Meta Pairwise Relationship Distillation for Unsupervised Person Re-identification

Haoxuanye Ji[1]   Le Wang[1*]   Sanping Zhou[1]   Wei Tang[2]   Nanning Zheng[1]   Gang Hua[3]
[1]Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
[2]University of Illinois at Chicago   [3]Wormpex AI Research

Abstract

Unsupervised person re-identification (Re-ID) remains challenging due to the lack of ground-truth labels. Existing methods often rely on estimated pseudo labels via iterative clustering and classification, and they are unfortunately highly susceptible to performance penalties incurred by the inaccurate estimated number of clusters. Alternatively, we propose the Meta Pairwise Relationship Distillation (MPRD) method to estimate the pseudo labels of sample pairs for unsupervised person Re-ID. Specifically, it consists of a Convolutional Neural Network (CNN) and Graph Convolutional Network (GCN), in which the GCN estimates the pseudo labels of sample pairs based on the current features extracted by CNN, and the CNN learns better features by involving high-fidelity positive and negative sample pairs imposed by GCN. To achieve this goal, a small amount of labeled samples are used to guide GCN training, which can distill meta knowledge to judge the difference in the neighborhood structure between positive and negative sample pairs. Extensive experiments on Market-1501, DukeMTMC-reID and MSMT17 datasets show that our method outperforms the state-of-the-art approaches.

1. Introduction

Given a query pedestrian image, person re-identification (Re-ID) aims to match it with target pedestrian images of the same identity. It remains challenging due to the large appearance variations caused by different viewing angles, light conditions and background clutters in disjoint scenes. Existing methods usually learn discriminative features in a supervised manner [39, 35, 2, 1, 25], which requires extensive manual labeling efforts. Due to the prohibitively high cost of such annotation, training person Re-ID systems in the unsupervised manner has become a popular and practical research topic.
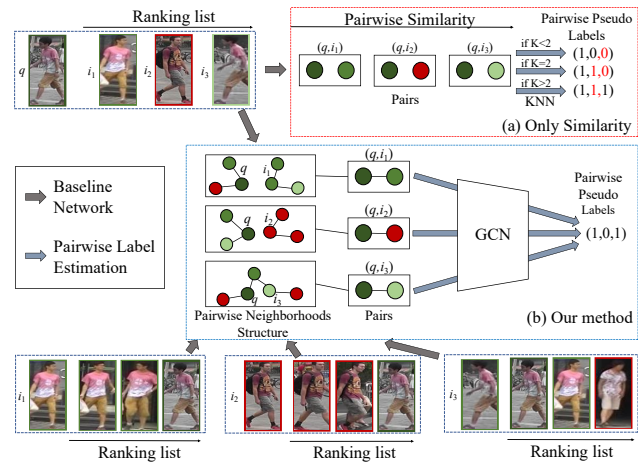

Figure 1. Illustrations of two pseudo label estimation methods, in which (a) the traditional method directly take the pairwise similarity to estimate pseudo labels, while (b) our method takes the pairwise neighborhood structures to estimate pseudo labels. Each circle denotes an individual image. The green circles represent the same identity as the query image, dark color indicates high visual similarity, while red circles represent other identities.

Recent unsupervised person Re-ID methods [13, 14, 6] attempted to learn discriminative feature embeddings from unlabeled training data based on iterative clustering and classification. However, it is often nontrivial to determine the number of clusters, and mishaps that wrongly estimate the cluster numbers often incurs excessive noise in the pseudo labels.

To address these issues, we reformulate the unsupervised discriminative feature learning as a pairwise relationship estimation problem. In this paper, we use the term *positive pair* to denote a pair of the pedestrian images of the same perceived identity; and conversely, *negative pair* to denote images with different perceived identities. In the embedding specified by a GCN, positive pairs are *pulled* closer; while negative pairs are *pushed* away from one another. With this *soft* semantic preserving rule replacing the clustering

algorithm, the dilemma of determining cluster numbers are circumvented. In the unsupervised learning paradigm, we will need to differentiate such positive pairs and negative pairs without relying on human annotations. One intuitive solution is thresholding visual similarity scores as the criterion, *i.e.*, considering two images with high visual similarity as a positive pair, and vice versa. However, as with many other thresholding based techniques, this criterion is unreliable in practice. For example, as shown in Figure 1 (a), pair (q, i_2) is higher in visual similarity score than pair (q, i_3), contradicting the ground-truth. Alternatively, we argue that a graph structure is more suitable to estimate pairwise labels, as shown in Figure 1 (b), which exploits contextual information to deduce the correct pairwise pseudo label for (q, i_2).

In this paper, we propose the Meta Pairwise Relationship Distillation (MPRD) method for unsupervised person Re-ID. It comprises a Convolutional Neural Network (CNN) and Graph Convolutional Network (GCN), where the GCN estimates the pseudo labels of sample pairs via the meta knowledge learned from small amount of labeled samples, and the CNN learns the discriminative features from input images according to the estimated pseudo labels.

Specifically, the CNN and GCN are trained in an alternating manner, which iteratively and respectively refines its per-image feature and pairwise pseudo labels. At each iteration, the CNN extracts the current per-image feature, and updates the feature memory by a linear combination of it and the previous features. Afterwards, the pairwise neighborhood structure is estimated by connecting every image with its neighbors, according to the visual similarity metric. The resulting graph structure is then fed into the GCN to infer the pseudo label for sample pairs. Empirically, we found that it is very hard to train the GCN without any supervision, therefore, we exploit a small amount of labeled metadata to explicitly supervise GCN, which greatly helps its robustness.

The GCN is only leveraged to provide pseudo supervision to the CNN training, and it is excluded in the testing stage. We evaluate our proposed method on Market-1501 [34], DukeMTMC-reID [20], and MSMT17 [26] datasets.

In summary, the contributions of this paper are summarized as follows.

1. We reformulate the unsupervised discriminative feature learning task as a pairwise relationship estimation problem, which avoids the error-prone step of estimating the number of clusters in most existing methods.
2. We propose the MPRD method for unsupervised person Re-ID, which incorporates a dedicated GCN as the pairwise pseudo label generator in the training stage and it iteratively refines its estimated labels with better CNN features.
3. We design an effective GCN that generates high-fidelity pseudo labels based on the pairwise neighborhood structures.

2. Related Work

Supervised Person Re-identification methods require labor-intensive labeled images during their training process. Early methods usually extract a global feature representation per image for image retrieval [28, 18, 10]. In PersonNet [28], a small-scale convolutional filter captures the fine-grained cues. By combining such cues and automatically determined scale weights, multi-scale discriminative features are learned in [18]. SPRe-ID [10] employs a human semantic parsing technique to capture the pixel-level discriminative clues. When the background is cluttered or the pedestrian is occluded, part-level features are shown to boost performance with the mining of discriminative body regions [22, 19, 42, 5, 41]. Attention and multi-loss are also used to enhance representation learning from a multi-view perspective [29, 33, 4, 21, 40].

Unsupervised Person Re-identification methods relieve the requirement for the cost-prohibitive annotations, which include hand-crafted feature based methods [12, 34], unsupervised domain adaptation methods [7, 36, 37, 16, 9, 11, 3, 31, 43] and fully unsupervised methods [13, 6, 27, 15, 24, 14]. It is very challenging to hand-craft robust features to handle the appearance variations incurred by different camera models, varying illuminations and viewpoints.

Methods based on unsupervised domain adaptation utilize prior knowledge on a source dataset with labels, and attempt to generalize on another unlabeled target dataset. HHL [36] enforces camera invariance and domain connectedness to improve the generalization. ECN [37] introduces an exemplar memory to store features of the target domain and accommodate examplar-invariance, camera-invariance, and neighborhood-invariance of the target domain properties. SSG [7] exploits the potential similarity (from the global body and local parts) of unlabeled samples to automatically build multiple clusters from different views. Mekhazni *et.al*. [16] design the Dissimilarity-based Maximum Mean Discrepancy loss to bridge the domain gap. ADTC [9] uses an unsupervised voxel attention and a two-stage clustering strategy to to account for the variations in images.

Some fully unsupervised methods are guided by pseudo supervision obtained from clustering results on the embeddings [13, 6, 14]. SSLR [15] replaces the hard one-hot label with soft labels to alleviate the error caused by unsupervised clustering. MLCR [24] predicts a "multi-label" for each training sample through Memory-based Positive Label Prediction (MPLP) and learns discriminative features via the Memory-based Multi-label classification loss. With the intrinsic "tracklet" structure and appearance, TSSL [27] eliminates the necessity of both pedestrian identity and camera labels.
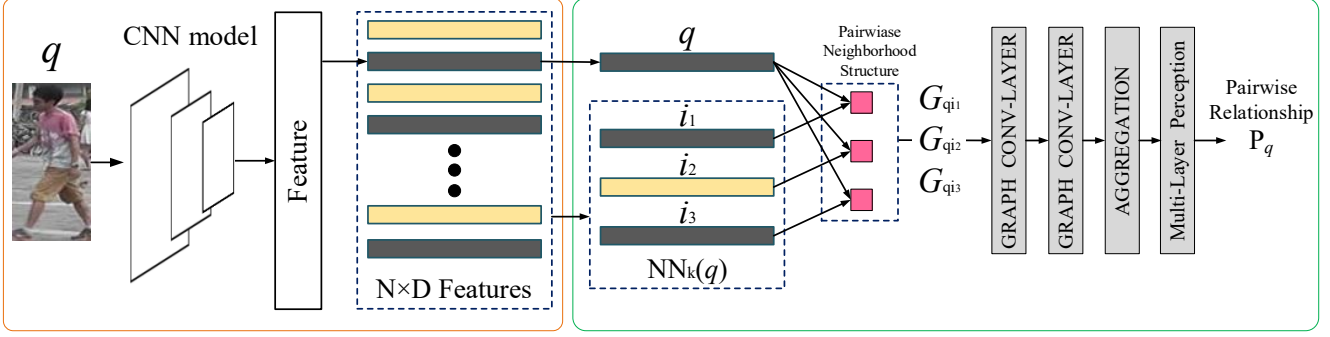
Figure 2. Overview of MPRD. An initialized backbone network extracts the feature of the training image. Then GCN infers the pairwise relationship between the features and their neighbors, which is used to train the CNN model.

The most relevant existing method is MLCR [24], which reformulates the unsupervised person Re-ID task as a multi-label classification problem. However, we argue that our MPRD differs from MLCR in two aspects. First, we reformulate the task as a pairwise relationship estimation problem; second, we design an effective GCN model to provide high-fidelity pseudo labels. The ablation study in Section 5.3 verifies the MPRD's performance advantage over MLCR.

3. Meta Pairwise Relationship Distillation

Given an unlabeled dataset $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$, where x_i denotes the ith input image, and N denotes the number of training samples, the MPRD estimates the pairwise pseudo labels for feature learning. As illustrated in Figure 2, the CNN learns discriminative features supervised by the pairwise pseudo labels generated by GCN; while the GCN estimates the pairwise pseudo labels based on CNN features. This interdependency is practically solved via alternating optimization of the GCN and the CNN.

3.1. CNN

Network backbone. The CNN module extracts discriminative features, which allows nearest neighbor search in the feature space. For simplicity, we adopt the backbone network in [8] as our CNN choice*, which consists of a feature extraction module and a feature memory module. In practice, the feature extraction module $\mathcal{F}$ extracts a d-dimensional feature $\mathcal{F}(\mathbf{x}_i)$ from each input image $\mathbf{x}_i$, and then ℓ_2-normalized by $\tilde{\mathcal{F}}(\mathbf{x}_i) \leftarrow \mathcal{F}(\mathbf{x}_i)/\|\mathcal{F}(\mathbf{x}_i)\|_2$, $\|\mathcal{F}(\mathbf{x}_i)\|_2$ indicates the norm of $\mathcal{F}(\mathbf{x}_i)$, the feature memory $\mathcal{M}$ stores all the features of training images. The feature memory is updated at the tth iteration as follows.

$$\begin{aligned} \mathcal{M}^{(t)}[i] &\leftarrow \gamma^{(t)}\tilde{\mathcal{F}}(\mathbf{x}_i) + (1-\gamma^{(t)})\tilde{\mathcal{M}}^{(t-1)}[i], \\ \tilde{\mathcal{M}}^{(t)}[i] &\leftarrow \mathcal{M}^{(t)}[i]/\|\mathcal{M}^{(t)}[i]\|_2, \end{aligned} \quad (1)$$

---

*Our method is compatible with various network backbones.

where $\gamma^{(t)}$ denotes an iteration-dependent updating rate. This feature memory mechanism practically implements a smoothing operation over the iterations, potentially reducing violent oscillations in features.

Loss function. Suppose the pairwise pseudo labels are provided by GCN, we introduce the Binomial Deviance (BD) loss [30] function $L_{\mathcal{F}}$ to train the CNN, which aims to minimize the distance in positive pairs and to maximize the distance in negative pairs.

$$\begin{aligned} L_{\mathcal{F}} = \frac{1}{n}\sum_{i=1}^{n} \frac{\delta}{|\mathbf{P}_i|} \sum_{\mathbf{x}_j \in \mathbf{P}_i} \zeta(\alpha(\lambda_1 - \langle \tilde{\mathcal{F}}(\mathbf{x}_i), \tilde{\mathcal{M}}[j]\rangle)) \\ + \frac{1}{|\mathbf{N}_i^*|} \sum_{\mathbf{x}_j \in \mathbf{N}_i^*} \zeta(\alpha(\langle \tilde{\mathcal{F}}(\mathbf{x}_i), \tilde{\mathcal{M}}[j]\rangle - \lambda_2)), \end{aligned} \quad (2)$$

where $\langle \cdot, \cdot \rangle$ represent inner product, since both $\tilde{F}(\mathbf{x}_i)$ and $\tilde{\mathcal{M}}[j]$ are ℓ_2-normalized unit vector, $\langle \tilde{\mathcal{F}}(\mathbf{x}_i), \tilde{\mathcal{M}}[j]\rangle$ denote the cosine similarity between them, $\zeta(x) = \log(1 + e^x)$, n is the batch size, $|\cdot|$ denotes the cardinality (number of elements), δ indicates the importance of positive pairs against negative pairs, λ_1 and λ_2 denote two margin parameters, and α is an amplification factor. Besides, $\mathbf{P}_i$ and $\mathbf{N}_i$ represent the positive pair list and the negative pair list, respectively. As $|\mathbf{N}_i| \gg |\mathbf{P}_i|$ in practice, we further focus on the hard negative pair list $\mathbf{N}_i^*$ with fixed size r as follows.

$$\mathbf{N}_i^* = \{\mathbf{x}_j \mid \mathbf{x}_j \in \text{top}(\langle \tilde{\mathcal{F}}(\mathbf{x}_i), \tilde{\mathcal{M}}[j]\rangle, r), \ \mathbf{x}_j \in \mathbf{N}_i\}, \quad (3)$$

where $\text{top}(\cdot, r)$ represent the r largest samples. Therefore, $\mathbf{N}_i^*$ contains the r closest negative samples to the query $\mathbf{x}_i$ in the embedding feature space.

After training the CNN, all positive pairs concentrate within a radius of λ_1; while all negative pairs locate elsewhere with a distance of at least λ_2. Afterwards, a nearest neighbor searching algorithm can be applied to solve the person Re-ID problem.
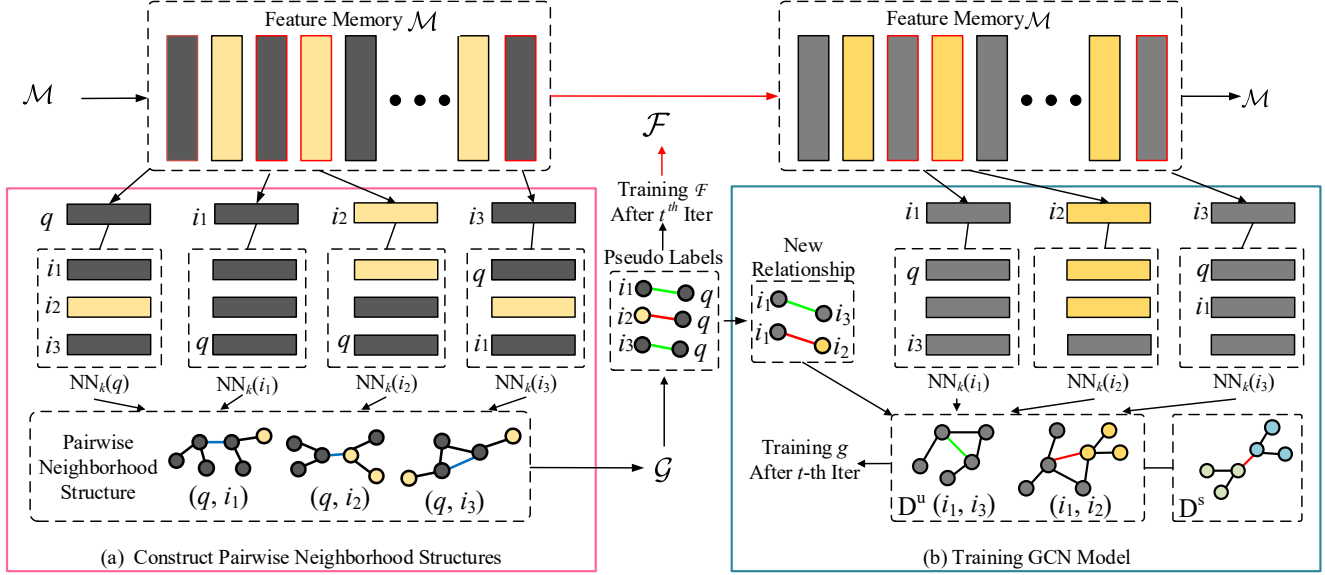
Figure 3. Illustration of the pairwise neighborhood structure and the training strategy of MPRD, where the red arrow means prohibit execution until training the CNN model after t^{th} iteration, the blue line means the pairs' relationship is not yet judged, the green line means the pair is considered to have the positive label, and the red line indicates negative label.

3.2. GCN

Network backbone. The GCN estimates the pseudo labels of sample pairs, so as to guide the CNN training with unlabeled data. As shown in Figure 2 (b), it takes the pairwise neighborhood structures $\mathbf{G}_{ij} = (\mathbf{A}_{ij}, \mathbf{V}_{ij})$ as inputs, where $\mathbf{A}_{ij}$ denotes the adjacent matrix, and $\mathbf{V}_{ij}$ indicates the node embedding. For image $\mathbf{x}_i$ and each images $\mathbf{x}_j$ in $\mathrm{NN}_k(\mathbf{x}_i) = \mathrm{top}(\langle \tilde{\mathcal{M}}[i], \tilde{\mathcal{M}}[j] \rangle, k)$, the pairwise neighborhood structure can be constructed by connecting images $\mathbf{x}_i$ and $\mathbf{x}_j$ with their neighbors. Therefore, the adjacent matrix of $\mathbf{G}_{ij}$ can be defined as follows:

$$\mathbf{A}_{ij}(b, a) = \mathbf{A}_{ij}(a, b) = \begin{cases} 1, & x_a \in \mathrm{NN}_k(\mathbf{x}_b) \\ 0, & otherwise \end{cases}, \quad (4)$$

where $b \in \{i, j\}$ denotes an image index in the extracted sample pair (x_i, x_j). Besides, the node embedding of $\mathbf{G}_{ij}$ can be achieved in two steps as follows. **(1)** We use the Double-Radius Node Labeling (DRNL) [32] to generate the position embedding of each node in $\mathbf{G}_{ij}$, which can distinguish nodes with different positions relative to sample pair $(\mathbf{x}_i, \mathbf{x}_j)$. **(2)** We concatenate the position embedding and the feature embedding of the nodes in $\mathbf{G}_{ij}$ as $\mathbf{V}_{ij}$.

The structure of our GCN is shown in Figure 2 (b), which consists of two graph convolutional layers, one graph aggregation layer and one multi-layer perception. In particular, the multi-layer perception (with its parameters denoted as Θ_m) contains two fully-connected layers, the graph aggregation layer (with its parameters denoted as Θ_a) includes a max-pooling layer and a 1-D convolutional layer. The recursive

function of our graph convolutional layers is,

$$\mathbf{Y}_{ij}^{(l+1)} = \sigma(\mathbf{D}^{-1}(\mathbf{A}_{ij} + \mathbf{I})\mathbf{Y}_{ij}^{(l)}\Theta_{g^{(l)}}), \quad \mathbf{Y}_{ij}^{(0)} = \mathbf{V}_{ij}, \quad (5)$$

where $\Theta_{g^{(l)}}$ indicates the parameters of the lth layer, $\mathbf{D}$ is the Laplacian matrix of $\mathbf{G}_{ij}$, σ denotes ReLU as the activation function, and $\mathbf{Y}_{ij}^{(l)}$ means the node-level embedding of the lth layer. In the training process, the graph convolutional layers extract features from the pairwise neighborhood structures, the graph aggregation layer aggregates the node-level features into the graph-level features, and the multi-layer perception estimates the pseudo labels of sample pairs.

Loss function. Our GCN takes the pairwise neighborhood structure as input, and outputs the likelihood of $\mathbf{x}_i$ and $\mathbf{x}_j$ being of the same identity. Let $\mathcal{G}$ denotes the mapping function of our GCN, whose parameters are $\Theta_{\mathcal{G}} = \{\Theta_{g^{(1)}}, \cdots, \Theta_{g^{(L)}}, \Theta_a, \Theta_m\}$. To obtain this mapping function, we apply the Binary Cross Entropy (BCE) loss to supervise the training process:

$$\mathrm{L}_{\mathcal{G}} = \frac{-1}{n'} \sum_{i=1}^{n'} \frac{1}{|\mathbf{P}_g^i|} \sum_{\mathbf{G}_{ij} \in \mathbf{P}_g^i} \log(g_{ij}) + \\ \frac{1}{|\mathbf{N}_g^i|} \sum_{\mathbf{G}_{ij} \in \mathbf{N}_g^i} (1 - \log(g_{ij})), \quad (6)$$

where g_{ij} denotes the prediction of $\mathbf{G}_{ij}$, n' is the batch size, $\mathbf{P}_g^i$ is the set of positive samples, in which the sample $\mathbf{G}_{is}$ in $\mathbf{P}_g^i$ has the positive sample pair (x_i, x_s), and $\mathbf{N}_g^i$ is the

Algorithm 1: Training MPRD.

Input: Initial $\mathcal{F}$, Initial $\mathcal{G}$, Unlabeled data $\mathbf{X}$,
Labeled data $\mathbf{Z}$, Feature memory $\mathcal{M}$,
Training epoch T.

Output: Best CNN model $\mathcal{F}$.

1 Initial $\mathbf{P} = \{\mathbf{P}_i = \{i\}|1 < i < N\}$;
2 for $t = 1, t \leq T, t{+}{+}$ **do**
3   **for** *each* $\mathbf{x}_i$ *in* $\mathbf{X}$ **do**
4   Randomly select $\mathbf{z}_j$ in $\mathbf{Z}$;
5   Generate pairwise neighborhood structures
       and its labels for $\mathcal{G}$;
6   Train $\mathcal{G}$ with parameters $\Theta_\mathcal{G}$ by Eq. (9);
7   **end**
8   **for** *each* $\mathbf{x}_i$ *in* $\mathbf{X}$ **do**
9   Update $\mathcal{M}$ by Eq. (1);
10   Update $\mathbf{P}_i$ by Eq. (7);
11   Train $\mathcal{F}$ with parameters $\Theta_\mathcal{F}$ by Eq. (8);
12   **end**
13 end

---

negative ones. In practice, we obtain the labels of $\mathbf{G}_{ij}$ in two ways to train our GCN model, which will be described in Section 4.2.

4. Optimization

Our CNN and GCN are optimized in an alternating manner, with the overall procedure summarized in Algorithm 1.

4.1. Updating $\mathcal{F}$ with $\mathcal{G}$ Fixed

Labels for CNN. Since the ground-truth labels for CNN training are unavailable, we resort to the GCN for pairwise pseudo labels. For each image pair $(\mathbf{x}_i, \mathbf{x}_j)$, we construct the pairwise neighborhood structures for GCN in three steps, as shown in Figure 3 (a): **(1)** CNN extracts the current feature $\tilde{\mathcal{F}}(\mathbf{x}_i)$; **(2)** Memory feature $\tilde{\mathcal{M}}[i]$ is updated by Eq. (1); and **(3)** the pairwise neighborhood structure $\mathbf{G}_{ij}$ is obtained by connecting image $\mathbf{x}_i$ and the image $\mathbf{x}_j$ in $\mathrm{NN}_k(\mathbf{x}_i)$ with their neighbors. Afterwards, $\mathbf{G}_{ij}$ is fed into the GCN to predict pairwise label g_{ij}. To refine g_{ij}, we subsequently apply a binary filtering to improve the fidelity of $\mathbf{P}_i$ as follows.

$$\mathbf{P}_i = \{\mathbf{x}_j \mid \mathbf{x}_j \in \mathrm{NN}_k(\mathbf{x}_i), g_{ij} > \mu\}, \qquad (7)$$

where μ is a predefined likelihood threshold that ensures $\mathbf{x}_i$ and $\mathbf{x}_j$ are of the same identity. Concurrently, the hard negative sample list $\mathbf{N}_i^*$ is obtained according to Eq. (3).

Parameter update. We apply the Stochastic Gradient Descent (SGD) algorithm to update the parameters of our CNN model, which can be formulated as follows,

$$\Theta_\mathcal{F}^{(t)} = \Theta_\mathcal{F}^{(t-1)} - \beta_1 \frac{\partial \mathrm{L}_\mathcal{F}}{\partial \Theta_\mathcal{F}} \bigg|_{\Theta_\mathcal{F}^{(t-1)}}, \qquad (8)$$

where $\Theta_\mathcal{F}^{(t)}$ denotes the parameters of our CNN model at the tth iteration, and β_1 means the learning rate.

4.2. Updating $\mathcal{G}$ with $\mathcal{F}$ Fixed

Labels for GCN. To ensure the quality of the generated pseudo pairwise labels by GCN, we propose to generate an initial, accurate pairwise neighborhood structure with a tiny amount of labeled meta data $\mathbf{Z} = \{\mathbf{z}_m\}_{m=1}^M$, where M is the number of labeled identities (we set $M = 5$, which is approximately 0.5% of all data. The labeled data is reserved exclusively to *jump start* GCN training, and these annotations are never presented to CNN training.).

After the initial *jump start* phase, the GCN is primarily trained with the remaining 99.5% unlabeled data. The label generation process is summarized as follows.

1. At the first iteration, for each $\mathbf{x}_i$ in the unlabeled training set $\mathbf{X}$, we generate positive pairs with different data augmentation techniques, including random Gaussian blur and grayscale conversion, to get its perturbed image $\mathbf{x}_i'$. We assume pairs such as $(\mathbf{x}_i, \mathbf{x}_i')$ are positive pairs. At subsequent iterations, the cardinality of $\mathbf{P}_i$ gradually increases. If $|\mathbf{P}_i| > 1$, we first randomly draw a sample $\mathbf{x}_{r_i}$ from $\mathbf{P}_i$. For every $\mathbf{x}_{r_j} \in \mathbf{P}_i$ where $r_i \neq r_j$, we assign $(\mathbf{x}_{r_i}, \mathbf{x}_{r_j})$ with a pseudo positive pair label if and only if $\mathbf{x}_{r_j} \in \mathrm{NN}_k(\mathbf{x}_{r_i})$.
   Concurrently at the first iteration, we randomly draw different images to form negative pairs. At subsequent iterations, we randomly select $|\mathbf{P}_i|$ images from $|\mathbf{N}_i^*|$ to generate negative pairs.
2. For each image $\mathbf{z}_i$ in the labeled meta dataset $\mathbf{Z}$, we randomly draw two images of the same identity to generate one positive pair, and two images of different identities to generate one negative pair. Besides, each image $\mathbf{x}_j$ in $\mathbf{X}$ can be paired with $\mathbf{z}_i$ to generate one negative pair.

After obtaining the sample pairs with pseudo labels, we construct a pairwise neighborhood structure $\mathbf{G}_{ij}$, and insert it into $\mathbf{P}_g^i$ or $\mathbf{N}_g^i$, accordingly.

Parameter update. With the pairwise neighborhood structures and pseudo labels ready, we apply the SGD algorithm to update the parameters of GCN as follows.

$$\Theta_\mathcal{G}^{(t)} = \Theta_\mathcal{G}^{(t-1)} - \beta_2 \frac{\partial \mathrm{L}_\mathcal{G}}{\partial \Theta_\mathcal{G}} \bigg|_{\Theta_\mathcal{G}^{(t-1)}}, \qquad (9)$$

with β_2 being the learning rate. In practice, we select $n'/2$ and $n'/2$ images from the labeled meta data and unlabeled data, respectively. Then we construct pairwise neighborhood structures to calculate $\partial \mathrm{L}_\mathcal{G}^u / \partial \Theta_\mathcal{G}$ from unlabeled data and $\partial \mathrm{L}_\mathcal{G}^m / \partial \Theta_\mathcal{G}$ from labeled meta data, and compute a linear combination of them,

$$\frac{\partial \mathrm{L}_\mathcal{G}}{\partial \Theta_\mathcal{G}} = \frac{\partial \mathrm{L}_\mathcal{G}^u}{\partial \Theta_\mathcal{G}} + \eta^{(t)} \frac{\partial \mathrm{L}_\mathcal{G}^m}{\partial \Theta_\mathcal{G}}, \qquad (10)$$

where $\eta^{(t)}$ is an iteration-dependent weighting factor, which represents the ratio of the number of pairwise neighborhood structure generated by metadata and generated by generated by the unlabeled data in each mini-batch. It is monotonically decreases with iterations. At early stages of training, a larger $\eta^{(t)}$ value helps alleviate the influence of noisy labels in the unlabeled data. At later iteration stages, the number of pairwise neighborhood structures with pseudo labels extracted from the unlabeled portion increases, hence $\eta^{(t)}$ is set to be smaller to match this trend.

5. Experiments

5.1. Datasets and Evaluation Protocol

Datasets. We evaluate our method on three standard large-scale Person Re-ID datasets, including Market-1501 [34], DukeMTMC-reID [20], and MSMT17 [26]. Market-1501 consists of 32,668 images of 1,501 identities captured by 6 cameras, in which the training set comprises 12,936 images of 751 identities, and the test data set comprises 19,732 images of 750 identities. DukeMTMC-reID consists of 36,411 images of 1,812 identities captured by 8 cameras, where the training dataset includes 16,522 images of 702 identities. MSMT17 is the largest Person Re-ID dataset, it includes 126,411 images of 4,101 identities captured by 15 cameras, and its training set has 32,621 images of 1,041 identities, while its test dataset has 93,820 bounding boxes of 3,060 identities.

Evaluation Protocol. Following [34, 20, 26], we evaluate performance with the retrieval precision metric *Cumulative Matching Characteristic (CMC) scores*, and the recall metric *Mean Average Precision (mAP)*.

5.2. Implementation Details

We implement our method in PyTorch [17] with a single NVIDIA GeForce GTX 1080Ti GPU. For the CNN part, we adopt ResNet50 [8], with the layers after pooling-5 removed, and a batch normalization layer appended. For an input image, $\mathcal{F}$ produces a 2048-dimensional feature. Similar to [24, 11], for the input image we use CamStyle [38] for data augmentation and resize it at 256×128, and then preprocess it with random crop, random rotation, random color jitter, and random erasing. For the perturbed image, we add random grayscale conversion and random Gaussian blur.

In an alternating manner, we train the CNN $\mathcal{F}$ and the GCN $\mathcal{G}$ using SGD with a 0.9 momentum. The number of training epochs is set to 40. For $\mathcal{F}$, the initial learning rate for the ResNet50 backbone is set to 0.01, and 0.1 for all other layers. The learning rate is reduced by a factor of 10 after 20 epochs, and the training mini-batch size is 32. For $\mathcal{G}$, the initial learning rate is 0.001, the likelihood threshold μ is 0.5, and the mini-batch size of 32. For the feature memory $\mathcal{M}$, the updating rate $\gamma^{(t)}$ starts at 1 at the first epoch and

Methods	Market-1501		DukeMTMC-reID	
	Rank-1	mAP	Rank-1	mAP
Super.	87.0	68.5	75.6	56.7
KNN	72.7	35.2	59.4	32.9
SS	72.8	39.9	60.0	34.3
MPLP*	80.0	44.5	64.6	39.8
MPRD w/o	75.8	43.1	61.1	34.9
MPRD #	73.0	39.3	57.4	36.7
MPRD	**83.0**	**51.1**	**67.4**	**43.7**
Single	46.1	15.5	38.3	14.6

Table 1. Performance with different pseudo label generation methods. "Super." and "Single" are baselines representing performance upper and lower bounds, respectively. All methods have incorporate the same Binomial deviance loss. The "*" mark in "MPLP*" indicates this implementation is based on released code from the authors. "MPRD#" and "MPRD w/o" denote ablated MPRD with CamStyle data augmentation removed and its GCN trained without unlabeled data, respectively.

linearly decreases to 0.5 at the 40th epoch. In the binomial deviance loss, the weight δ is fixed at 5 and r is the number of 1% negative samples as in [24]. Moreover, we set $\alpha = 4.0$ and $\lambda_1 = \lambda_2 = 0.2$. The small amount of labeled meta data involves five labeled identities that are randomly selected from the training data. Specifically, the small amount of labeled metadata is only used for training $\mathcal{G}$. The value of k is the maximum between 8 and the number of images whose cosine similarity to the input image is larger than 0.6.

5.3. Ablation Study

Effectiveness of MPRD. We compare MPRD against other pseudo label generation methods, including the KNN search, cosine similarity score (denoted as "SS"), and selection by MPLP. SS selects positive samples with a similarity threshold. MPLP is proposed in [24], which predicts pseudo-labels with high accuracy via similarity scores and cycle consistency. For KNN, we empirically set $K = 8$, where its performance peaks; for SS, we set the similarity threshold at 0.6; for MPLP, we incorporate it with the binomial deviance loss based on its released code by the authors. Under the same setting (Section 5.2), we also conduct 2 baseline experiments, *i.e.*, fully supervised re-ID with ground-truth (denoted as "Super." in Table 1); erroneously supervised re-ID with image index as labeles (denoted as "Single" in Table 1), which serve as the performance upper bound and lower bound, respectively. Additionally, two ablated variants of MPRD are compared, *i.e.*, MPRD with CamStyle data augmentation removed ("MPRD#"), and MPRD with its GCN trained purely on labeled meta data (*i.e.*, without vast majority of the unlabeled data, denoted as "MPRD w/o" in Table 1).

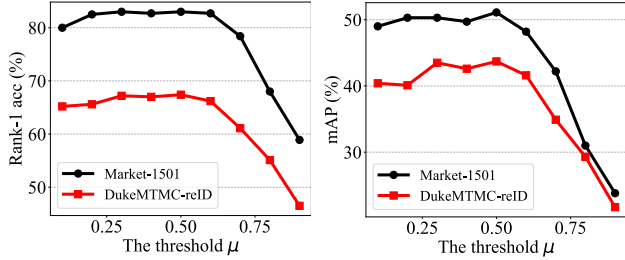According to Table 1, SS outperforms KNN, which could

Figure 4. Evaluation of different likelihood threshold μ in GCN.

Methods	market-1501		DukeMTMC-reID	
	Rank-1	mAP	Rank-1	mAP
MLCR*	80.1	44.7	64.9	40.6
MLCR(+5id)*	80.2	45.0	65.3	40.9
MPRD(0id)	80.9	46.8	65.6	40.1
MPRD	**83.0**	**51.1**	**67.4**	**43.7**

Table 2. Ablation study of the effect of meta data. "MLCR(+5id)*" denotes a modified MLCR that have extra access to the same amount (5 identities) of meta data as extra supervision. "MPRD(0id)" represents an ablated version of our proposed MPRD with the labeled meta data-based *jump start* procedure completely removed.

be attributed to its more flexible pseudo-labels. After incorporating the GCN model, MPRD achieves dramatic performance advantages over SS, *i.e.*, with rank-1 accuracy increased from 72.8% to 83.0%, and mAP increased from 39.9% to 51.1% on Market-1501. Comparing "MPRD w/o" and MPRD, we also verify the necessarity of incorporating the vast majority of unlabeled data in training $\mathcal{G}$. On Market-1501, MPRD achieves 7.2 and 8 percentage points advantage over "MPRD w/o" in Rank-1 accuracy and mAP, respectively. A similar trend also appears on DukeMTMC-reID. Additionally, "MPRD#" achieves 73.0% rank-1 accuracy and 39.3% mAP without CamStyle. These results demonstrate the effectiveness of the proposed MPRD, and show that CamStyle boosts the performance.

Impact of the likelihood threshold μ. In Eq. (7), μ determines whether two images are of the same identity, and its sensitivity analysis is presented in Figure 4. Performance is analyzed with μ values from 0.1 to 0.9 at a step size of 0.1. We observe that both rank-1 accuracy and mAP metrics increase slowly and smoothly till approximately 0.5 and abruptly drop afterwards. We speculate that extreme μ values degrade performance, *i.e.*, too small μ values lead to many false positive pairs while too large μ values incur many false negative pairs. Based on these experiments, we empirically fix $\mu = 0.5$.

Effect of small amount of labeled meta data. Since our approach introduces a small amount of labeled meta data to *jump start* the training of GCN, we analyze its impacts on the competing MLCR method. For fair comparison, we let MLCR have access to the same amount of labeled meta data as extra supervision. We also compare the effect of different amounts of labeled meta data in different variants of MPRD.

We let MLCR have access to the same labeled meta data by replacing the pseudo labels with the ground-truth labels, whenever the input training data belong to the labeled meta dataset. In Table 2, the upper part shows that with the extra small amount of labeled meta data, "MLCR(+5id)*" marginally outperforms its original version MLCR*, possibly due to the portion of such labeled meta dataset is too small (approximately 0.5% of all data).

The bottom part of Table 2 compares MPRD variants

with different amounts of labeled meta dataset, where "5id" means five labeled identities (meta data is only used for training the GCN). If this labeled data-based *jump start* portion is completely removed, "MPRD(0id)" suffers from only a small performance degradation, and still outperforms the competing "MPLR*" in Table 1. When the amount of labeled identities is 5, we observe that both rank-1 accuracy and mAP increase on Market-1501 and DukeMTMC-reID.

5.4. Comparison with the State-of-the-Art

We evaluate the proposed MPRD on Market-1501 [34], DukeMTMC-reID [20] and MSMT17 [26] datasets. Although a small amount of labeled meta data are used to guide the training process of the GCN, our method also belongs to unsupervised Person Re-Identification because there are only few labeled data are used to train GCN, and training the feature extraction CNN module only uses unlabeled data. The proposed method is compared against the state-of-the-art unsupervised Person Re-ID methods: LOMO [12], BOW [34], BUC [13], DBC [6], and the recent TSSL [27], SSLR [15], MLCR [24], JVTC [11]. Table 3 and Table 4 summarize the comparison.

Table 3 shows the results of the proposed method and state-of-the-art methods on Market-1501 and DukeMTMC-reID. On Market-1501, our MPRD achieves 2.7% higher rank-1 accuracy and 5.6% higher mAP than MLCR. Compared with JVTC, our MPRD achieves 10.1% higher rank-1 accuracy and 9.3% higher mAP. On DukeMTMC-reID, our MPRD achieves 2.2% higher rank-1 accuracy and 3.5% higher mAP than MLCR. Compared with JVTC, our MPRD has a sight 0.2% lower rank-1 accuracy but achieves 1.5% higher mAP. We also conduct experiments on MSMT17, and the results are presented in Table 4. From the table, our MPRD achieves 37.7% rank-1 accuracy and 14.6% mAP.

Of all the competing algorithms, MLCR is the most relevant one to our proposed MPRD. As is verified in the above results, MPRD outperforms it on Market-1501, DukeMTMC-reID and MSMT17. We speculate that this performance advantage arise from the following aspects. Our proposed

Methods	Market-1501				DukeMTMC-reID			
	Rank-1	Rank-5	Rank-10	mAP	Rank-1	Rank-5	Rank-10	mAP
LOMO [12] (CVPR15)	27.2	41.6	49.1	8.0	12.3	21.3	26.6	4.8
BOW [34] (ICCV15)	35.8	52.4	60.3	14.8	17.1	28.8	34.9	8.3
BUC [13] (AAAI19)	66.2	79.6	84.5	38.3	47.4	62.6	68.4	27.5
DBC [6] (BMVC19)	69.2	83.0	87.8	41.3	51.5	64.6	70.1	30.0
TSSL [27] (AAAI20)	71.2	-	-	43.3	62.2	-	-	38.5
SSLR [15] (CVPR20)	71.7	83.8	87.4	37.8	52.5	63.5	68.9	28.6
MLCR [24] (CVPR20)	80.3	89.4	92.3	45.5	65.2	75.9	80.0	40.2
JVTC [11] (ECCV20)	72.9	84.2	88.7	41.8	**67.6**	78.0	81.6	42.2
MPRD	**83.0**	**91.3**	**93.6**	**51.1**	67.4	**78.7**	**81.8**	**43.7**

Table 3. Unsupervised person re-ID performance with state-of-the-art methods on Market-1501 and DukeMTMC-reID datasets.

Methods	MSMT17			
	Rank-1	Rank-5	Rank-10	mAP
MLCR [24]	35.4	44.8	49.8	11.2
JVTC [11]	**39.0**	50.9	56.8	**15.1**
MPRD	37.7	**51.3**	**57.1**	14.6

Table 4. Unsupervised person re-ID performance with state-of-the-art methods on MSMT17 dataset.

MPRD introduces the neighbor structure information between sample pairs via its GCN. Through iterative and alternating training, the GCN gradually learns and refines the distinctions in neighborhood structure between positive and negative sample pairs, and provides higher fidelity pseudo-supervision for the CNN training. The alternating, collaborative training of the GCN and the CNN could be responsible for the performance benefits.

5.5. Qualitative Results

To intuitively understand the effectiveness of MPRD, we visualize via t-SNE [23] the learned features on Market-1501 training set, without and with the GCN, as shown in Figure 5. By comparing the two sets of learned features side-by-side, after introducing the GCN, points of the same identity are pulled closer to each other, as shown in Example3 where yellow dots are more concentrated on the right. Challenging cases (Example1 and Example2) where points of different identities are embedded too close to each other without GCN are resolved with the introduction of GCN. With Example2, the magenta points, blue points, and cyan points are highly proximate to one another in the embedding space without GCN. On the contrary, they are well separated in the embedding space with GCN.

6. Conclusion

In this paper, we propose the MPRD method to address the unsupervised person Re-ID task. Unlike previous methods that estimate the pseudo labels through either iterative
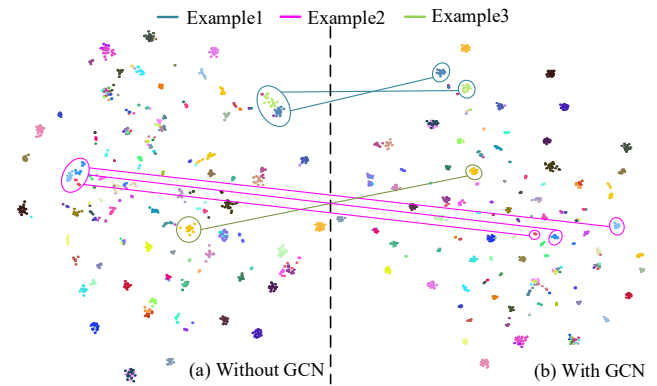

Figure 5. T-SNE visualization of learned features on 100 identities of Market-1501 training set. Points with the same color are of the same identity. The distribution of the features learned (a) without GCN, and (b) with the GCN.

clustering or classification, it is unnecessary for our method to determine the number of clusters in the training stage. The proposed MPRD reformulates the unsupervised discriminative feature learning task into a pairwise relationship estimation problem. A GCN is used to estimate the pairwise relationship of sample pairs based on the graph structure among the pairs' neighbors. CNN learns the discriminative features from input images according to these estimated pairwise relationship labels. Extensive experiments on Market-1501, DukeMTMC-reID, and MSMT17 datasets demonstrate the effectiveness of the proposed method for the unsupervised Person Re-ID task.

References

[1] Binghui Chen, Weihong Deng, and Jiani Hu. Mixed high-order attention network for person re-identification. In *ICCV*, pages 371–381, 2019. 1

[2] Guangyi Chen, Chunze Lin, Liangliang Ren, Jiwen Lu, and Jie Zhou. Self-critical attention learning for person re-identification. In *ICCV*, pages 9636–9645, 2019. 1

[3] Guangyi Chen, Yuhao Lu, Jiwen Lu, and Jie Zhou. Deep credible metric learning for unsupervised domain adaptation person re-identification. In *ECCV*, pages 643–659, 2020. 2

[4] Tianlong Chen, Shaojin Ding, Jingyi Xie, Ye Yuan, Wuyang Chen, Yang Yang, Zhou Ren, and Zhangyang Wang. Abd-net: Attentive but diverse person re-identification. In *ICCV*, pages 8350–8360, 2019. 2

[5] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, pages 1335–1344, 2016. 2

[6] Guodong Ding, Salman Khan, and Zhenmin Tang. Dispersion based clustering for unsupervised person re-identification. In *BMVC*, page 264, 2019. 1, 2, 7, 8

[7] Yang Fu, Yunchao Wei, Guanshuo Wang, Yuqian Zhou, Honghui Shi, and Thomas Huang. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *ICCV*, pages 6111–6120, 2019. 2

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3, 6

[9] Zilong Ji, Xiaolong Zou, Xiaohan Lin, Xiao Liu, Tiejun Huang, and Si Wu. An attention-driven two-stage clustering method for unsupervised person re-identification. In *ECCV*, pages 20–36, 2020. 2

[10] Mahdi M. Kalayeh, Emrah Basaran, Muhittin Gokmen, Mustafa E. Kamasak, and Mubarak Shah. Human semantic parsing for person re-identification. In *CVPR*, pages 1062–1071, 2018. 2

[11] Jianing Li and Shiliang Zhang. Joint visual and temporal consistency for unsupervised domain adaptive person re-identification. In *ECCV*, pages 483–499, 2020. 2, 6, 7, 8

[12] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, pages 2197–2206, 2015. 2, 7, 8

[13] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. A bottom-up clustering approach to unsupervised person re-identification. In *AAAI*, pages 8738–8745, 2019. 1, 2, 7, 8

[14] Yutian Lin, Yu Wu, Chenggang Yan, Mingliang Xu, and Yi Yang. Unsupervised person re-identification via cross-camera similarity exploration. *IEEE TIP*, 29:5481–5490, 2020. 1, 2

[15] Yutian Lin, Lingxi Xie, Yu Wu, Chenggang Yan, and Qi Tian. Unsupervised person re-identification via softened similarity learning. In *CVPR*, pages 3387–3396, 2020. 2, 7, 8

[16] Djebril Mekhazni, Amran Bhuiyan, George Ekladious, and Eric Granger. Unsupervised domain adaptation in the dis-similarity space for person re-identification. In *ECCV*, pages 159–174, 2020. 2

[17] Adam Paszke, S. Gross, Soumith Chintala, G. Chanan, E. Yang, Zachary Devito, Zeming Lin, Alban Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NeurIPS-W*, 2017. 6

[18] Xuelin Qian, Yanwei Fu, Yu-Gang Jiang, Tao Xiang, and Xiangyang Xue. Multi-scale deep learning architectures for person re-identification. In *ICCV*, pages 5409–5418, 2017. 2

[19] Ruijie Quan, Xuanyi Dong, Yu Wu, Linchao Zhu, and Yi Yang. Auto-reid: Searching for a part-aware convnet for person re-identification. In *ICCV*, pages 3749–3758, 2019. 2

[20] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, pages 17–35, 2016. 2, 6, 7

[21] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Mask-guided contrastive attention model for person re-identification. In *CVPR*, pages 1179–1188, 2018. 2

[22] Yifan Sun, Qin Xu, Yali Li, Chi Zhang, Yikang Li, Shengjin Wang, and Jian Sun. Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification. In *CVPR*, pages 393–402, 2019. 2

[23] Laurens Van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11):2579–2605, 2008. 8

[24] Dongkai Wang and Shiliang Zhang. Unsupervised person re-identification via multi-label classification. In *CVPR*, pages 10978–10987, 2020. 2, 3, 6, 7, 8

[25] Guan'an Wang, Shuo Yang, Huanyu Liu, Zhicheng Wang, Yang Yang, Shuliang Wang, Gang Yu, Erjin Zhou, and Jian Sun. High-order information matters: Learning relation and topology for occluded person re-identification. In *CVPR*, pages 6448–6457, 2020. 1

[26] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, pages 79–88, 2018. 2, 6, 7

[27] Guile Wu, Xiatian Zhu, and Shaogang Gong. Tracklet self-supervised learning for unsupervised person re-identification. In *AAAI*, pages 12362–12369, 2020. 2, 7, 8

[28] Lin Wu, Chunhua Shen, and Anton van den Hengel. Person-net: Person re-identification with deep convolutional neural networks. In *arXiv:1601.07255*, 2016. 2

[29] Bryan (Ning) Xia, Yuan Gong, Yizhe Zhang, and Christian Poellabauer. Second order non-local attention networks for person re-identification. In *ICCV*, pages 3759–3768, 2019. 2

[30] Dong Yi, Zhen Lei, and Stan Z. Li. Deep metric learning for practical person re-identification. In *ICPR*, pages 34–39, 2014. 3

[31] Yunpeng Zhai, Qixiang Ye, Shijian Lu, Mengxi Jia, Rongrong Ji, and Yonghong Tian. Multiple expert brainstorming for domain adaptive person re-identification. In *ECCV*, pages 594–611, 2020. 2

[32] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. In *NeurIPS*, pages 5165–5175, 2018. 4

[33] 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*, pages 8506–8514, 2019. 2

[34] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jing dong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, pages 1116–1124, 2015. 2, 6, 7, 8

[35] Zhedong Zheng, Xiaodong Yanga, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, pages 2138–2147, 2019. 1

[36] Zhun Zhong, Liang Zheng, Shaozi Li, and Yi Yang. Generalizing a person retrieval model hetero-and homogeneously. In *ECCV*, pages 172–188, 2018. 2

[37] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*, pages 598–607, 2019. 2

[38] Zhun Zhong, Liang Zheng, Zhedong Zheng, Shaozi Li, and Yi Yang. Camera style adaptation for person re-identification. In *CVPR*, pages 5157–5166, 2016. 6

[39] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-scale feature learning for person re-identification. In *ICCV*, pages 3701–3711, 2019. 1

[40] Sanping Zhou, Fei Wang, Zeyi Huang, and Jinjun Wang. Discriminative feature learning with consistent attention regularization for person re-identification. In *In ICCV*, pages 8039–8048, 2019. 2

[41] Sanping Zhou, Jinjun Wang, Jiayun Wang, Yihong Gong, and Nanning Zheng. Point to set similarity based deep feature learning for person re-identification. In *CVPR*, pages 5028–5037, 2017. 2

[42] Kuan Zhu, Haiyun Guo, Zhiwei Liu, Ming Tang, and Jinqiao Wang. Identity-guided human semantic parsing for person re-identification. In *ECCV*, pages 346–363, 2020. 2

[43] Yang Zou, Xiaodong Yang, Zhiding Yu, B.V.K. Vijaya Kumar, and Jan Kautz. Joint disentangling and adaptation for cross-domain person re-identification. In *ECCV*, pages 87–104, 2020. 2