency of keypoints but also serve as meaningful proposals for downstream tasks. We also have proposed a progressive deformable point decoder and a slacked regression loss for better prediction and supervision. The effectiveness of our framework has been demonstrated by in-depth experiments on MP-100 dataset.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 62132006, in part by the Natural Science Foundation of Jiangxi Province of China under Grants 20223AEI91002, 20232BAB202001, and 20224BAB212012, and in part by the Project funded by China Postdoctoral Science Foundation under Grant 2022M721417.

References

- [1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012. 2, 3
- [2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, pages 3686–3693, 2014. 2
- [3] Bjorn Browatzki and Christian Wallraven. 3fabrec: Fast few-shot face alignment by reconstruction. In *CVPR*, pages 6110–6120, 2020. 2
- [4] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. Cross-domain adaptation for animal pose estimation. In *ICCV*, pages 9498–9507, 2019. 2
- [5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020. 4
- [6] Junjie Chen, Li Niu, Liu Liu, and Liqing Zhang. Weak-shot fine-grained classification via similarity transfer. In *NeurIPS*, pages 7306–7318, 2021. 2
- [7] Junjie Chen, Li Niu, and Liqing Zhang. Depth privileged scene recognition via dual attention hallucination. *IEEE Transactions on Image Processing*, 30:9164–9178, 2021. 3
- [8] Junjie Chen, Li Niu, Siyuan Zhou, Jianlou Si, Chen Qian, and Liqing Zhang. Weak-shot semantic segmentation via dual similarity transfer. In *NeurIPS*, pages 32525–32536, 2022. 3
- [9] Junjie Chen, Li Niu, Jianfu Zhang, Jianlou Si, Chen Qian, and Liqing Zhang. Amodal instance segmentation via prior-guided expansion. In *AAAI*, pages 313–321, 2023. 2
- [10] Runnan Chen, Yuexin Ma, Lingjie Liu, Nenglu Chen, Zhiming Cui, Guodong Wei, and Wenping Wang. Semi-supervised anatomical landmark detection via shape-regulated self-training. *Neurocomputing*, 471:335–345, 2022. 2
- [11] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, pages 7103–7112, 2018. 2
- [12] Yinbo Chen, Zhuang Liu, Huijuan Xu, Trevor Darrell, and Xiaolong Wang. Meta-baseline: Exploring simple meta-learning for few-shot learning. In *ICCV*, pages 9062–9071, 2021. 2
- [13] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *CVPR*, pages 5386–5395, 2020. 2
- [14] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. pages 17864–17875, 2021. 4
- [15] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In *CVPR*, pages 8890–8899, 2020. 3
- [16] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *CVPR*, pages 4013–4022, 2020. 2
- [17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017. 6
- [18] Yuying Ge, Ruimao Zhang, and Ping Luo. Metacloth: Learning unseen tasks of dense fashion landmark detection from a few samples. *IEEE Transactions on Image Processing*, 31: 1120–1133, 2021. 2
- [19] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint regression. In *CVPR*, pages 14676–14686, 2021. 2
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3
- [21] Xingzhe He, Gaurav Bharaj, David Ferman, Helge Rhodin, and Pablo Garrido. Few-shot geometry-aware keypoint localization. In *CVPR*, pages 21337–21348, 2023. 2
- [22] Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):814–830, 2015. 2
- [23] Ahmad Humayun, Fuxin Li, and James M Rehg. Rigor: Reusing inference in graph cuts for generating object regions. In *CVPR*, pages 336–343, 2014. 2, 3
- [24] Ayush Jaiswal, Yue Wu, Pradeep Natarajan, and Premkumar Natarajan. Class-agnostic object detection. In *WCAV*, pages 919–928, 2021. 3
- [25] Philipp Krähenbühl and Vladlen Koltun. Geodesic object proposals. In *ECCV*, pages 725–739. Springer, 2014. 2, 3
- [26] Rollyn Labuguen, Jumpei Matsumoto, Salvador Blanco Negrete, Hiroshi Nishimaru, Hisao Nishijo, Masahiko Takada, Yasuhiro Go, Ken-ichi Inoue, and Tomohiro Shibata. Macaquepose: a novel “in the wild” macaque monkey pose dataset for markerless motion capture. *Frontiers in behavioral neuroscience*, 14:581154, 2021. 2
- [27] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *CVPR*, pages 10863–10872, 2019. 2
- [28] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *ICCV*, pages 11025–11034, 2021. 2
- [29] Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu. Cr-net: Cross-reference networks for few-shot segmentation. In *CVPR*, pages 4165–4173, 2020. 2
- [30] Changsheng Lu and Piotr Koniusz. Few-shot keypoint detection with uncertainty learning for unseen species. In *CVPR*, pages 19416–19426, 2022. 2
- [31] Changsheng Lu, Hao Zhu, and Piotr Koniusz. From saliency to dino: Saliency-guided vision transformer for few-shot keypoint detection. *arXiv preprint arXiv:2304.03140*, 2023. 2

- [32] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. 2016. [2](#)
- [33] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In *CVPR*, pages 13264–13273, 2021. [2](#)
- [34] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, Zhibin Wang, and Anton van den Hengel. Poseur: Direct human pose regression with transformers. In *ECCV*, pages 72–88. Springer, 2022. [2](#)
- [35] Akihiro Nakamura and Tatsuya Harada. Revisiting fine-tuning for few-shot learning. *arXiv preprint arXiv:1910.00216*, 2019. [6](#)
- [36] Xuecheng Nie, Jiashi Feng, Jianfeng Zhang, and Shuicheng Yan. Single-stage multi-person pose machines. In *ICCV*, pages 6951–6960, 2019. [2](#)
- [37] Pedro O O Pinheiro, Ronan Collobert, and Piotr Dollár. Learning to segment object candidates. 2015. [2](#), [3](#)
- [38] N Dinesh Reddy, Minh Vo, and Srinivasa G Narasimhan. Carfusion: Combining point tracking and part detection for dynamic 3d reconstruction of vehicles. In *CVPR*, pages 1906–1915, 2018. [2](#)
- [39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. 2015. [3](#)
- [40] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *CVPR*, pages 11069–11078, 2022. [2](#)
- [41] Min Shi, Zihao Huang, Xianzheng Ma, Xiaowei Hu, and Zhiguo Cao. Matching is not enough: A two-stage framework for category-agnostic pose estimation. In *CVPR*, pages 7308–7317, 2023. [1](#), [3](#), [4](#), [5](#), [6](#), [8](#)
- [42] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. 2017. [6](#)
- [43] Xibin Song, Peng Wang, Dingfu Zhou, Rui Zhu, Chenye Guan, Yuchao Dai, Hao Su, Hongdong Li, and Ruigang Yang. ApolloCar3d: A large 3d car instance understanding benchmark for autonomous driving. In *CVPR*, pages 5452–5462, 2019. [2](#)
- [44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2017. [5](#)
- [45] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. [2](#)
- [46] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020. [2](#)
- [47] Zhen Wei, Bingkun Liu, Weinong Wang, and Yu-Wing Tai. Few-shot model adaptation for customized facial landmark detection, segmentation, stylization and shadow removal. *arXiv preprint arXiv:2104.09457*, 2021. [2](#)
- [48] Lumin Xu, Sheng Jin, Wang Zeng, Wentao Liu, Chen Qian, Wanli Ouyang, Ping Luo, and Xiaogang Wang. Pose for everything: Towards category-agnostic pose estimation. In *ECCV*, pages 398–416, 2022. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#)
- [49] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *ECCV*, pages 736–753. Springer, 2022. [3](#)
- [50] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vit-pose: Simple vision transformer baselines for human pose estimation. pages 38571–38584, 2022. [2](#)
- [51] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2878–2890, 2012. [5](#)
- [52] Zihao Yin, Ping Gong, Chunyu Wang, Yizhou Yu, and Yizhou Wang. One-shot medical landmark localization by edge-guided transform and noisy landmark refinement. In *ECCV*, pages 473–489. Springer, 2022. [2](#)
- [53] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *CVPR*, pages 5217–5226, 2019. [2](#)
- [54] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *CVPR*, pages 7093–7102, 2020. [2](#)
- [55] Weilin Zhang and Yu-Xiong Wang. Hallucination improves few-shot object detection. In *CVPR*, pages 13008–13017, 2021. [2](#)
- [56] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. [2](#), [3](#), [5](#)
- [57] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, pages 391–405. Springer, 2014. [2](#), [3](#)