Points in Context for Efficient Person Search

Yingji Zhong[1] Dongkai Wang[1] Yaowei Wang[2] Shiliang Zhang[1]
[1]Peking University, [2]Peng Cheng Laboratory
[1]{zhongyj, dongkai.wang, slzhang.jdl}@pku.edu.cn, [2]wangyw@pcl.ac.cn,

Abstract

Person search uniformly performs person detection and feature extraction. Most of current works extract person features by detecting bounding boxes and cropping person features with RoIPooling or RoIAlign. This two-stage detector framework suffers from degraded efficiency, sensitivity to detection errors, as well as the incapability of capturing contextual cues outside bounding boxes. This paper targets at addressing those issues with a one-stage person search model named Points-in-Context Network (PCNet). Instead of utilizing feature cropping, person features are extracted by adaptively aggregating features at local points throughout the entire feature pyramid. This way allows the feature extractor to explore beneficial contextual cues outside the person foreground, leading to better feature discriminative power and robustness to detection errors. Experiments show that PCNet achieves competitive accuracy and faster inference speed compared with the state-of-the-art. For instance, on the PRW dataset, PCNet achieves 83.7% rank-1 accuracy and a lightweight version of PCNet achieves 81.8% rank-1 accuracy at 22.3 fps speed. Code is availabe at: https://github.com/zhongyingji/PCNet.

1. Introduction

Person search aims at retrieving the query person across raw video frames. Being able to access raw video frames, person search could jointly optimize person detection and re-identification (reid), thus shows better potentials in efficiency and accuracy than traditional person reid task. Most person search works can be summarized into two categories according to their frameworks for detection and reid feature learning, *i.e.*, i) separate model [1, 32, 7], which performs pedestrian detection and reid feature extraction [31, 24, 13, 14, 35, 25, 30, 15, 34, 12, 29, 26] sequentially in two separate models, and ii) joint model [28, 8, 33, 19, 2, 4] which jointly detects person and extracts reid features with one model through end-to-end training. Detailed review to existing works will be given in Sec. 2.

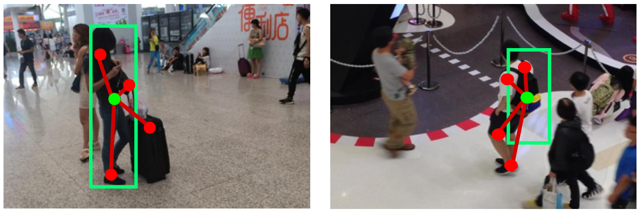


Figure 1. Illustration of the deficiency of extracting reid from bounding boxes. Feature from bounding box suffers from the incapability of capturing contextual cues outside bounding boxes, as well as the sensitivity to detection error. The proposed PCNet addresses the issues by adaptively aggregating the points in context.

The joint model enjoys better capability of jointly optimizing detection and reid feature learning. Most of current joint models are implemented based on two-stage detectors like Faster-RCNN [23]. They apply feature cropping like RoIPooling [10, 23] or RoIAlign [9] with proposals from RPN [23] and couple reid feature extraction with detection in detector head [28]. As illustrated in Fig. 1, extracting reid features from bounding boxes with cropping is incapable of capturing contextual cues, which could be meaningful for differentiating persons [1]. Feature cropping also makes the extracted person features sensitive to detection errors.

This work aims at addressing the above issues in a unified model by exploiting more contextual cues for reid feature extraction. Different from previous works, our idea is to jointly perform person detection and feature extraction based on the one-stage detector [20, 21, 22, 18, 17, 27, 38] without feature cropping to retain more contextual cues. Specifically, we propose a Points-in-Context Network (PCNet) to extract the reid feature by adaptively aggregating informative points throughout the feature pyramid. This way effectively identifies meaningful local points and fuses contextual cues on those points as the reid feature. Inheriting the faster speed of one-stage detector, PCNet shows superior inference speed compared to previous works.

PCNet considers informative points across multiple levels of the feature pyramid for context aggregation. As illustrated in Fig. 2, we firstly introduce the Contextual Point

Pooling (CPP) module which adaptively aggregates the informative points within each level. For each point in a certain level, CPP firstly predicts a set of offsets that indicates the surrounding points, then samples corresponding features to enhance this point. To encourage those surrounding points to capture more contextual cues, we introduce a distance loss to scatter them on the person foreground, i.e., avoid gathering those points in a local region.

CPP performs context aggregation on each level of the pyramid for feature extraction, thus cannot fully utilize the context from other levels. As different levels of pyramid present discriminative power for varied scales, solely considering a single scale limits the discriminative power of CPP features. For instance, features from the top levels could be more discriminative to semantics, while those from lower levels are stronger in differentiating local details. To enhance features extracted by CPP, we further propose the Cross Scale Aggregation (CSA) module to aggregate points across pyramid levels. With CSA module, the reid feature is capable of fusing multi-scale cues including both high level semantics and low level details, thus could be more discriminative.

We test the PCNet on wildly used person search benchmarks, *i.e.*, CUHK-SYSU [28], PRW [36], and LSPS [37], respectively. Experimental results show that, our PCNet achieves competitive performance compared with recent works. For instance, PCNet achieves the rank-1 accuracy of 83.7% on the PRW, outperforming the recent APNet [37] by 2.3%. Moreover, PCNet exhibits superior inference speed, *e.g.*, a lightweight version of PCNet achieves 81.8% rank-1 accuracy on PRW, outperforming APNet in both accuracy and speed, *i.e.*, 22.3 *vs.* 13 fps.

From extensive experiments, we could conclude that PCNet exploits more spatial contexts and enjoys fast inference and better robustness to detection errors. Our contribution can be summarized as follows: i) To the best of our knowledge, PCNet is an early person search method implemented on the one-stage detector, which is free of feature cropping. This property allows PCNet to capture more context information and gain better robustness to detection errors; ii) The Contextual Point Pooling (CPP) and Cross Scale Aggregation (CSA) modules are further proposed to aggregate meaningful cues across the feature pyramid for reid feature extraction; iii) PCNet achieves competitive performance on current benchmarks, *i.e.*, CUHK-SYSU, PRW, and LSPS, while exhibiting faster inference speed.

2. Related Work

Person search is closely related to object detection and person search. Recent works on those two topics will be briefly introduced in this section.

Object Detection. Current detectors can be briefly summarized into two-stage detectors and one-stage detectors, respectively. Two-stage detectors [23, 9] firstly use region proposal network (RPN) [23] to generate proposals. The proposals are applied for feature cropping like RoIPooling [10, 23] or RoIAlign [9]. Cropped features are then used to refine the proposals by the detector head. One-stage detectors [20, 21, 22, 18, 17, 27, 38] do not generate intermediate proposals. They detect bounding boxes by regressing each pixel on the feature map. Therefore, no feature cropping operation is applied in one-stage detector. Except the yolo-based detectors [20, 21, 22], most one-stage detectors are built up with the feature pyramid to cover various possible scales of objects. One-stage detector is generally superior in the utilization of spatial contexts as well as the inference speed. This is mainly because it discards the feature cropping operation, thus is not affected by the proposal-wise computations.

Person Search. As mentioned before, relevant works can be categorized into separate models and joint models, respectively. Separate model [1, 32, 7] decomposes person search into two sequential tasks, i.e., detection and reid, and applies two models for those two tasks, respectively. It first crops detected person images from input frame, then performs feature extraction with traditional reid methods [31, 24, 13, 14, 35, 25, 30, 15, 12, 29, 26]. Chen *et al.* [1] offlinely segment the input frame to enable the reid model to focus on the foreground. Lan *et al.* [32] fuse the multi-level features from reid network to solve the resolution variance issue. Besides, Han *et al.* [7] finetune the detection model driven by the loss from reid model, aiming to refine the bounding boxes for better feature extraction.

Joint model [28, 8, 33, 19, 37, 2, 4, 5] detects bounding boxes and extracts reid feature within one model, thus is capable of jointly optimizing two tasks. Current works are based on the two-stage detector like Faster-RCNN. Given the feature generated by RoIPooling, Xiao *et al.* [28] extract reid feature by detector head, and integrate the reid task into the detection framework. Liu *et al.* [8] utilize more context cues by shrinking the region containing the query. However, the reid feature is still extracted with RoI pooling. Zhong *et al.* [37] alleviate the misalignment issue caused by occlusion or limited camera views by estimating the holistic boxes to align the feature. Dong *et al.* [4] utilize a siamese network during training, and impose the feature similarity on both branches, which enforces the model to discriminate identities based on appearance rather than context. Chen *et al.* [2] disentangle the reid feature into norm and angle to alleviate the contradiction of detection and reid tasks.

Another line of joint model works focus on the similarity calculation by forwarding each query-gallery pair into the network. Munjal *et al.* [19] utilize a query-guided method to guide the proposal generation and compute the similarity based on RoI pooled features. Dong *et al.* [5] apply a similar mechanism. Those methods generally achieve better

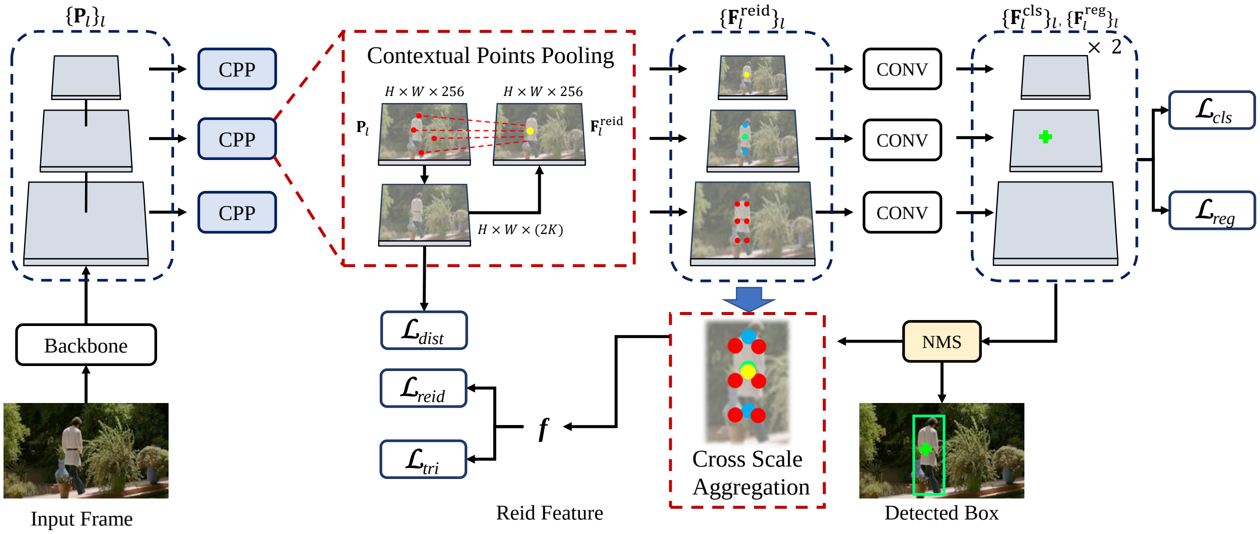


Figure 2. Architecture of the proposed Points-in-Context Network (PCNet). We show the pyramid of three levels to simplify the illustration. PCNet outputs three sets of pyramid, *i.e.*, $\{\mathbf{F}_l^{\text{reg}}\}$, $\{\mathbf{F}_l^{\text{cls}}\}$ and $\{\mathbf{F}_l^{\text{reid}}\}$, with l from 3 to 7 as in RetinaNet [17]. Context Points Pooling (CPP) gathers informative points for each position. Cross Scale Aggregation (CSA) aggregates points of across different levels on the pyramid. $\mathcal{L}$ refers to the proposed loss during training. The green cross refers to positive sample when training. During inference, the green cross refers to the position which survives NMS. Note that PCNet enjoys the box-free property since no feature cropping operations are applied. Best viewed on screen.

accuracy, but require heavy computations during inference.

Current joint models of person search apply two-stage detector like Faster-RCNN for feature extraction [28, 8, 37, 2, 4] or similarity calculation [33, 19, 5]. Our proposed PCNet makes it possible to leverage one-stage detector in person search which is capable of alleviating the issues of limited context and sensitivity to detection error as shown in Fig. 1. To the best of our knowledge, this is the first work discarding the usage of feature cropping while instead aggregating points from context for reid feature extraction. We will show in the experiment that PCNet exhibits competitive performance and faster inference speed.

3. Overview

Given a query person image q, person search targets at retrieving the person with same identity from n raw video frames $\{V_i\}_{i=1}^n$. We focus on the joint person search model. For the i_{th} frame, the model requires to detect a set of bounding boxes $\{B_j^i\}$ and extract their corresponding reid features $\{f_j^i\}$. We omit the superscript i and subscript j in the following part the simplify the representation.

To utilize the contextual cues and get rid of feature cropping, our PCNet is implemented on the one-stage detector. As shown in Fig. 2, the backbone generates the feature pyramid $\{\mathbf{P}_l\}$, where l=3:7 for RetinaNet. By processing each level of the pyramid individually, the detector head generates the feature pyramid $\{\mathbf{F}_l^{\text{reid}}\}$ for reid feature extraction, $\{\mathbf{F}_l^{\text{reg}}\}$ and $\{\mathbf{F}_l^{\text{cls}}\}$ for person detection, respectively.

As mentioned in [1], joint model confronts the contradiction of learning commonness and differences for detection and reid, respectively. To alleviate this issue, we design the detector head which firstly predicts the reid feature, then the detection feature. Based on the feature pyramid $\{\mathbf{P}_l\}$, we first compute $\{\mathbf{F}_l^{\text{reid}}\}$ as,

$$\mathbf{F}_l^{\text{reid}} = \text{CPP}(\mathbf{P}_l), \quad (1)$$

where $\text{CPP}(\cdot)$ denotes Contextual Points Pooling, which converts feature map from the pyramid into the reid feature map. As shown in Fig. 1, contextual cues outside person foreground helps to differentiate persons. To improve the discriminative power of $\mathbf{F}^{\text{reid}}$, CPP enhances it with both person foreground cues and informative contexts. More details of CPP will be given in Sec. 3.1.

$\{\mathbf{F}_l^{\text{reg}}\}$ and $\{\mathbf{F}_l^{\text{cls}}\}$ are then predicted based on $\{\mathbf{F}_l^{\text{reid}}\}$ to gain the discriminative power on persons and backgrounds, *i.e.*,

$$(\mathbf{F}_l^{\text{reg}}, \mathbf{F}_l^{\text{cls}}) = \text{conv}(\mathbf{F}_l^{\text{reid}}), \quad (2)$$

where $\text{conv}(\cdot)$ is consecutive convolution applied in RetinaNet [17]. Note that, $\{\mathbf{F}_l^{\text{reid}}\}$ contains more discriminative cues, thus could be processed by Eq. (2) to differentiate persons and backgrounds. The effectiveness of the detector head is validated in Sec. 4.3.

Following RetinaNet, we use non-maximum suppression (NMS) on $\{\mathbf{F}_l^{\text{reg}}\}$ and $\{\mathbf{F}_l^{\text{cls}}\}$ to get the final person detection result $\mathbf{R}$, which includes pyramid level l the person is detected from, its spatial location $\mathbf{s} = (x, y)$ on the feature map, as well as the regressed bounding box, *i.e.*,

$$\mathbf{R}_p = (l_p, \mathbf{s}_p, B_p) = \mathrm{NMS}(\{\mathbf{F}_l^{\mathrm{reg}}\}, \{\mathbf{F}_l^{\mathrm{cls}}\}), \tag{3}$$

where p is the index of p-th person in input image. During training, $\mathbf{R}_p$ can be acquired from the groundtruth. To simplify the notation, we omit the subscript p in following parts.

Instead of feature cropping, we generate the reid feature f based on $\mathbf{R}$ and $\{\mathbf{F}_l^{\mathrm{reid}}\}$ using the Cross Scale Aggregation (CSA). CSA aggregates features across levels of the feature pyramid, *i.e.*,

$$f = \mathrm{CSA}(\mathbf{R}, \{\mathbf{F}_l^{\mathrm{reid}}\}), \tag{4}$$

where f denotes the feature for reid. Extracted with CPP and CSA, f aggregates informative points from the feature pyramid across different pyramid levels. The following parts present details of CPP, CSA, and our training objectives.

3.1. Contextual Points Pooling

The feature map $\{\mathbf{F}_l^{\mathrm{reid}}\}$ is applied for reid feature extraction and detection in Eq. (4) and Eq. (2). To enhance its discriminative power, CPP embeds $\{\mathbf{F}_l^{\mathrm{reid}}\}$ with more contextual cues. Moreover, it is likely for the context to imply the box boundary which benefits the following detection task.

CPP individually processes each level of the feature pyramid $\{\mathbf{P}_l\}$ to generate the $\{\mathbf{F}_l^{\mathrm{reid}}\}$, *e.g.*, CPP generates $\mathbf{F}_l^{\mathrm{reid}}$ from $\mathbf{P}_l$. Specifically, for a location $\mathbf{s} = (x, y)$ on the l-th level of $\{\mathbf{F}_l^{\mathrm{reid}}\}$, *i.e.*, the $\mathbf{F}_l^{\mathrm{reid}}(\mathbf{s})$, CPP computes it by aggregating contextual points surrounding the location of $\mathbf{s}$ on the $\mathbf{P}_l$. We denote the computation of CPP as,

$$\mathbf{F}_l^{\mathrm{reid}}(\mathbf{s}) = \sum_{k=1}^{K} w_{k,l} \cdot \mathbf{P}_l(\mathbf{s} + \mathbf{o}_k), \tag{5}$$

where $\{\mathbf{o}_k\}_{k=1}^{K}$ denotes the offsets and K is the number of considered contextual points. $\{\mathbf{s} + \mathbf{o}_k\}_{k=1}^{K}$ locate the points for aggregation. $\{w_{k,l}\}_{k=1}^{K}$ refers to aggregation weights of the l-th pyramid.

Eq. (5) can be implemented with deformable convolution [3], which contains both weights $\{w_{k,l}\}_{k=1}^{K}$ and offsets $\{\mathbf{o}_k\}_{k=1}^{K}$. However, our experiments show that, simply applying deformable convolution leads to small offset values. In other words, contextual points of Eq. (5) tend to fall inside a local area of $\mathbf{s}$. This runs counter to our goal, *i.e.*, utilizing the context to the largest extent.

To involve more contextual cues, we propose a distance loss $\mathcal{S}$ for training, which scatters the points to match the body prior. Suppose the location $\mathbf{s}^*$ is a positive sample during training, *e.g.*, assigned with a groundtruth bounding box of height h^*. We split its K points with locations $\{\mathbf{s}^* + \mathbf{o}_k\}_{k=1}^{K}$ into M sets, where each set contains $\lfloor K/M \rfloor$ points depicting a specific body region. The distance loss $\mathcal{S}$ is computed to keep apart these sets. We implement this constraint by ensuring the vertical distances of the mass center between adjacent sets, *i.e.*,

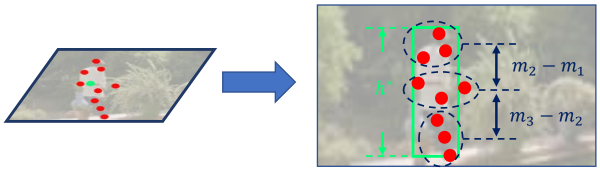


Figure 3. Illustration of the distance loss Eq. (6). Number of points K and splitted sets M are set to 9 and 3, respectively. The green point denotes a positive sample during training. Its contextual points are mapped back to the input image. m refers to vertical dimension of the mass point of each set.

$$\mathcal{S}(\mathbf{s}^*) = \frac{1}{h^*} \sum_{i=1}^{M-1} ((m_{i+1} - m_i) - \frac{h^*}{M})^2, \tag{6}$$

where m is the vertical dimension of the mass center of each set. Fig. 3 illustrates the computation to Eq. (6).

Eq. (6) prevents the points from gathering in a local region, thus is helpful to involve more contextual cues. Note that, we only compute $\mathcal{S}(\mathbf{s}^*)$ on locations assigned as positive throughout the entire feature pyramid, which is identified following the strategy of RetinaNet [17].The effectiveness of the distance loss will be validated in experiments.

3.2. Cross Scale Aggregation

With learned $\{\mathbf{F}_l^{\mathrm{reid}}\}$ and detection result $\mathbf{R}$, an intuitive way to obtain reid feature is extracting point feature from the corresponding location of a pyramid level, *i.e.*,

$$f' = \{\mathbf{F}_l^{\mathrm{reid}}\}(\mathbf{s}, l), \tag{7}$$

where the location $\mathbf{s}$ and pyramid level l are included in the detection result $\mathbf{R}$.

Eq. (7) only considers feature at a specific level of the pyramid. As different levels of pyramid present discriminative power for varied scales, solely considering a single scale limits the discriminative power of learned f. For instance, f from the lower level of pyramid might be not discriminative to high level semantics. We thus propose a Cross Scale Aggregation (CSA) module to fuses multi-scale cues on different pyramid levels, *e.g.*, aggregate more points across the feature pyramid of $\{\mathbf{F}_l^{\mathrm{reid}}\}$.

Suppose the location $\mathbf{s}$ in the l-th pyramid survives the NMS, and generates the detection result $\mathbf{R} = (l, \mathbf{s}, B)$. To enhance the reid feature in Eq. (7), CSA firstly selects candidate points from $\{\mathbf{F}_l^{\mathrm{reid}}\}$. We illustrate the procedure of candidate points selection in Fig. 4, where we map the points on feature pyramid back to the original image. As shown in Fig. 4, feature points falling within the detected bounding box B are selected as candidate points, which composes a candidate set $\Omega = \{\mathbf{s}_i, l_i\}$, where $\mathbf{s}_i, l_i$ denotes the location of a point, and its corresponding pyramid level.

Each feature in Ω depicts the detection person from different aspects, *e.g.*, different scale or body parts. CSA aggregates points in Ω with linear combination as the final reid feature f, *i.e.*,

$$f = f' + \sum_{\{\mathbf{s}_i, l_i\} \in \Omega} \mathrm{W}(\bar{f}_i, f') \times \bar{f}_i, \tag{8}$$

$$\bar{f}_i = \{\mathbf{F}_l^{\mathrm{reid}}\}(\mathbf{s}_i, l_i), \tag{9}$$

where $\{\mathbf{s}_i, l_i\}$ is one of the candidate points, and $\bar{f}_i$ denotes its feature vector on $\{\mathbf{F}_l^{\mathrm{reid}}\}$, $\mathrm{W}(\cdot)$ computes the aggregation weight. It is calculated by:

$$\mathrm{W}(\bar{f}_i, f') = e^{\kappa(\bar{f}_i, f')} / \sum_j e^{\kappa(\bar{f}_j, f')}, \tag{10}$$

where $\kappa(\bar{f}_i, f') = \phi(\bar{f}_i)^T \psi(f')$ is the relation function, where both ϕ and ψ are implemented by fully connected layer.

CSA aggregates point features across the pyramid into the final reid feature f. Our method thus outputs a set of tuples. Each tuple can be denoted as $\{f_p, B_p\}$, where p is the index of detected person, f_p and B_p are the corresponding reid feature and detection box.

f embeds multi-scale cues, and enjoys stronger discriminative power than individual features in $\{\mathbf{F}_l^{\mathrm{reid}}\}$. We hence could utilize f to enhance other features in $\{\mathbf{F}_l^{\mathrm{reid}}\}$. To implement this, we propose a variant of the triplet loss, which considers f and features in $\{\mathbf{F}_l^{\mathrm{reid}}\}$ for distance computation.

During training, for a location $\mathbf{s}^*$ assigned as positive and its corresponding person identity id^*, the triplet loss on $\mathbf{s}^*$ can be computed as

$$\mathcal{T}(\mathbf{s}^*) = \max(\sigma + ||f - f'||^2 - \min_j ||f - f'_j||^2, 0), \tag{11}$$

where f and f' are computed with Eq. (8) and Eq. (7) respectively on location $\mathbf{s}^*$. f'_j refers to a feature in the training batch with identity differs with id^*, which also comes from Eq. (7). The threshold value σ is set to 0.5 by default. Eq. (11) makes f' get closer to f. It thus optimizes features in $\{\mathbf{F}_l^{\mathrm{reid}}\}$ to gain better discriminative power to multi-scale cues, and in turn optimizes the $\{\mathbf{P}_l\}$ with back propagation.

Discussions. The candidate set of CSA relies on bounding box. However, CSA is different with the commonly used feature cropping. Feature cropping is implemented with the simple average or max pooling operation, while CSA aggregates the points selectively according to Eq. (10). Moreover, candidate set is not necessary for CSA if memory and computation allow, *e.g.*, aggregating all points throughout the feature pyramid weighted by Eq. (10). In this case, no bounding box is required.

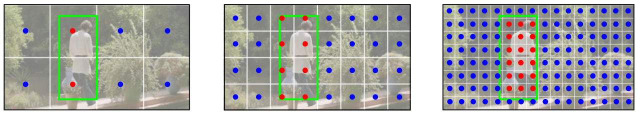


Figure 4. Illustration of candidate points selection (3 levels of feature pyramid are shown for simplicity). Green box refers to the detected box after NMS. We map each location on feature pyramid back to the original image. Points falling within the box are marked with red and are selected into the candidate set Ω.

Compared with feature cropping, CSA generates reid feature and is superior in following aspects: i) Most works apply feature cropping on output of conv4 and the following conv5, and can not utilize contexts outside bounding boxes. CSA can utilize more information since context is retained in RetinaNet. ii) Feature cropping relies on the heavy conv5 layer to extract discriminative reid feature. CSA extracts features with the light linear combination of Eq. (8), which is more efficient. Experiments will show the superiority of CSA over feature cropping in one-stage detector.

3.3. Training Objective

During training, each location across the pyramid is assigned as positive or negative. We adopt the identical strategy of RetinaNet [17] for the positive/negative assignment. Each location is binded to an anchor of a specific scale or an aspect ratio. A location is assigned as positive only if the overlap between its binded anchor and groundtruth boxes exceeds a threshold. For those locations assigned as positive, they are assigned with groundtruth labels including locations on the pyramid $\mathbf{s}^*$, regression targets (x^*, y^*, w^*, h^*) and person identity id^*. The following part introduces our training objective.

The detection loss of PCNet follows RetinaNet, *i.e.*, smooth l_1 loss [6] for regression $\mathcal{L}_{reg}$ and focal loss [17] $\mathcal{L}_{cls}$ for classification. We refer readers to [17] for detailed parameter settings and the computation.

Our training considers the distance loss of Eq. (6) and triplet loss of Eq. (11) to train the CPP and CSA, respectively. We can calculate $\mathcal{L}_{dist}$ and $\mathcal{L}_{tri}$ as:

$$\mathcal{L}_{dist} = \sum_k \mathcal{S}(\mathbf{s}_k^*), \ \mathcal{L}_{tri} = \sum_k \mathcal{T}(\mathbf{s}_k^*). \tag{12}$$

OIM loss [28] is applied for reid feature supervision. The loss is posted on the reid feature computed by Eq. (8), *i.e.*,

$$\mathcal{L}_{reid} = \sum_k \mathrm{OIM}(f_k, \mathrm{id}_k^*). \tag{13}$$

Based on above loss functions, the overall training objective of PCNet thus be formulated as:

$$\mathcal{L} = \mathcal{L}_{cls} + \frac{1}{N} \cdot (\mathcal{L}_{reg} + \mathcal{L}_{reid} + \mathcal{L}_{dist} + \mathcal{L}_{tri}), \tag{14}$$

where N represents number of positive samples.

4. Experiments

4.1. Datasets

CUHK-SYSU [28] consists of 5,694 frames from movie snapshots and 12,490 frames from street snap. The dataset provides 23,430 labeled bounding boxes of 8,432 labeled identites. 96,143 bounding boxes are provided in total. 11,206 frames with 5,532 identities compose the training set, while the rest composes the test set. During inference, different gallery sizes are defined for each query. We use the gallery size of 100 in the following by default.

PRW [36] is captured by 6 cameras deployed at different locations in a campus. The dataset contains 11,816 frames, with 932 labeled identites in total. 5,704 frames are selected as training set with 482 identities, the rest 6,112 frames make up the test set. The dataset annotates 34,304 bounding boxes with identity.

LSPS [37] is captured by 15 cameras deployed at various locations in campus. All bounding boxes are detected from Faster-RCNN [23]. A total of 51,836 frames are collected, among which 60,433 bounding boxes of 4,067 identites are annotated. Training set consists of 18,163 frames of 1,041 identites while the rest makes up the test set. During testing, 2 pseudo camera views are considered, which are cropped views from 2 of the 15 cameras.

Evaluation Metrics. Following the common practice in [28, 8, 33, 19, 37, 2, 4, 5], rank-1 accuracy and mean Average Precision (mAP) are adopted as evaluation metrics. The retrieved bounding box is considered as true positive only if it is the identical person with query and its IoU with groundtruth box is larger than 0.5.

4.2. Implementation Details

Configuration of RetinaNet is almost retained. Each position is assigned with an anchor of aspect ratio 2. Number of aggregated points K in CPP is set to 9. Number of sets M in distance loss is set to 3. Altering K and M does not influence the performance much. All experiments are based on the reid feature of 256-dim and ResNet50 [11] backbone. The dimension of feature map in detector head is set to 256. Input frames are resized to a maximum size of 1333× 800 without changing the aspect ratio. The model is optimized by SGD with a batchsize of 4. For CUHK-SYSU, model is trained for 60k iterations with an initial learning rate of 5e-5, which is decayed by 10 at 50k iteration. On PRW and LSPS, model is trained with an initial learning rate of 1e-4. On PRW, the training lasts 70k iterations and the learning rate is decayed by 10 at 60k iteration. On LSPS, model is trained for 120k iterations, the learning rate is decayed by 10 at 80k and 100k iteration, respectively.

CPP	$\mathcal{L}_{dist}$	CSA	$\mathcal{L}_{tri}$	mAP(%)	rank-1(%)
✓				26.4	68.2
✓	✓			32.9	72.8
✓	✓	✓		37.8	78.8
✓	✓	✓	✓	38.0	80.6
3×3 conv				21.6	59.7
		✓	✓	26.9	68.4

Table 1. Effectiveness of each component. 3×3 conv replaces CPP with normal convolution.

order of detector head		mAP(%)	rank-1(%)
1_{st} task	2_{nd} task		
reid	det based on $\mathbf{F}^{\text{reid}}$	**38.0**	**80.6**
detection	reid based on $\mathbf{F}^{\text{reg}}$	28.7	66.6
	reid based on $\mathbf{F}^{\text{cls}}$	31.2	70.7

Table 2. Analysis on the order of reid and detection tasks in detector head.

one-stage detector	box-free	mAP(%)	rank-1(%)
parallel reid branch	✓	15.7	40.2
parallel reid branch‡	✓	30.4	70.9
RoIAlign		25.2	64.8
PCNet‡	✓	**32.9**	**72.8**

Table 3. Different forms of joint person search model based on one-stage detector RetinaNet [17]. Models with ‡ utilize CPP for context aggregation and $\mathcal{L}_{dist}$ loss. Models with box-free property do not apply feature cropping.

4.3. Ablation Study

Components analysis. We analyze the effectiveness of each proposed components. The results are shown in Table 1. Based on the detector head with a serialized order, aggregating the points by CPP achieves the rank-1 accuracy of 68.2%. By scattering the points to prevent the points from gathering in a local region, $\mathcal{L}_{dist}$ brings a 6.5% mAP improvement. This demonstrates that the way of sampling the points is of great importance for enhancing the feature. CSA aggregates points across the feature pyramid. It remarkably enhances the rank-1 performance by 6.0%, which validates the necessity to integrate multi-scale information for reid. We also evaluate the proposed triplet loss in Eq. (11), and it enhances the rank-1 by 1.8%, which validates the effectiveness of the triplet loss on feature learning.

Analysis on CPP. Eq. (5) shows that CPP adaptively aggregates the informative points for each position. To show its effectiveness, we replace it with the vanilla 3×3 convolution and the results are shown in the lower block of Table 1. The vanilla convolution only achieves the rank-1 of 59.7%, drastically degrades the performance of CPP. The results show that, it is likely for the informative points to distribute out of rigid 3×3 grid. Thus, the vanilla convolution tends to overlook the informative context. Fig. 5 illustrates the learned points of CPP for each detected box. Note that context outside box is auxiliary to appearance cues on

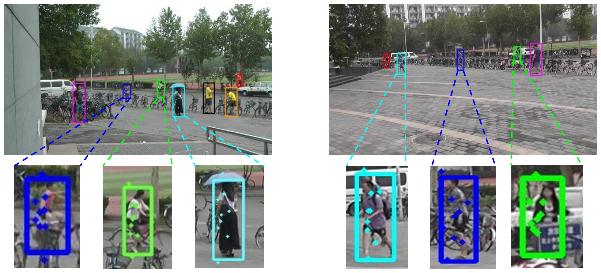


Figure 5. Examples of the sampled points of CPP for the detected boxes. Best viewed on screen and zoomed in.

foregrounds, thus most points are inside box. Certain discriminative context outside boxes, *e.g.*, accessaries like umbrella and bike are also captured.

Order of detector head. As mentioned in Sec. 3, the detector head handles reid task and detection tasks in a serialized manner. It firstly predicts reid feature map, based on which detection maps are predicted afterwards. We compare different order of the two tasks. As shown in Table 2, another prediction order is considered. Detection maps are firstly predicted, *e.g.*, $\mathbf{F}^{\text{cls}}$ and $\mathbf{F}^{\text{reg}}$. Table 2 shows that, generating $\mathbf{F}^{\text{reid}}$ based on either of them results in a sharp performance decrease. For instance, the rank-1 accuracy drops by more than 10% if reid feature is generated from $\mathbf{F}^{\text{reg}}$. Since the detection feature maps have discarded the information to differentiate persons, *e.g.*, only retaining the commonness, it is hard for the following part to recover the lost information. Hence, we can draw the conclusion that commenness can be decoded from differences while it is not the case for the reverse order.

Fig. 6 illustrates that though features $\mathbf{F}^{\text{reid}}$ encode the differences of persons, they can indicate background, *e.g.*, it is likely for those with small activations to be background. Thus, in this order, the learned $\mathbf{F}^{\text{reid}}$ actually benefits detection task.

Discussion. Current methods can not apply the similar scheme of our detector head, *i.e.*, the feature for detecting boxes are decoded from reid feature. They never know the reid feature before they get the boxes. Thus, they must detect first. PCNet is based on one-stage detector, each point in feature map is a potential detected person, thus we can get their reid feature before detection.

One-stage detector for person search. To the best of our knowledge, no joint models based on one-stage detector are proposed. In Table 3, we compare our PCNet with different models based on RetinaNet. The simplest way is to add another parallel branch for reid feature extraction, which is commonly adopted in person search model based on two-stage detector [28, 2]. This naive method only achieves 40.2% rank-1 accuracy. We think the failure is due to the fact that, reid feature extraction is agnostic of the region of person. In contrast, with CPP and the $\mathcal{L}_{dist}$, the parallel reid branch achieves 70.9% accuracy. The combination of CPP and $\mathcal{L}_{dist}$ ensures that each position in the feature map is aware of the rough region of person if it exists. However, the parallel model is inferior to the serialized detector head in Sec.3 by 2.5% in mAP, validating the superiority of serialization in solving the contradiction issue in the box-free model.

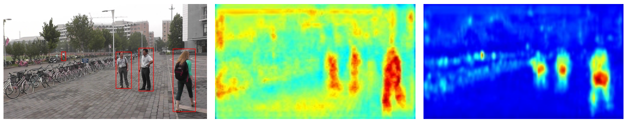
We also evaluate a variant of RetinaNet which applies RoIAlign with the detected boxes on each level of the feature pyramid followed by 3×3 convolution, and sum them up as reid feature. Though the feature cropping is applied at last and context information is kept to the largest extent, it only achieves the degraded 64.8% rank-1. We attribute the failure to the incapbility of 3× 3 convolution. Most person search works use the heavy conv5 [11] to extract reid feature, the 3× 3 convolution is incapable of discriminating the informative parts within the rigid box, *e.g.*, failing to filter out the noises.

	Mem (G)	Params (M)	Speed (fps)	mAP (%)	rank-1 (%)
OIM [28]	2.3	32.3	14.8	33.4	75.2
NAE+ [2]	2.5	32.0	10.5	**41.9**	78.8
APNet [37]	2.5	64.8	13.0	40.6	80.6
PCNet	**1.6**	**30.3**	**22.3**	40.7	**81.8**

Table 4. Comparison with efficient joint models for person search with maximum input of 1333×800. 'Mem' refers to memory consumption during inference. Both memory and speed are measured on a single V100 GPU with batchsize 1. More thorough comparison is referred to supplementary material.

Efficiency. We compare the overhead of PCNet with other efficient models in Table 4. Note that, the methods listed in Table 4 are faster than current works since they are joint models. In this part for our PCNet, we reduce the dimension of feature map in detector head to 64. Compared with the commonly used baseline OIM [28], PCNet outperforms it by 6.6% in rank-1 accuracy with less parameters. PCNet also outperforms APNet [37] with only half of its parameters and faster inference speed, *i.e.*, 22.3 fps *v.s.* 12.0 fps. We further compare PCNet with NAE+ [2]. With significant faster speed, *i.e.*, 22.3 fps *v.s.* 10.5 fps, PCNet exceeds NAE+ by 3.0% in rank-1 accuracy.

Performance upper bound. To verify whether the extracted reid feature is discriminative, we replace the detected boxes with the groundtruth boxes and evaluate the person search performance. On PRW, PCNet achieves the upper bound of 47.1% mAP and 86.1% rank-1 accuracy, which is an outstanding result compared with the upper bound rank-1 of 83.4% in BINet [4]. This shows that, the model can discriminate persons effectively with the proposed CPP and CSA which utilize context and multi-scale information.



(a) (b) (c)

Figure 6. Visualization of: (a) Detection result of an input frame; (b) Activation of the reid feature map; (c) Activation of the detection feature map. Best viewed on screen.

4.4. Comparison with recent works

We compare PCNet with recent works on CUHK-SYSU, PRW and LSPS. The results are shown in Table 5 and Table 6. In this part, we additionally assign multiple anchors for each position of PCNet like RetinaNet [17]. Each position is assigned with anchors of aspect ratios of {1:1, 2:1} and 2 sub-octave scales ($2^{k/3}$ for k≤2). Thus, 4 anchors are assigned for each position. We denote the model as PCNet†.

In Table 5, the first block lists the separete models which require two models for detection and reid, respectively. The second block lists the methods which require $\mathcal{O}(N^2)$ cost during inference, *e.g.*, forwarding each query-gallery pair through the network. The last block exhibits joint person serach methods. Due to the squared computation cost, methods in the second block is the slowest during inference.

PRW. PCNet achieves the competitive rank-1 accuracy of 83.1%, outperforming most previous works. Compared with the methods that require $\mathcal{O}(N^2)$ computation cost during inference, our model also exhibits competitive performance, *e.g.*, compared to IGPN [5] with 256-dim feature, PCNet† shows a superior mAP of 45.5% and rank-1 accuracy of 83.7% . Compared with the joint models, PCNet achieves highest rank-1, surpassing BINet [4] by 1.4%. The lower mAP of PCNet is due to the lower recall of one-stage detector, which can be remedied by muliple anchors. Moreover, BINet is based on OIM, which is slower than our PCNet as shown in Table 4.

LSPS. We report the performance on the LSPS dataset in Table 6. We observe that with multiple anchors, our PCNet outperforms OIM [28] and APNet [37] by 6.9% and 3.2% in rank-1 accuracy respectivley, with significantly faster inference speed as illustrated in Table 4. LSPS contains lots of partial persons at test time. CPP of PCNet does not rely on feature cropping *w.r.t* bounding boxes, thus is robust to misalignment issue caused by partial persons.

CUHK-SYSU. The performances are listed with the gallery size of 100. PCNet† outperforms the CGPS [33] by 2.0% in rank-1. Note that CGPS requires $\mathcal{O}(N^2)$ computation during inference. Though exhibiting the faster inference speed as shown in Table 4, PCNet does not perform as well as the recent joint models. We attribute the lower performance to the RetinaNet which PCNet is based on. Performance of the one-stage RetinaNet heavily relies on the anchor setting. Due to the various aspect ratios and scales of CUHK-SYSU, the anchor setting is hard to tune. We also evalute the performance of different gallery sizes in supplementary material. The gap with other methods shrinks as gallery size increases.

Methods	Ref	R50	Det	**CUHK**		**PRW**	
				mAP	r-1	mAP	r-1
MGTS [1]	ECCV18	×2	FR	83.0	83.7	32.6	**72.1**
CLSA [32]	ECCV18	×2	FR	87.2	88.5	38.7	65.0
RDLR [7]	ICCV19	×2	FP	**93.0**	**94.2**	**42.9**	**72.1**
NPSM [8]	ICCV17	×1	-	77.9	81.2	24.2	53.1
CGPS [33]	CVPR19	×1	FR	84.1	86.5	33.4	73.6
QEEPS [19]	CVPR19	×1	FR	88.9	89.1	37.1	76.7
IGPN^{256} [5]	CVPR20	×1	FR	85.3	85.7	42.9	82.1
IGPN [5]	CVPR20	×1	FR	**90.3**	**91.4**	**47.2**	**87.0**
OIM [28]	CVPR17	×1	FR	75.5	78.7	21.3	49.9
NAE+ [2]	CVPR20	×1	FR	**92.1**	**92.9**	44.0	81.1
APNet [37]	CVPR20	×1*	FR	88.9	89.3	41.9	81.4
BINet [4]	CVPR20	×1	FR	90.0	90.7	45.3	81.7
PCNet	ours	×1	Ret	86.3	89.4	41.9	83.1
PCNet†	ours	×1	Ret	87.7	88.5	**45.5**	**83.7**

Table 5. Comparison with recent works. R50 and Det refer to ResNet50 and detector, respectively. $\times n$ represents number of ResNet50 the model utilizes. FR, FP and Ret represent the detectors Faster-RCNN [23], Feature Pyramid Network [16] and RetinaNet [17] respectively. $\times 1^{*}$ means that the parameter approaches ×2 though it is a single model. PCNet† assigns multiple anchors for each position in the feature map. Best results in each block are marked in **bold**.

Methods	Ref	R50	Det	**LSPS**	
				mAP	r-1
OIM [28]	CVPR17	×1	FR	14.0	46.0
APNet [37]	CVPR20	×1*	FR	16.4	49.7
PCNet	ours	×1	Ret	16.7	49.5
PCNet†	ours	×1	Ret	**17.5**	**52.9**

Table 6. Comparison on LSPS dataset. Based on the dataset from the work of APNet [37], we report the performance with faces being blurred. The overall performances drop a bit compared to the ones reported in APNet.

5. Conclusion

In this paper, we propose the PCNet based on one-stage detector, which is the first joint model for person search without utilizing feature cropping for reid feature extraction. PCNet extracts reid feature by adaptively aggregating the points in context throughout the feature pyramid. The CPP module aggregates the informative points within each level, while the CSA considers the multi-scale context across levels. Thus, PCNet can greatly utilize the context outside the bounding box, leading to discriminative reid feature and robustness to detection errors. Experiments show that PCNet achieves competitive performance while exhibiting faster inference speed.

References

[1] Di Chen, Shanshan Zhang, Wanli Ouyang, Jian Yang, and Ying Tai. Person search via a mask-guided two-stream cnn model. In *ECCV*, 2018.

[2] Di Chen, Shanshan Zhang, Jian Yang, and Bernt Schiele. Norm-aware embedding for efficient person search. In *CVPR*, 2020.

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017.

[4] Wenkai Dong, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Bi-directional interaction network for person search. In *CVPR*, 2020.

[5] Wenkai Dong, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Instance guided proposal network for person search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[6] Ross Girshick. Fast r-cnn. In *CVPR*, 2015.

[7] Chuchu Han, Jiacheng Ye, Yunshan Zhong, Xin Tan, Chi Zhang, Changxin Gao, and Nong Sang. Re-id driven localization refinement for person search. In *ICCV*, 2019.

[8] Liu Hao, Jiashi Feng, Zequn Jie, Karlekar Jayashree, Zhao Bo, Meibin Qi, Jianguo Jiang, and Shuicheng Yan. Neural person search machines. In *ICCV*, 2017.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen. Interaction-and-aggregation network for person re-identification. In *CVPR*, 2019.

[13] Mahdi M Kalayeh, Emrah Basaran, Muhittin Gökmen, Mustafa E Kamasak, and Mubarak Shah. Human semantic parsing for person re-identification. In *CVPR*, 2018.

[14] Jianing Li, Shiliang Zhang, Qi Tian, Meng Wang, and Wen Gao. Pose-guided representation learning for person re-identification. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[15] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *CVPR*, 2018.

[16] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[18] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.

[19] Bharti Munjal, Sikandar Amin, Federico Tombari, and Fabio Galasso. Query-guided end-to-end person search. In *CVPR*, 2019.

[20] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[21] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *ICCV*, 2017.

[22] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[23] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[24] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. Pose-driven deep convolutional model for person re-identification. In *ICCV*, 2017.

[25] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *ECCV*, 2018.

[26] Chiat-Pin Tay, Sharmili Roy, and Kim-Hui Yap. Aanet: Attribute attention network for person re-identifications. In *CVPR*, 2019.

[27] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *ICCV*, 2019.

[28] Xiao Tong, Li Shuang, Bochao Wang, Lin Liang, and Xiaogang Wang. Detection and identification feature learning for person search. In *CVPR*, 2017.

[29] Cheng Wang, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Mancs: A multi-task attentional network with curriculum sampling for person re-identification. In *ECCV*, 2018.

[30] Guanshuo Wang, Yufeng Yuan, Chen Xiong, Jiwei Li, and Zhou Xi. Learning discriminative features with multiple granularities for person re-identification. In *ACM MM*, 2018.

[31] Longhui Wei, Shiliang Zhang, Hantao Yao, Wen Gao, and Qi Tian. Glad: Global-local-alignment descriptor for pedestrian retrieval. In *ACM MM*, 2017.

[32] Lan Xu, Xiatian Zhu, and Shaogang Gong. Person search by multi-scale matching. In *ECCV*, 2018.

[33] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, and Xiaokang Yang. Learning context graph for person search. In *CVPR*, 2019.

[34] Wenjie Yang, Houjing Huang, Zhang Zhang, Xiaotang Chen, Kaiqi Huang, and Shu Zhang. Towards rich feature discovery with class activation maps augmentation for person re-identification. In *CVPR*, 2019.

[35] Feng Zheng, Cheng Deng, Xing Sun, Xinyang Jiang, Xiaowei Guo, Zongqiao Yu, Feiyue Huang, and Rongrong Ji. Pyramidal person re-identification via multi-loss dynamic training. In *CVPR*, 2019.

[36] Liang Zheng, Hengheng Zhang, Shaoyan Sun, Manmohan Chandraker, and Qi Tian. Person re-identification in the wild. In *CVPR*, 2017.

[37] Yingji Zhong, Xiaoyu Wang, and Shiliang Zhang. Robust partial matching for person search in the wild. In *CVPR*, 2020.

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