

OIMNet++: Prototypical Normalization and Localization-aware Learning for Person Search

Sanghoon Lee¹, Youngmin Oh¹, Donghyeon Baek¹,
Junghyup Lee¹, and Bumsub Ham^{1,2*}

<https://cvlab.yonsei.ac.kr/projects/OIMNetPlus>

¹Yonsei University ²Korea Institute of Science and Technology (KIST)

Abstract. We address the task of person search, that is, localizing and re-identifying query persons from a set of raw scene images. Recent approaches are typically built upon OIMNet, a pioneer work on person search, that learns joint person representations for performing both detection and person re-identification (reID) tasks. To obtain the representations, they extract features from pedestrian proposals, and then project them on a unit hypersphere with L2 normalization. These methods also incorporate all positive proposals, that sufficiently overlap with the ground truth, equally to learn person representations for reID. We have found that 1) the L2 normalization without considering feature distributions degenerates the discriminative power of person representations, and 2) positive proposals often also depict background clutter and person overlaps, which could encode noisy features to person representations. In this paper, we introduce OIMNet++ that addresses the aforementioned limitations. To this end, we introduce a novel normalization layer, dubbed ProtoNorm, that calibrates features from pedestrian proposals, while considering a long-tail distribution of person IDs, enabling L2 normalized person representations to be discriminative. We also propose a localization-aware feature learning scheme that encourages better-aligned proposals to contribute more in learning discriminative representations. Experimental results and analysis on standard person search benchmarks demonstrate the effectiveness of OIMNet++.

1 Introduction

Person search aims at jointly localizing and re-identifying a query person from a set of raw scene images [36,42]. Different from person re-identification (reID) [39,41], person search incorporates pedestrian detection in a unified framework, facilitating retrieving query persons without hand-labelled [22] or auto-detected [18,40] pedestrian bounding boxes during inference. This provides a wide range of applications, particularly where the bounding boxes are expensive to obtain, including large-scale surveillance and pedestrian analysis. Person search is extremely challenging, since it inherits problems from both pedestrian detection (*e.g.*, background clutter and scale variations) and person reID (*e.g.*, large intra-class variations).

* Corresponding author.

Recent approaches to person search focus on extracting person representations that are eligible to handle both detection and reID tasks [3,7,16,37]. They typically build on top of OIMNet [36], a pioneer work on person search. OIMNet and its variants leverage a 2D object detection framework [28], and obtain person features from pedestrian proposals. The obtained feature representations are further projected onto a unit hypersphere by applying L2 normalization. These approaches also leverage the OIM loss [36] that employs a lookup table (LUT) consisting of features that describe each ID in a training set, and exploits them as supervisory signals to guide learning discriminative features. While these methods have allowed significant advances for person search, there are two main limitations. First, L2 normalization is effective only when the person representations are roughly centered around zero. Moreover, the person representations are encouraged to have similar variances (*e.g.*, unit variance) across channel dimensions over a whole training set. This is because it promotes each channel to contribute equally in determining a decision boundary, while preventing a small set of channels from dominating the decision boundary. Current person search methods implicitly assume that features obtained from pedestrian proposals have zero-mean and unit variance, *i.e.*, standardized. Accordingly, they project the features onto a unit hypersphere using L2 normalization, without explicit constraints, which rather degenerates the discriminative power of person representations (Fig. 1(b)). A standard way of alleviating this problem is to *shift and scaling* the feature distribution using BatchNorm [14]. However, we have observed that the standardization using BatchNorm is not able to offer satisfactory results. This is mainly due to the extreme class imbalance across person IDs, which is largely inevitable in training person search networks. Specifically, human trajectory patterns are highly diverse across person IDs, and the extent of a person’s exposure to cameras in public is difficult to model [6]. Person search datasets obtained from real-world environments thus contain person images whose ID labels are extremely imbalanced, forming a long-tail distribution across person IDs. As BatchNorm calibrates the feature distribution with input features directly, it is easily biased towards dominant IDs. This restrains the discriminative power of features on a unit hypersphere (Fig. 1(c)). Second, the OIM loss updates each feature in the LUT using an exponential moving average with a fixed momentum. Each object proposal contributes equally when updating the corresponding feature in the LUT, regardless of the localization accuracy w.r.t the ground truth. This is suboptimal in that not all object proposals are equally discriminative, as they are often misaligned during training. Namely, the OIM loss ignores the fact that reID relies on detection during training; better localized object proposals could contribute more to learning discriminative features for reID.

In this paper, we present a simple but highly effective approach to person search, dubbed OIMNet++, built upon OIMNet [36]. We introduce two modifications of OIMNet that offer significant performance gains over the vanilla one. First, we propose a novel normalization layer, ProtoNorm, to obtain discriminative person representations on a hypersphere. ProtoNorm exploits prototypical

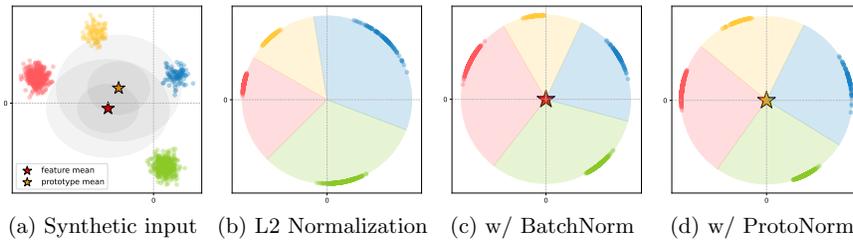


Fig. 1: We visualize in (a) synthetic 2D features in circles, where each color represents an ID label. We represent mean obtained from input features and ID prototypes with stars colored in red and yellow, respectively. Note that pink and green features are sampled $4\times$ more. The features are clearly not zero-centered with unit variance. In this case, simply applying L2 normalization degenerates the discriminative power, as shown in (b), where background colors indicate decision boundaries. Adopting a feature standardization with feature mean and variance, *i.e.*, in a BatchNorm-fashion, prior to L2 normalization, alleviates this problem in (c). However, this does not consider a sample distribution across IDs to calibrate the feature distribution. The distribution is thus biased towards majority IDs, which weakens the inter-class separability. Instead, calibrating feature distribution using ID prototypes with ProtoNorm provides highly discriminative L2-normalized features in (d), where each ID is assigned similar angular space. (Best viewed by zooming in with color.)

features for individual person IDs to calibrate feature distributions, such that the distributions are less biased towards dominant IDs (Fig. 1(d)). This enhances the intra-class compactness for majority IDs, as well as the inter-class separability for minority ones. Second, we propose a localization-aware OIM loss (LOIM) that adaptively updates each feature in the LUT w.r.t the localization accuracy of each object proposal during training. As better localized object bounding boxes translate to less noisy features, *e.g.*, with less background clutter and person overlaps, we encourage better-aligned proposals to incorporate more to construct the LUT. In this way, we can train a network with more discriminative features as guidance. By employing ProtoNorm and the LOIM loss to a vanilla OIMNet [36], we establish OIMNet++, setting a new state of the art on standard benchmarks for person search, including CUHK-SYSU [36] and PRW [42]. The main contributions of our work can be summarized as follows:

- We introduce a simple yet effective normalization layer, ProtoNorm, to learn discriminative person representations for person search. ProtoNorm alleviates the class imbalance problem in person search, while retaining the effectiveness of BatchNorm.
- We propose a novel LOIM loss that assigns larger degree of update to better aligned object proposals. This allows to compose the LUT with less noisy features, better guiding the discriminative feature learning for person search, compared with the OIM loss [36].
- We set a new state of the art on standard person search benchmarks [36,42] and demonstrate the effectiveness of our approach with extensive experiments and ablation studies.

2 Related Work

2.1 Person Search

Many approaches attempt to decompose person search into pedestrian detection and person reID tasks. Current person search methods can be categorized into two groups. The first line of works [2,8,10,17] design a two-step method that performs pedestrian detection to obtain cropped person images, which are then sequentially fed into a person reID network to extract person representations. These methods further employ auxiliary modules to enhance the discriminative power of person representations. For example, they exploit an off-the-shelf instance segmentation network [20] to focus more on foreground regions [2], design a multi-layer feature aggregation module [17], or take a query image as an additional input with a siamese network [8]. While these methods have achieved remarkable performances for person search, they require a heavy computational cost, due to the separate feature extractors, and prohibit joint optimization between detection and reID tasks. Another line of works [3,7,16,37] formulate person search as a joint feature learning problem, and optimize person search models with multi-task objectives in an end-to-end manner. Given an input image, they extract joint features for detection and reID tasks, enabling an efficient pipeline. A main challenge in training these models is known to be the contradictory objectives between detection and reID; pedestrian detection tries to extract the commonness across different person IDs, while reID focuses on the uniqueness [2,3]. To address this problem, recent works propose to disentangle person representations into detection- and reID-related features [3], stop gradient flows in certain layers of a network [16], or extract reID features prior to detection [37]. Similar to ours, the work of [16] proposes to adaptively update features in the LUT, while considering the hardest negative example. This, however, still ignores the localization accuracy, when adjusting momentum values for the updates. All the aforementioned person search approaches also ignore the class imbalance problem across person IDs and the corresponding detrimental effect in learning discriminative person representations.

2.2 Feature Normalization

The seminal work of [14] introduces a BatchNorm layer, where intermediate features of deep networks are standardized, followed by applying an affine transform. This improves the generalization capability of a network and stabilizes the training process. BatchNorm has become an indispensable component of modern deep neural networks [12,31]. While many variants have been proposed for various applications [1,13,19,30,32,34,35,38], there are no attempts to leverage normalization methods for person search, to our knowledge. Thus, we mainly describe representative works in the context of person reID, which is closely related to person search.

The work of [25] has shown that adding a BatchNorm layer right before a final classifier boosts performance by a large margin, especially when computing

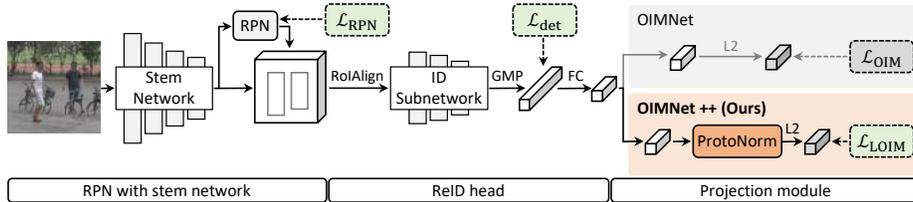


Fig. 2: An overview of OIMNet++. Similar to OIMNet [36], OIMNet++ mainly consists of three parts: An RPN with a stem network, a reID head, and a projection module. The main differences between OIMNet++ (bottom) and OIMNet (top) are the projection module and the training loss. We incorporate a ProtoNorm layer to explicitly standardize features prior to L2 normalization, while considering the class imbalance problem in person search. We also exploit the LOIM loss that leverages localization accuracies of object proposals to learn discriminative features. See text for details.

distances with L2 normalized person representations. As will be shown in our experiments, this simple normalization scheme also notably improves the performance of person search methods, establishing a strong baseline. However, we have found that a vanilla BatchNorm might be susceptible to the presence of class imbalance, since statistics are dominated by majority IDs. To address this issue, ProtoNorm incorporates minority IDs to calibrate feature distributions. This standardizes features better than BatchNorm, outperforming the strong baseline. Recently, the works of [4,15,44] propose to exploit both InstanceNorm [32] and BatchNorm in order to alleviate the influence of identity-irrelevant features. Although these methods achieve a better generalization ability, especially in a cross-domain setting, they still ignore the class imbalance across person IDs, when calibrating feature distributions. This suggests that our ProtoNorm further boosts the performance in a complementary way, providing more accurate feature distributions for InstanceNorm and BatchNorm. Instead of combining different normalization layers, the work of [45] adopts multiple BatchNorm layers. It computes the mini-batch statistics separately for each camera to reduce the distribution gap across different cameras. Instead of standardizing features for each ID, we exploit a single set of statistics based on prototypical features for individual person IDs for standardization.

3 Method

We show in Fig. 2 an overview of our approach. Following the previous works [3,7,16,43], we build OIMNet++ upon OIMNet [36]. In this section, we briefly review OIMNet (Sec. 3.1), and provide a detailed description of OIMNet++ (Sec. 3.2), including ProtoNorm and the LOIM loss.

3.1 OIMNet

Network. OIMNet [36] mainly consists of three components: A region proposal network (RPN) with a stem network, a reID head, and a projection module. Given an input image, it employs the RPN to generate pedestrian proposals

with feature maps obtained from the stem network. These proposals form candidates to be matched with a query person. While features obtained from a stem network are able to discriminate between persons and background, they are not able to discriminate between person IDs. Thus, OIMNet further employs a reID head, consisting of an identification subnetwork with a global max pooling (GMP) layer, followed by a fully-connected (FC) layer, to refine the features from object proposals. The projection module further projects the features on a unit hypersphere with L2 normalization. At test time, OIMNet computes distances between L2-normalized features for query and gallery persons for matching.

Loss. OIMNet is trained with the following loss:

$$\mathcal{L} = \lambda_{\text{RPN}}\mathcal{L}_{\text{RPN}} + \lambda_{\text{det}}\mathcal{L}_{\text{det}} + \mathcal{L}_{\text{OIM}}, \quad (1)$$

where \mathcal{L}_{RPN} , \mathcal{L}_{det} , and \mathcal{L}_{OIM} are RPN, detection, and OIM losses, respectively, and λ_{RPN} and λ_{det} are balancing parameters for corresponding terms. The RPN and detection losses consist of binary classification and offset regression terms, for anchors and proposals, respectively. They facilitate OIMNet to perform pedestrian detection.

On the other hand, to learn discriminative person representations for reID, OIMNet employs the OIM loss [36]. It leverages a LUT that stores features representing labelled IDs in a training set. We denote by $\mathbf{v}_l \in \mathbb{R}^D$, $l \in \{1, \dots, L\}$ an L2-normalized feature within the LUT representing the l -th ID, where L is the number of labelled IDs and D is the channel dimension. Meanwhile, there are also pedestrian instances without corresponding ID labels. These instances form a set of unlabelled IDs that can be regarded as negatives for the labelled ones. The OIM loss also leverages a circular queue to store features obtained from the unlabelled IDs for training. Let us denote by $\mathbf{u}_q \in \mathbb{R}^D$, $q \in \{1, \dots, Q\}$ an L2-normalized q -th feature within the queue, where Q is the queue size. Given an L2-normalized feature $\mathbf{x}_t \in \mathbb{R}^D$, with the t -th ID label, the OIM loss is formally defined by

$$\mathcal{L}_{\text{OIM}} = \mathbb{E}_{\mathbf{x}}[-\log p_t], \quad (2)$$

where

$$p_t = \frac{\exp(\mathbf{v}_t^\top \mathbf{x}_t / \tau)}{\sum_{i=1}^L \exp(\mathbf{v}_i^\top \mathbf{x}_t / \tau) + \sum_{j=1}^Q \exp(\mathbf{u}_j^\top \mathbf{x}_t / \tau)}, \quad (3)$$

and τ is a temperature value. Namely, this term encourages an input feature to be embedded near to the corresponding ID feature within the LUT, while being distant from negative ones within the LUT and the circular queue. The OIM loss subsequently updates \mathbf{v}_t with a fixed momentum η as follows:

$$\mathbf{v}_t \leftarrow \eta \mathbf{v}_t + (1 - \eta) \mathbf{x}_t. \quad (4)$$

Note that \mathbf{v}_t is L2 normalized after every update. The OIM loss stabilizes a training process, even with a large number of person IDs, and allows to leverage unlabelled IDs for learning discriminative person representations.

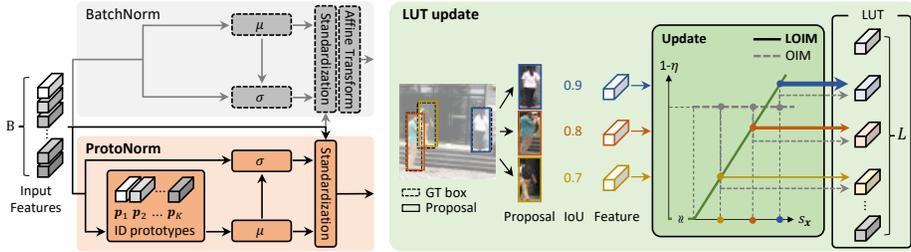


Fig. 3: **Left:** A comparison between BatchNorm [14] and ProtoNorm. BatchNorm computes feature statistics with input features directly. On the other hand, ProtoNorm aggregates multiple features with the same ID into a single prototype. ProtoNorm then computes mean and variance based on the prototype features, alleviating the bias towards dominant IDs. **Right:** LUT update scheme within the LOIM loss. The vanilla OIM loss assigns equal momentum values for all positive proposals, regardless of the localization qualities. The LOIM loss, instead, assigns an adaptive momentum value to each proposal w.r.t its IoU with the ground truth. Thicker arrows indicate larger degree of updates to the LUT. See text for details.

3.2 OIMNet++

While OIMNet [36] has allowed significant advances for person search, there are two main limitations. First, performing L2 normalization without considering the feature distribution could be problematic, especially when features are not well standardized. In this case, L2 normalization rather degenerates the discriminative power of the features on a unit hypersphere. Note that we can employ a BatchNorm [14] layer, prior to L2 normalization, that explicitly standardizes features to be zero-centered with a unit variance. As will be shown in our experiments, this improves the performance of OIMNet drastically. However, current person search datasets [36,42] form a long-tail distribution across ID labels, making BatchNorm a suboptimal choice for standardization. This weakens the intra-class compactness for majority IDs and the inter-class separability for minority ones (Fig. 1(c)). Second, during training, the RPN outputs pedestrian proposals that are often misaligned. On the one hand, small misalignments are acceptable, making the reID head robust to misalignment at test time. Large misalignments caused by person overlaps or mis-detections (Fig. 3), on the other hand, distract discriminative feature learning. The OIM loss simply assumes that all proposals could contribute equally to learning discriminative features, and updates the LUT with a fixed momentum, as in (4).

In the following, we describe our approach, dubbed OIMNet++, that addresses the aforementioned limitations. Since OIMNet++ and OIMNet [36] share the RPN and the reID head, together with the training losses for the RPN and pedestrian detection, we mainly describe ProtoNorm in the projection module and the LOIM loss in detail.

ProtoNorm. BatchNorm calibrates feature distributions using channel-wise feature statistics computed with input features directly. This makes the distributions susceptible to frequencies of ID labels, and thus they are biased towards

frequent IDs. Instead of directly calibrating the distributions, we propose to exploit mini-batch statistics over prototypical features for individual person IDs. Specifically, to obtain the prototypical feature for a particular ID, we average features for the corresponding ID within a mini-batch. We provide in Fig. 3 an illustration of ProtoNorm and BatchNorm.

Concretely, let us denote a set of features, $\mathbf{X} = \{\mathbf{x}^1, \dots, \mathbf{x}^B\}$, where B is the mini-batch size, along with corresponding set of ID labels $Y = \{y^1, \dots, y^B\}$, where $y^i \in \{1, \dots, L\}$. We denote by $\mathbf{X}^i(d)$ the d -th channel element within the i -th feature \mathbf{x}^i . We first obtain a prototypical feature representing t -th ID, denoted by $\mathbf{p}_t \in \mathbb{R}^D$, as follows:

$$\mathbf{p}_t(d) = \frac{\sum_{b=1}^B \mathbf{X}^b(d) \mathbb{1}[y^b = t]}{\sum_{b=1}^B \mathbb{1}[y^b = t]}, \quad (5)$$

where $\mathbb{1}[\cdot]$ is an indicator function whose value is 1 when the argument is true, and 0 otherwise. With the prototypical features, \mathbf{p}_t , in hand, we compute mean and variance vectors of input features \mathbf{X} , denoted by $\boldsymbol{\mu} \in \mathbb{R}^D$ and $\boldsymbol{\sigma} \in \mathbb{R}^D$, respectively, as follows:

$$\boldsymbol{\mu}(d) = \frac{1}{K} \sum_{k=1}^K \mathbf{p}_k(d) \quad \text{and} \quad \boldsymbol{\sigma}(d) = \sqrt{\frac{1}{B} \sum_{b=1}^B (\mathbf{X}^b(d) - \boldsymbol{\mu}(d))^2}, \quad (6)$$

where K is the number of unique IDs in the set of ID labels Y . We then standardize the features as $\frac{\mathbf{X}^b(d) - \boldsymbol{\mu}(d)}{\boldsymbol{\sigma}(d)}$ ¹. Instead of using input features directly, exploiting prototypical features in ProtoNorm offers standardization with mean and variance that are less biased towards dominant IDs. ProtoNorm adopts a weighted summation of the input features, where the weight can be represented as $\frac{1}{K \sum_{b=1}^B \mathbb{1}[y^b = t]}$ for the t -th ID, and is inversely proportional to an occurrence of an ID. Namely, ProtoNorm adaptively assigns larger weight values to minority ID features, while setting to smaller ones for majority IDs. This steers the mean towards minority IDs, encouraging inter-class separateness for L2-normalized person representations. Note that we may apply the weighting method in an image level to address the class imbalance problem across person IDs, *e.g.*, using a class-balanced mini-batch sampling technique [25]. This is, however, not scalable to person search, as each image depicts different number of person instances.

Similar to BatchNorm, we track running mean and variance during training and exploit them as estimates for a global distribution of prototypical features at test time. This assumes that mean and variance sampled from the global distribution are less biased towards dominant IDs, enables calibrating the feature distribution without ID labels at test time.

¹ We could apply a learnable affine transform after standardization, similar to BatchNorm. We have empirically found that affine parameters for scaling and offset converge to constant (but not zero) and zero values, respectively. This suggests that the effect of the affine transform is canceled out by L2 normalization, and thus we omit the transform when ProtoNorm is followed by L2 normalization.

LOIM loss. The OIM loss [36] encourages involving positive proposals, that overlap with a ground truth more than a pre-defined threshold, equally in learning discriminative features. Since not all proposals are equally created, they should contribute to feature learning differently. The features in the LUT should thus be chosen more carefully. Specifically, the LUT should accept discriminative features only for the update, while discarding noisy ones. However, estimating the degree of noise within a feature obtained from a proposal is ambiguous. Previous works [2,43] have relied on an auxiliary supervision obtained from, *e.g.*, an off-the-shelf pose estimator [9], or an instance segmentation network [20]. They are computationally expensive, and require additional datasets for training. Intersection-of-union (IoU) between a proposal and its ground-truth bounding box, on the other hand, serves as a good indicator for estimating extent of noise within the proposal. Namely, a proposal with a large IoU score tightly covers a person-of-interest, with less background clutter and person overlaps. Note that ground-truth bounding boxes are already available in person search datasets, suggesting that leveraging IoU scores does not require additional labelling effort. To implement this idea, we exploit the IoU score of each proposal to update features in the LUT. We assign small momentum values for proposals with large IoU scores, as these proposals are able to provide less noisy features. Let us denote by $s_{\mathbf{x}} \in [0, 1]$ the IoU score between a proposal and its ground-truth bounding box. Concretely, we update the features within the LUT as follows:

$$\mathbf{v}_t \leftarrow (1 - c_{\mathbf{x}})\mathbf{v}_t + c_{\mathbf{x}}\mathbf{x}_t, \quad (7)$$

where $c_{\mathbf{x}}$ computes an adaptive momentum, defined using the IoU score, as follows:

$$c_{\mathbf{x}} = \text{clip}(s_{\mathbf{x}}, 0, 1 - \epsilon). \quad (8)$$

$\text{clip}(\cdot, 0, 1 - \epsilon)$ is a clipping function with lower and upper bounds set to 0 and $1 - \epsilon$, respectively, and ϵ is a hyperparameter. We simply set ϵ to 0.1 to prevent perfectly-aligned proposals (*i.e.*, $s_{\mathbf{x}} = 1$) from totally overriding the corresponding feature after the LUT update. The LOIM loss is defined by $\mathcal{L}_{\text{LOIM}} = \mathbb{E}_{\mathbf{x}}[-\log p_t]$, whereas \mathbf{v}_t is updated with the adaptive momentum, as in (7). By leveraging IoU scores to define momentum values, noisy features are discouraged to form the LUT, better guiding discriminative feature learning. This also encourages a network to favor extracting better-localized features at test time.

4 Experiments

4.1 Implementation details

Network. Following the previous works [3,7,16,43], we exploit ResNet50 [12] pretrained on ImageNet [29] as our backbone network. Concretely, we split ResNet50 at conv4-6 layer, and establish a stem network and an identification subnetwork with preceding and succeeding layers, respectively. We employ

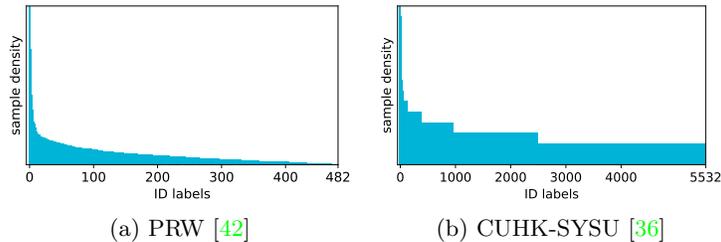


Fig. 4: Density of training samples across ID labels in PRW [42] and CUHK-SYSU [36] datasets. Since human trajectory patterns in public are highly diverse across persons [6], person IDs, collected from a real-world environment, are extremely imbalanced, forming a long-tail distribution.

RoIAlign [11] to crop 14×14 proposal feature maps obtained from the stem network, and set the channel dimension of person representations, D , to 256, following OIMNet [36]. We leverage the circular queue with size, Q , 5000 and 500 for CUHK-SYSU [36] and PRW [42], respectively. Note that we exploit feature maps obtained from conv5-3 of ResNet50 only as inputs to the RoIAlign layer, rather than fusing multi-level features in a pyramid fashion [23].

Dataset. We use two standard benchmarks for training and evaluation: 1) The PRW dataset [42] is collected using 6 cameras in a university. It includes 11,816 images with 4,310 pedestrian bounding boxes, which are labelled to 932 IDs. We adopt the train/test splits provided by the authors, and use 5,704 images containing 482 IDs for training. There are 2,057 query images with 450 different IDs, whereas a gallery set contains 6,112 images. For each query image, we use the whole gallery set for evaluation. 2) The CUHK-SYSU dataset [36] is collected from urban scenes and movie clips. It is composed of 18,184 images that contain 96,143 pedestrian bounding boxes with 8,432 labelled IDs. We use official train/test splits provided by the authors. Concretely, we use 11,206 images with 5,532 IDs for training, 6,978 gallery images with 2,900 query person instances for testing. Following the standard protocol [36], we sample 100 gallery images for each query person during evaluation. For both datasets, we adjust input images to the size of $900 \times 1,500$ for training and testing. We visualize in Fig. 4 the distributions for the number of training samples across ID labels.

Training. We use the same training strategy and hyperparameter setting as the ones in [3]. Specifically, we train our model for 20 epochs for both the PRW [42] and CUHK-SYSU [36] datasets, and set the batch size to 5. We employ a warm-up strategy, gradually increasing a learning rate to 0.003 during the first epoch, which is divided by 10 at the 16th epoch. We assign the same balancing parameter for each term in the training objective, *i.e.*, $\lambda_{\text{RPN}} = 1$ and $\lambda_{\text{det}} = 1$, and set the temperature value τ to 0.33. We set the momentum value η to 0.5 for training a network with the OIM loss. We train our model using PyTorch [27] end-to-end, which takes about 5 and 10 hours for PRW and CUHK-SYSU datasets, respectively, with a Titan RTX GPU.

Table 1: Quantitative comparison with the state of the art for person search. We report mAP (%) and rank-1 accuracy (%) on CUHK-SYSU [36] and PRW [42] datasets. For each category of person search methods, numbers in bold indicate the best and under-scored ones indicate the second best. R50 and DC are abbreviations for ResNet50 [12] and deformable convolution [5], respectively. We report our average scores over 4 runs with standard deviations in parentheses.

	Method	Backbone	CUHK-SYSU [36]		PRW [42]	
			mAP	rank-1	mAP	rank-1
Two-step	MGTS [2]	R50	83.0	83.7	32.6	72.1
	RDLR [10]	R50	<u>93.0</u>	<u>94.2</u>	42.9	70.2
	IGPN [8]	R50	90.3	91.4	47.2	<u>87.0</u>
	TCTS [33]	R50	93.9	95.1	<u>46.8</u>	87.5
End-to-end	OIM [36]	R50	75.5	78.7	21.3	49.4
	NPSM [24]	R50	77.9	81.2	24.2	53.1
	QEEPS [26]	R50	88.9	89.1	37.1	76.7
	NAE+ [3]	R50	<u>92.1</u>	92.9	44.0	81.1
	BINet [7]	R50	90.0	90.7	45.3	81.7
	PGA [16]	R50	90.2	91.8	42.5	<u>83.5</u>
	AlignPS [37]	R50	93.1	<u>93.4</u>	<u>45.9</u>	81.9
	OIMNet++ (Ours)	R50	93.1 (0.24)	93.9 (0.30)	46.8 (0.51)	83.9 (0.59)
	PGA* [16]	R50-Dilation	92.3	94.7	44.2	85.2
	AlignPS+ [37]	R50-DC [5]	94.0	<u>94.5</u>	<u>46.1</u>	82.1
OIMNet+++ (Ours)	R50-ProtoNorm	<u>93.1</u> (0.21)	94.1 (0.25)	47.7 (0.19)	<u>84.8</u> (0.20)	

4.2 Comparison with the state of the art

We provide in Table 1 a quantitative comparison between our method with the state of the art [2,3,7,8,10,16,24,26,33,36,37] for person search. For fair comparison, we categorize person search methods into two-step [2,8,10,33] and end-to-end [3,7,16,24,26,36,37] approaches. The end-to-end approaches are further split into two groups according to the backbone network.

Overall, we can see from the experimental results that OIMNet++ provides highly discriminative person representations for person search. In particular, OIMNet++ shows high mAP scores. This indicates that our model is able to offer retrieval results with less false positives, *i.e.*, matches that are not likely to be a false alarm. Among the end-to-end approaches that adopt vanilla ResNet50 [12] as a backbone network, OIMNet++ achieves the state-of-the-art performance. Note that OIMNet++ even outperforms PGA [16] that requires additional parameters and computational overhead at test time due to an auxiliary attention module.

Recent works [16,37] modify a backbone network to further boost the performance. For example, PGA* [16] provides a variant by reducing the dilation rate of a conv5 block in ResNet50 from 2 to 1 to obtain features of high resolution. AlignPS+ [37] additionally exploits deformable convolutions from conv3 to conv5 blocks within ResNet50. Similarly, we replace BatchNorm layers within

Table 2: Ablative analysis of our approach. We measure the mAP (%) and rank-1 accuracy (%) on PRW [42] using person representations obtained from detected and annotated bounding boxes to evaluate search and reID performances separately. BN and PN indicates BatchNorm [14] and ProtoNorm, respectively. Numbers in bold indicate the best performance and the underscored ones indicate the second best. All results are obtained by averaging scores over 4 runs.

BN	PN	\mathcal{L}_{OIM}	$\mathcal{L}_{\text{LOIM}}$	Search		ReID	
				mAP	rank-1	mAP	rank-1
		✓		42.0	80.5	44.3	82.6
✓		✓		44.3	81.5	46.6	83.2
	✓	✓		<u>46.3</u>	82.7	<u>48.4</u>	84.6
✓			✓	45.1	<u>82.9</u>	47.4	<u>84.7</u>
	✓		✓	46.8	83.9	49.0	86.2

a conv5 block with ProtoNorm for OIMNet+++. In this case, we apply a learnable affine transformation after ProtoNorm layers within ResNet50, as in BatchNorm. Note that our modification, compared to other ones for PGA [16] and AlignPS [37], does not require additional computational overheads or parameters at test time. This places our model at a disadvantage, but we can see from the the results in the last row of Table 1 that OIMNet+++ shows the person search performances comparable with competitive approaches, even including two-step ones [2,8,10,33].

4.3 Discussion

Ablation study. We provide in Table 2 an ablation study of our approach using different combinations of components and losses. We measure mAP (%) and rank-1 accuracy (%) on the test set of PRW [42]. To better evaluate the discriminative power of person representations, we also measure the reID performance using annotated bounding boxes. We can see from the first and second rows that a person search model trained with the OIM loss only shows the worst performance², and incorporating a BatchNorm layer can boost the performance significantly. This suggests that applying the L2 normalization to person representations without standardization techniques degenerates the discriminative power. By replacing BatchNorm with our ProtoNorm in the third row, we can achieve additional performance gains. This demonstrates the effectiveness of ProtoNorm calibrating feature distributions while explicitly considering the class imbalance problem in person search. The results coincide with our finding in the toy experiment illustrated in Fig. 1, confirming once more the importance of a class-unbiased standardization scheme prior to projecting features on a unit

² The model in the first row is exactly same as the original OIMNet [36], apart from the RoIAlign module in ours. Note that re-implementing OIMNet using common practices in recent works [3,16,21] (an improved learning rate scheduler, larger batch size, and the RoIAlign module) performs significantly better than the original OIMNet shown in Table 1. Similar findings are also reported in [3,21].

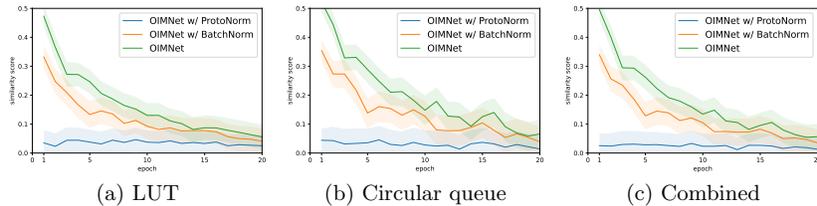


Fig. 5: We plot average cosine similarity scores between features in (a) the LUT, (b) the circular queue, and a (c) concatenation of the two, over training epochs on the PRW dataset [42]. To better demonstrate the advantages of normalization operators, we train all models with the OIM loss. We also illustrate the standard deviation in transparent colors. (Best viewed in color.)

hypersphere. We can observe from the second and fourth rows that the LOIM loss boosts the performance drastically. This suggests that selectively updating LUT features with the localization accuracy in the LOIM loss helps learning more discriminative representations than the OIM loss. Lastly, jointly exploiting ProtoNorm and the LOIM loss in the last row shows the best performance, and the two components complement each other.

Inter-class separability. To demonstrate the effectiveness of ProtoNorm, we compute cosine similarity scores between features in the LUT, the circular queue, and both. We average the scores for all possible pairs and show in Fig. 5 the results over training epochs. We compare the results between three variants of OIMNet [36]; a vanilla OIMNet, and OIMNets equipped with BatchNorm and ProtoNorm. The variants are trained using the OIM loss. Low similarity scores indicate that the features in the LUT or the circular queue for different IDs encode different information, suggesting a strong inter-class separability. We can observe that employing BatchNorm offers better results in terms of the inter-class separability, compared to the vanilla model, which also demonstrates the importance of calibrating the feature distribution prior to L2 normalizations. We can also see that ProtoNorm obtaining feature statistics less biased towards dominant IDs provides lower similarity scores than BatchNorm, even when trained with a small number of epochs, encouraging more inter-class separability. Moreover, the average distances do not deviate from the initial point severely with ProtoNorm, suggesting that ProtoNorm also stabilizes training process.

Qualitative analysis. We provide in Fig. 6 the visual comparisons between retrieval results for OIMNet [36] and OIMNet++ on PRW [42] and CUHK-SYSU [36]. We can see that OIMNet++ provides person representations that capture subtle discriminative cues, *e.g.*, hair and glasses (first row), as ProtoNorm in OIMNet++ enhances the inter-class separability. We can also observe the effectiveness of the LOIM loss. For example, OIMNet++ is more robust to occlusions (second and third row) and person overlaps (fourth row), since the LOIM loss favors pedestrian proposals with better localization accuracies to train with features in the LUT.



Fig. 6: Qualitative comparison between OIMNet [36] and OIMNet++. For each query image (left), we visualize top-1 search results, where red and green boxes indicate failure and correct cases, respectively. The first two rows are from PRW [42], and the remaining ones are from CUHK-SYSU [36]. For each image, we magnify the person-of-interest at the top-left corner for a better visualization.

5 Conclusion

We have introduced OIMNet++ for person search that addresses the limitations of existing methods. To this end, we have presented a novel normalization scheme, dubbed ProtoNorm, that provides better statistics for feature standardization, even under the extreme class imbalance across person IDs. We have also introduced the LOIM loss that exploits the localization accuracy of each proposal to learn more discriminative representations. Finally, we have demonstrated the effectiveness of each component with extensive ablation studies, and have shown that OIMNet++ outperforms other person search methods on the standard person search benchmarks by a large margin.

Acknowledgements. This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2022-00143524, Development of Fundamental Technology and Integrated Solution for Next-Generation Automatic Artificial Intelligence System, and No.2021-0-02068, Artificial Intelligence Innovation Hub), the Yonsei Signature Research Cluster Program of 2022 (2022-22-0002), and the KIST Institutional Program (Project No.2E31051-21-203).

References

1. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016) [4](#)
2. Chen, D., Zhang, S., Ouyang, W., Yang, J., Tai, Y.: Person search via a mask-guided two-stream CNN model. In: ECCV (2018) [4](#), [9](#), [11](#), [12](#)
3. Chen, D., Zhang, S., Yang, J., Schiele, B.: Norm-aware embedding for efficient person search. In: CVPR (2020) [2](#), [4](#), [5](#), [9](#), [10](#), [11](#), [12](#)
4. Choi, S., Kim, T., Jeong, M., Park, H., Kim, C.: Meta batch-instance normalization for generalizable person re-identification. In: CVPR (2021) [5](#)
5. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: ICCV (2017) [11](#)
6. De Montjoye, Y.A., Hidalgo, C.A., Verleysen, M., Blondel, V.D.: Unique in the crowd: The privacy bounds of human mobility. Scientific reports **3**(1), 1–5 (2013) [2](#), [10](#)
7. Dong, W., Zhang, Z., Song, C., Tan, T.: Bi-directional interaction network for person search. In: CVPR (2020) [2](#), [4](#), [5](#), [9](#), [11](#)
8. Dong, W., Zhang, Z., Song, C., Tan, T.: Instance guided proposal network for person search. In: CVPR (2020) [4](#), [11](#), [12](#)
9. Fang, H.S., Xie, S., Tai, Y.W., Lu, C.: RMPE: Regional multi-person pose estimation. In: ICCV (2017) [9](#)
10. Han, C., Ye, J., Zhong, Y., Tan, X., Zhang, C., Gao, C., Sang, N.: Re-id driven localization refinement for person search. In: CVPR (2019) [4](#), [11](#), [12](#)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV (2017) [10](#)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) [4](#), [9](#), [11](#)
13. Ioffe, S.: Batch renormalization: Towards reducing minibatch dependence in batch-normalized models. NeurIPS (2017) [4](#)
14. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML (2015) [2](#), [4](#), [7](#), [12](#)
15. Jin, X., Lan, C., Zeng, W., Chen, Z., Zhang, L.: Style normalization and restitution for generalizable person re-identification. In: CVPR (2020) [5](#)
16. Kim, H., Joung, S., Kim, I.J., Sohn, K.: Prototype-guided saliency feature learning for person search. In: CVPR (2021) [2](#), [4](#), [5](#), [9](#), [11](#), [12](#)
17. Lan, X., Zhu, X., Gong, S.: Person search by multi-scale matching. In: ECCV (2018) [4](#)
18. Li, W., Zhao, R., Xiao, T., Wang, X.: DeepReid: Deep filter pairing neural network for person re-identification. In: CVPR (2014) [1](#)
19. Li, X., Sun, W., Wu, T.: Attentive normalization. In: ECCV (2020) [4](#)
20. Li, Y., Qi, H., Dai, J., Ji, X., Wei, Y.: Fully convolutional instance-aware semantic segmentation. In: CVPR (2017) [4](#), [9](#)
21. Li, Z., Miao, D.: Sequential end-to-end network for efficient person search. In: AAAI (2021) [12](#)
22. Liao, S., Hu, Y., Zhu, X., Li, S.Z.: Person re-identification by local maximal occurrence representation and metric learning. In: CVPR (2015) [1](#)
23. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017) [10](#)
24. Liu, H., Feng, J., Jie, Z., Jayashree, K., Zhao, B., Qi, M., Jiang, J., Yan, S.: Neural person search machines. In: ICCV (2017) [11](#)

25. Luo, H., Gu, Y., Liao, X., Lai, S., Jiang, W.: Bag of tricks and a strong baseline for deep person re-identification. In: CVPR Workshops (2019) [4](#), [8](#)
26. Munjal, B., Amin, S., Tombari, F., Galasso, F.: Query-guided end-to-end person search. In: CVPR (2019) [11](#)
27. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017) [10](#)
28. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. NeurIPS (2015) [2](#)
29. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. IJCV (2015) [9](#)
30. Shao, W., Meng, T., Li, J., Zhang, R., Li, Y., Wang, X., Luo, P.: SSN: Learning sparse switchable normalization via sparsestmax. In: CVPR (2019) [4](#)
31. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR (2016) [4](#)
32. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022 (2016) [4](#), [5](#)
33. Wang, C., Ma, B., Chang, H., Shan, S., Chen, X.: TCTS: A task-consistent two-stage framework for person search. In: CVPR (2020) [11](#), [12](#)
34. Wang, G., Peng, J., Luo, P., Wang, X., Lin, L.: Batch kalman normalization: Towards training deep neural networks with micro-batches. arXiv preprint arXiv:1802.03133 (2018) [4](#)
35. Wu, Y., He, K.: Group normalization. In: ECCV (2018) [4](#)
36. Xiao, T., Li, S., Wang, B., Lin, L., Wang, X.: Joint detection and identification feature learning for person search. In: CVPR (2017) [1](#), [2](#), [3](#), [5](#), [6](#), [7](#), [9](#), [10](#), [11](#), [12](#), [13](#), [14](#)
37. Yan, Y., Li, J., Qin, J., Bai, S., Liao, S., Liu, L., Zhu, F., Shao, L.: Anchor-free person search. In: CVPR (2021) [2](#), [4](#), [11](#), [12](#)
38. Yao, Z., Cao, Y., Zheng, S., Huang, G., Lin, S.: Cross-iteration batch normalization. In: CVPR (2021) [4](#)
39. Ye, M., Shen, J., Lin, G., Xiang, T., Shao, L., Hoi, S.C.: Deep learning for person re-identification: A survey and outlook. IEEE TPAMI (2021) [1](#)
40. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: ICCV (2015) [1](#)
41. Zheng, L., Yang, Y., Hauptmann, A.G.: Person re-identification: Past, present and future. arXiv preprint arXiv:1610.02984 (2016) [1](#)
42. Zheng, L., Zhang, H., Sun, S., Chandraker, M., Yang, Y., Tian, Q.: Person re-identification in the wild. In: CVPR (2017) [1](#), [3](#), [7](#), [10](#), [11](#), [12](#), [13](#), [14](#)
43. Zhong, Y., Wang, X., Zhang, S.: Robust partial matching for person search in the wild. In: CVPR (2020) [5](#), [9](#)
44. Zhou, K., Yang, Y., Cavallaro, A., Xiang, T.: Omni-scale feature learning for person re-identification. In: ICCV (2019) [5](#)
45. Zhuang, Z., Wei, L., Xie, L., Zhang, T., Zhang, H., Wu, H., Ai, H., Tian, Q.: Rethinking the distribution gap of person re-identification with camera-based batch normalization. In: ECCV (2020) [5](#)