Deep Miner: A Deep and Multi-branch Network which Mines Rich and Diverse Features for Person Re-identification

Abdallah Benzine[1], Mohamed El Amine Seddik[2], Julien Desmarais[1]

[1]Digeiz AI Lab

[2]Ecole Polytechnique

Abstract

Most recent person re-identification approaches are based on the use of deep convolutional neural networks (CNNs). These networks, although effective in multiple tasks such as classification or object detection, tend to focus on the most discriminative part of an object rather than retrieving all its relevant features. This behavior penalizes the performance of a CNN for the re-identification task, since it should identify diverse and fine grained features. It is then essential to make the network learn a wide variety of finer characteristics in order to make the re-identification process of people effective and robust to finer changes.

In this article, we introduce Deep Miner, a method that allows CNNs to "mine" richer and more diverse features about people for their re-identification. Deep Miner is specifically composed of three types of branches: a Global branch (G-branch), a Local branch (L-branch) and an Input-Erased branch (IE-branch). G-branch corresponds to the initial backbone which predicts global characteristics, while L-branch retrieves part level resolution features. The IE-branch for its part, receives partially suppressed feature maps as input thereby allowing the network to "mine" new features (those ignored by G-branch) as output. For this special purpose, a dedicated suppression procedure for identifying and removing features within a given CNN is introduced. This suppression procedure has the major benefit of being simple, while it produces a model that significantly outperforms state-of-the-art (SOTA) re-identification methods. Specifically, we conduct experiments on four standard person re-identification benchmarks and witness an absolute performance gain up to 6.5% mAP compared to SOTA.

1. Introduction

In recent years, person re-identification (Re-ID) has attracted major interest due to its important role in various computer vision applications: video surveillance, human authentication, human-machine interaction etc. The main
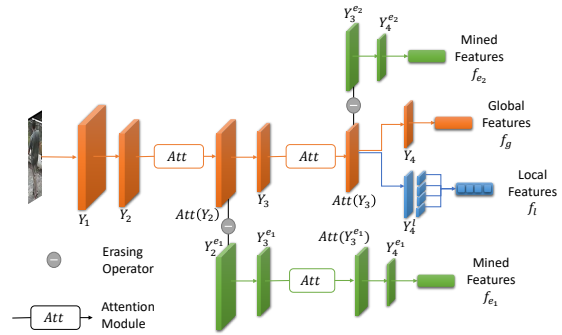

Figure 1: **Deep Miner Model Architecture.** Given a standard CNN backbone, several branches are created to enrich and diversify the features for the purpose of person re-identification. The proposed Deep Miner model is made of three types of branches: **(i)** The main branch G (in orange) is the original backbone and predicts the standard *global features* f_g; **(ii)** Several Input-Erased (IE) branches (in green) that takes as input erased feature maps and predict *mined features* f_{e_1} and f_{e_2}; **(iii)** The local branch (in blue) that outputs local features f_l and in which a uniform partition strategy is employed for *part level feature* resolution as proposed by [38]. In the global branch, and the bottom IE Branch, attention modules are used in order to improve their feature representation.

objective of person Re-ID is to determine whether a given person has already appeared over a network of cameras, which technically implies a robust modelling of the global appearance of individuals. The Re-ID task is particularly challenging because of significant appearance changes – often caused by variations in the background, the lightening conditions, the body pose and the subject orientation w.r.t. the recording camera.

In order to overcome these issues, one of the main goals of person Re-ID models is to produce rich representations of any input image for person matching. Notably, CNNs are known to be robust to appearance changes and spatial location variations, as their global features are invariant to such
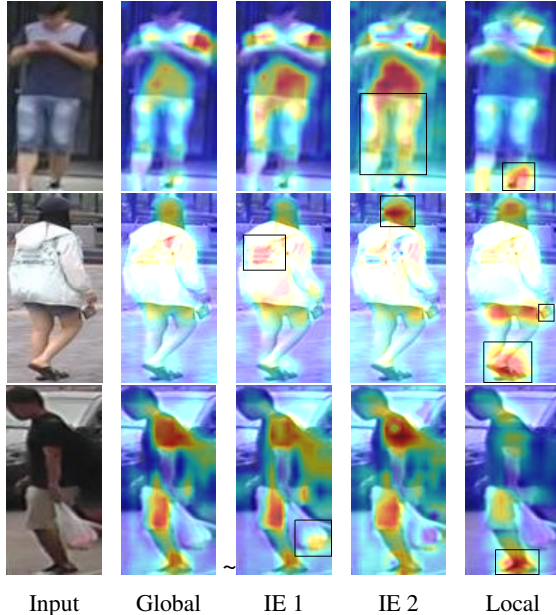
| Input | Global | IE 1 | IE 2 | Local |

Figure 2: **Feature visualization for three examples.** Warmer color denotes higher value. The global branch (second column) focuses only on some features but ignores other important ones. Thanks to the Input Erased branches (third and fourth columns), Deep Miner discovers new important features (localized by the black boxes). For instance, in the first row, the IE-branches are more attentive to the person pant. In the second row, they discover some patterns on the coat and get attentive to its cap. In the third row, they find out the plastic bag. The local branch (last column) helps the network to focus on local features such as the shoes of the subject or the object handled by the subject in the second row.

transformations. Nevertheless, the aforementioned global features are prone to ignore detailed and potentially relevant information for identifying specific person representations. To enforce the learning of detailed features, attention mechanisms and aggregating global part-based representations were introduced in the literature, yielding very promising results [5, 7, 31]. Specifically, attention mechanisms allow to reduce the influence of background noise and to focus on relevant features, while part-based models divide feature maps into spatial horizontal parts thereby allowing the network to focus on fine-grained and local features.

Despite their observed effectiveness in various tasks, these approaches do not provide ways to enrich and diversify an individual's representation. In fact, deep learning models are shown to exhibit a biased learning behavior [4, 3, 32]; in the sense that they retrieve sufficiently partial attributes concepts which contribute to reduce the training loss over the seen classes, rather than learning all-sided

details and concepts. Basically, deep networks tend to focus on surface statistical regularities rather than more general abstract concepts. This behavior is problematic in the context of re-identification, since the network is required to provide the richest and most diverse possible representations.

In this paper, we propose to address this problem by adding Input Erased Branches (IE-Branch) into a standard backbone. Precisely, an IE-Branch takes partially removed feature maps as input in the aim of producing (that is "mining") more diversified features as output (as depicted in Figure 2). In particular, the removed regions correspond quite intuitively to areas where the network has strong activations and are determined by a simple suppression operation (see subsection 3.2). The proposed Deep Miner model is therefore made as the combination of IE-branches with local and global branches. The multi-branch architecture of Deep Miner is depicted on Figure 1.

The main contributions brought by this work may be summarized in the following items: **(i)** We provide a multi-branch model allowing the mining of rich and diverse features for people re-identification. The proposed model includes three types of branches: a Global branch (G-branch), a Local branch (L-branch) and an Input-Erased Branch (IE-Branch); the latter being responsible of mining extra features; **(ii)** IE-Branches are constructed by adding an erase operation on the global branch feature maps, thereby allowing the network to discover new relevant features; **(iii)** Extensive experiments were conducted on Market1501 [44], DukeMTMC-ReID [24], CUHK03 [17] and MSMT17[37]. We demonstrate that our model significantly outperforms the existing SOTA methods on all benchmarks.

2. Related Work

There has been an extensive amount of works around the problem of people re-identification. This section particularly recalls the main advances and techniques to tackle this task, which we regroup subsequently in terms of different approaches:

Part-level features which essentially pushes the model to discover fine-grained information by extracting features at a part-level scale. Specifically, this approach consists in dividing the input image into multiple overlapping parts in order to learn part-level features [40, 17]. In the same vein, other methods of body division were also proposed; Pose-Driven Deep Convolutional (PDC) leverages human pose information to transform a global body image into normalized part regions, Part-based Convolution Baseline (PCB) [31] learns part level features by dividing the feature map equally – Essentially, the network has a 6-branch structure by dividing the feature maps into six horizontal stripes

and an independent loss is applied to each strip. Based on PCB, stronger part based methods were notably developed [43, 22, 36]. However, theses division strategies usually suffer from misalignment between corresponding parts – because of large variations in poses, viewpoints and scales. In order to avoid this misalignment, [38] suggest to concatenate the part-level feature vectors into a single vector and then apply a single loss to the concatenated vector. This strategy is more effective than applying individual loss to each part-level feature vector. As shall be seen subsequently, we particularly employ this strategy to learn the local branch of our Deep Miner model.

Metric learning methods contribute also to enhance the representations power of Re-ID features as notably shown in [6, 12, 30, 24]. For instance, the batch hard triplet loss introduced in [12] retrieves the hardest positive and the hardest negative samples for each pedestrian image in a batch. Moreover, the soft-margin triplet loss[1] [12] extends the hard triplet loss by using the SOFTPLUS function to remove the margin hyper-parameter. Finally, authors in [26] improve the training and testing processes by gallery-to-gallery affinities and through the use of a group-shuffling random walk network.

Attention Modules were also suggested to improve the feature representation of Re-ID models [5, 7, 13, 33, 27, 18, 39]. Specifically, a dual attention matching network with inter-class and intra-class attention module that captures the context information of video sequences was proposed in [27]. In [18], a multi-task model jointly learns soft pixel-level and hard region-level attention to improve the discriminative feature representations. In [39], the final feature embedding is obtained by combining global and part features through the use of pose information to learn attention masks.

In the aim of learning fine grained features, a series of works focus on using erasing methods – in various contexts beyond the Re-ID paradigm. We briefly recall the main works following this approach in the following paragraph and we particularly emphasize that our proposed Deep Miner model relies on a feature erasing approach.

Feature erasing methods were commonly used for weakly-supervised object localization [41, 9, 2]. Technically, these methods rely on erasing the most discriminative part to encourage the CNN to detect other parts of an object. Our Deep Miner Model relies on the same principal whereas it fundamentally differs in terms of the targeted purpose and from the implementation standpoint. Indeed, Deep Miner aims to enrich the feature representation of deep neural networks in order to properly distinguish a person from another one. Similar to the proposed Deep Miner model, authors in [8] introduce a Salient Feature Extraction unit

which suppresses the salient features learned in the previous cascaded stage thereby extracting other potential salient features. Still, our method differs from [8] through different aspects: **(i)** the erasing operation used in Deep Miner is conceptually simpler – since it consists in averaging and thresholding operations yielding an erasing mask – while being very efficient in terms of outcome; **(ii)** the obtained erasing mask is then multiplied by features maps of the initial CNN backbone allowing to create an Input-Erased branch that will discover new relevant features; **(iii)** by simply incorporating such simple Input-Erased branches into the standard Resnet50 backbone, Deep Miner achieves the same *mAP* in Market1501 as [8] which include many other complex modules (e.g. attention modules); **(iv)** Deep Miner contains a local branch with part level resolution that is absent in [8] and the two models differ in the used attention module; **(v)** finally, [8] integrates a non-local multi-stage feature fusion which we found unnecessary for Deep Miner to achieve high re-identification performance.

3. Proposed Method

This section presents in more details our proposed method. As previously discussed, Deep Miner aims to enable the learning of more rich and fine grained features in the context of person re-identification. Specifically, Deep Miner relies on a given CNN backbone (e.g., Resnet50), and enriching it with new branches (each of them is described subsequently) to allow the network to mine richer and more diverse features. The overall architecture of the network is described in Figure 1.

3.1. Global Branch

The Global branch (G-branch) corresponds to a standard CNN backbone like Resnet50. This backbone is composed of B convolutional blocks[2] $\mathbf{B}_i$ for $i \in [B]$[3]. The output of each block is denoted $\mathbf{Y}_i$. We particularly apply a global max pooling to the last convolutional layer (with stride being set to 1) yielding to an output vector $\boldsymbol{f}_g$. The global feature representation of a person is then obtained as a linear transformation of $\boldsymbol{f}_g$.

3.2. Input Erased Branch

As we discussed earlier, recent works [4, 3, 32] have demonstrated that CNNs tend to focus only on the most discriminative parts of an image. In the context of person re-identification, this behavior is problematic since the network may not use important information to predict valuable features to increase person identification.

To this end though, we propose to add new branches to the initial backbone network to mine a larger diversity

---

[1]This loss function is used for our Deep Miner model training.

[2]As a standard residual network block.
[3]The notation $[n] = \{1, \ldots, n\}$.
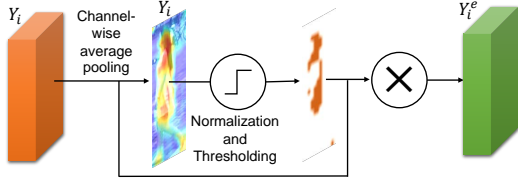
Figure 3: Erasing Operation (*ErO*).

Figure 4: Local Branch of Deep Miner.

of features from the images (yielding our Deep Miner proposed method). Specifically, we add Input-Erased branches (IE-branches) to the initial backbone. An IE-Branch can be added after any convolutional block $\mathbf{B}_i$[4] of the global branch and provide a feature vector in the same way as the global branch. The convolutional block $\mathbf{B}_{i+1}$ of the G-branch takes as input the unerased feature maps $\mathbf{Y}_i$[5], while the corresponding block in the IE-branch takes as input a partially erased version of it. An *Erasing Operation (ErO)* (illustrated in Figure 3) is particularly applied to the feature maps $\mathbf{Y}_i$ in order to obtain the erased features denoted $\mathbf{Y}_i^e$. The *ErO* operation consists first in compressing $\mathbf{Y}_i$ through channel-wise average pooling so to get the average features maps $\tilde{Y}_i$. A min-max normalization is then applied to $\tilde{Y}_i$ to obtain $\tilde{Y}_i^n = \text{MIN-MAX-NORM}(\tilde{Y}_i)$. Given a thresholding parameter $\tau \in [0,1]$, an erasing mask M_i is computed as follows:

$$M_i(x,y) = \begin{cases} 0, & \text{if } \tilde{Y}_i^n(x,y) > \tau \\ 1, & \text{otherwise} \end{cases} \quad (1)$$

where $I(x,y)$ stands for the pixel intensity at position (x,y) of a 2D map I. As such, the erasing mask M_i is simply obtained through averaging and thresholding operations. Furthermore, the erased features $\mathbf{Y}_i^e$ are then obtained by element-wise multiplication between M_i and each channel of $\mathbf{Y}_i$, i.e., $\mathbf{Y}_i^e(c) = M_i \odot \mathbf{Y}_i(c)$ for c indexing the channels.

As depicted in Figure 1, creating an IE branch k at block $\mathbf{B}_i$ of the backbone will result in an additional branch composed of $B-i$ convolutional blocks which we denote $\mathbf{B}_j^{e_k}$ for $j \in \{i+1, \dots, B\}$, each of them being identical to the corresponding convolutional block of the main branch (G-branch). In Figure 1, an IE-Branch 1 is created after the block $\mathbf{B}_2$ (bottom IE-Branch). This IE-Branch is composed of the blocks $\mathbf{B}_3^{e_1}$ and $\mathbf{B}_4^{e_1}$ which have the same layers architecture as $\mathbf{B}_3$ and $\mathbf{B}_4$ and are initialised with same pretrained weights. If an attention module (subsection 3.4) is added to the backbone, the same attention module is added to the IE-branch. In the same way, another IE-Branch is created after block $\mathbf{B}_3$. We stress however that the weights are

---

[4]If an attention module is used after block $\mathbf{B}_i$, the IE-branch is added after this attention module.

[5]If an attention module is used after block $\mathbf{B}_i$, $\mathbf{Y}_i$ stands for the output of this attention module.
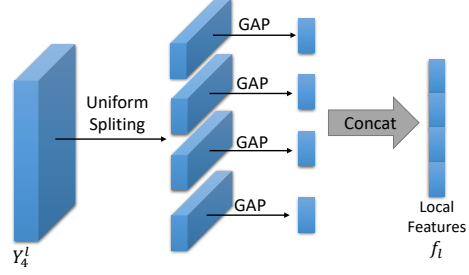
not shared between the different branches during training, so to let the network discover new features.

Like in the global branch, we apply a global max pooling to the last convolutional layer of an IE Branch k yielding to an output vector $\mathbf{f}_{e_k}$ followed by a linear layer.

Note that a similar erasing operation was already introduced in [8]. Nevertheless, the IE-branch in our proposed Deep Miner model differs from [8] in two fundamental ways: **(i)** the erasing mask is computed differently involving only a thresholding parameter τ. Indeed, *ErO* is simply based on average pooling and thresholding operations, while [8] uses a complex salient feature extractor that divides the features maps into several stripes, the selector guiding each stripe to mine important information; **(ii)** the erasing operation is performed in a cascaded way at the end of the Resnet50 backbone in [8], while it is performed at different stages of the backbone in our Deep Miner model. This notably allows our model to mine new relevant features at different resolutions and semantic levels.

3.3. Local Branch

While the IE-erased branches help the network to mine more diverse features, these branches are still based on the global appearance of a person. To help the network to mine local and more precise features, a local branch is added to Deep Miner. This branch is placed after the convolutional block $\mathbf{B}_3$ and is composed of the convolutional block $\mathbf{B}_4^l$ which have the same layers architecture as $\mathbf{B}_4$ and is initialised with the same pretrained weights. Like in the IE-Branches, the weights are not shared between the different branches during training, so to let the network discover new localized features.

As illustrated in Figure 4, the local branch outputs features maps that are then partitioned into 4 horizontal stripes. A global average pooling is then applied to each strip to obtain 4 local features vectors. The 4 local vectors are then concatenated yielding to an output vector $\mathbf{f}_l$ as performed in [38].

4

3.4. Attention Modules

Attention modules are commonly used in various deep learning application tasks and specifically in the context of person re-identification. The proposed Deep Miner model is also compatible with attention modules, which notably yield an enhancement of the model ability to retrieve more relevant features. To stress out the effectiveness of attention modules on the proposed method, we implement a simple attention module composed of a Spatial Attention Module (SAM) and a CHannel Attention Module (CHAM) which we describe subsequently. Features maps Y_i are first processed by SAM, the result of which is then processed by CHAM. The obtained features after SAM and CHAM are denoted $\text{ATT}(Y_i) = \text{CHAM}(\text{SAM}(Y_i))$.

SAM focuses on the most relevant features within the spatial dimension which is essentially based on [7, 38]. Indeed, SAM captures and aggregates related features in the spatial domain. An illustration of this module is depicted in Figure 5. The input feature maps Y_i of dimension $H \times W \times C$ corresponding to the output of the convolutional block B_i are fed into two convolutional layers to get two features maps A_i and B_i both of dimension $H \times W \times \frac{C}{8}$. The tensor A_i is transposed and reshaped to shape $D \times C$ and B_i is reshaped to $C \times D$, with $D = H \times W$. An affinity matrix $\tilde{F}_i = A_i^r \cdot B_i^r \in \mathbb{R}^{D \times D}$ is computed by a matrix multiplication between the reshaped tensors A_i and B_i denoted respectively as A_i^r and B_i^r. A Softmax activation is then applied to $\tilde{F}_i$ leading to F_i. After a reshaping operation, Y_i is multiplied (map-wise) by F_i and a Batch Normalization layer is applied to the resulting tensor followed by a multiplication with a learnable scalar γ. This parameter adjusts the importance of the SAM transformation. The result is then added to the input Y_i to get $\text{SAM}(Y_i)$, the resulting tensor of the Spatial Attention Module.
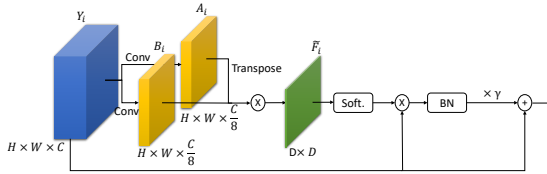

Figure 5: Spatial Attention Module (SAM).

In contrast, CHAM as illustrated in Figure 6, explores the correlation and the inter-dependencies between channel features. It is specifically based on the Squeeze-and-Exitation block [14]. Besides, compared to [14], the global average pooling at the beginning of the block is removed to preserve spatial information into the attention block. A convolutional layer is applied to $\text{SAM}(Y_i)$ to obtain feature maps of size $H \times W \times \frac{C}{16}$ for which a second convolution layer is applied to obtain feature maps of size $H \times W \times C$. A Soft-max activation is applied to the result that is then multiplied element-wise by $\text{SAM}(Y_i)$ to obtain the result of CHAM.
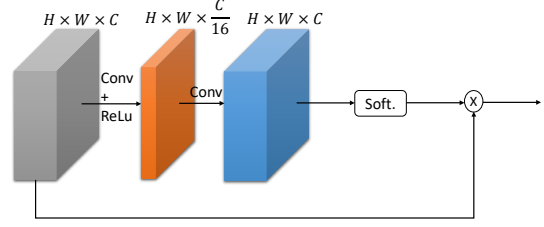

Figure 6: CHannel Attention Module (CHAM).

As depicted in Figure 1, the attention module is placed right after the blocks B_2 and B_3 of the G-branch as well as after the block $B_3^{e_1}$ of IE-branch 1 (bottom IE branch).

3.5. Loss Functions

Each branch $\text{Br} \in \{g, l, e_1, e_2, \ldots\}$ (among global branch, local branch and IE branches) outputs a corresponding feature vector f_{Br}. We apply the sames losses for each feature vector, i.e., an ID loss with label smoothing $\mathcal{L}_{\text{ID}}^{\text{Br}}$, a soft margin triplet loss $\mathcal{L}_{\text{triplet}}^{\text{Br}}$ and a center loss $\mathcal{L}_{\text{center}}^{\text{Br}}$ yielding a global loss for each branch Br as

$$\mathcal{L}_{\text{Br}} = \mathcal{L}_{\text{ID}}^{\text{Br}} + \mathcal{L}_{\text{triplet}}^{\text{Br}} + \lambda \mathcal{L}_{\text{center}}^{\text{Br}} \tag{2}$$

where $\lambda > 0$ is an hyper-parameter ($\lambda = 5 \cdot 10^{-4}$ in all our experiments). The ID loss is specifically defined as

$$\mathcal{L}_{\text{ID}}^{\text{Br}} = -\frac{1}{N} \sum_{i=1}^{N} q_i \log\left(p_i^{\text{Br}}\right) \tag{3}$$

with N standing for the number of samples, p_i^{Br} denotes the predicted probability for identity i, while q_i stands for the (ground-truth) smoothed label, and is defined as

$$q_i = \begin{cases} 1 - \epsilon\frac{N-1}{N}, & \text{if } i = y \\ \frac{\epsilon}{N}, & \text{otherwise} \end{cases} \tag{4}$$

where y is the (hard) ground-truth identity and ϵ is a precision parameter ($\epsilon = 0.1$ in practice) used to enforce the model to be less confident on the training set. Note that we apply the BNNeck [21] to the feature vector f_{Br} right before the linear layer predicting the IDs probabilities.

The soft margin triplet loss $\mathcal{L}_{\text{triplet}}^{\text{Br}}$ is employed to enhance the final ranking performance of the Re-ID model. $\mathcal{L}_{\text{triplet}}^{\text{Br}}$ is formally defined as

$$\sum_{i=1}^{P} \sum_{k=1}^{K} \text{SOFTPLUS}(\max_{\ell \in [K]} \left\| f_{\text{Br}}(x_k^i) - f_{\text{Br}}(x_\ell^i) \right\|_2$$
$$- \min_{\substack{j \in [P] \setminus \{i\} \\ \ell \in [K]}} \left\| f_{\text{Br}}(x_k^i) - f_{\text{Br}}(x_\ell^j) \right\|_2) \tag{5}$$

5

where P stands for the number of identities per batch, K is the number of samples per identity, $\boldsymbol{x}_k^i$ is the k-th image of person i and $\boldsymbol{f}_{\texttt{Br}}(\boldsymbol{x}_k^i)$ is the corresponding predicted feature vector (extracted among the branches of Deep Miner).

Following [21], the Center Loss $\mathcal{L}_{\text{center}}^{\texttt{Br}}$ is also considered for training our Deep Miner model. It simultaneously learns a center for deep features of each identity and penalizes the distances between the deep features and their corresponding identity centers. As such, intra-class compactness is increased. The Center Loss is particularly defined as

$$\mathcal{L}_{\text{center}}^{\texttt{Br}} = \frac{1}{2} \sum_{i=1}^{P} \sum_{k=1}^{K} \left\| \boldsymbol{f}_{\texttt{Br}}(\boldsymbol{x}_k^i) - \boldsymbol{c}_{\texttt{Br}}^i \right\|_2^2 \qquad (6)$$

where $\boldsymbol{c}_{\texttt{Br}}^i$ is the center of identity i for the branch $\texttt{Br}$.

4. Experiments

4.1. Experimental details

Deep Miner is implemented in PyTorch and trained on a single Nvidia GV100 GPU. All images are resized into 384×128 pixels. Random Horizontal flipping and erasing are used during training. Each mini batch consists of $N = P \times n$ images where P is the number of randomly selected identities and n is the number of samples per identity. We take $P = 16$ and $n = 4$. We employ Adam as the optimizer with a warm-up strategy for the learning rate. We spent 10 epochs linearly increasing the learning rate from 3.5×10^{-5} to 3.5×10^{-4}. Then, the learning rate is decayed to 3.5×10^{-5} and 3.5×10^{-6} at 40-th epoch and 70-th epoch respectively. The model is trained until convergence. The feature vector of an input image produced by Deep Miner corresponds the concatenation of the feature vectors predicted by each branch after application of the BNNeck. A `Resnet50` backbone is used in all experiments.

4.2. Evaluation Metrics

To compare the performance of the proposed method with previous SOTA methods, the Cumulative Matching Characteristics (CMC) at *Rank-1* and mean Average Precision (*mAP*) are adopted.

4.3. Datasets

We considered four standard re-identification benchmarks, the details of which are provided subsequently.

Market1501 [44] consists of 1501 identities and 32668 images. 12936 images of 751 subjects are used for training while 19732 images of 750 subjects are used for testing with 3368 query images and 16364 gallery images. The images are shot by six cameras. The Deformable Part Model is used to generate the bounding boxes [11].

DukeMTMC-ReID [24] contains 36411 images of 1404 identities captured by more than 2 cameras. The training subset contains 702 identities with 16522 images and the testing subset has other 702 identities. The gallery set contains 17661 images and the query set contains 2228 images.

CUHK03 [17] contains 1467 identities and a total of 14096 labeled images and 14097 detected captured by two camera views. 767 identities are used for training and 700 identities are used for testing. The labeled dataset contains 7368 training images, 5328 gallery and 1400 query images for testing, while the detected dataset contains 7365 images for training, 5332 gallery, and 1400 query images for test.

MSMT17 [37] is the largest and more challenging person Re-ID dataset. 4101 identities and 126441 images are captured by a 15-camera network (12 outdoor and 3 indoor). Faster RCNN [23] is used to annotate the bounding boxes.

4.4. Ablation Studies

To demonstrate the effectiveness of the IE and Local branches on the performance of Deep Miner, we incrementally evaluate each module on **Market-1501**. First, we evaluate the effect of the IE-branch.

Erasing threshold: We evaluate a model with a single IE branch and vary the erasing threshold (Figure 7). Compared to the baseline, adding the IE Branch improves the *mAP* with all the evaluated thresholds. Nevertheless, a careful choice of the optimal value of the threshold is needed for an optimal performance. With a low threshold, too much features are erased and the IE-Branch cannot mine new significant features. With a high threshold, no enough features are removed are the IE-branch does not discover new ones. The highest scores are obtained with $\tau = 0.8$ with 94.7% *Rank-1* score and 88.0% *mAP* score.
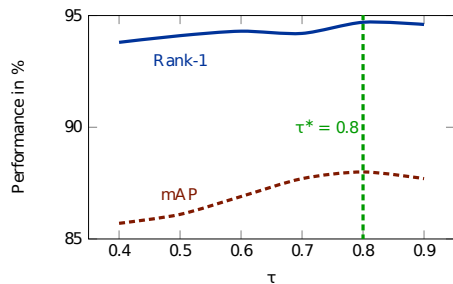
Figure 7: Influence of the erasing threshold τ with a single IE-Branch. The IE-Branch is placed after B_3, the third `Resnet50` convolution block. The baseline (only global branch) has 94.2% *Rank-1* and 84.7% *mAP*.

IE-Branch Position: We now evaluate the importance of the position of the IE-branch (see Table 1). The IE-branch has a significant impact on the model performance, whatever its position in the CNN backbone. Still, the optimal scores are obtained at a particular position of the backbone. In fact, the best scores are obtained when the IE-branch is placed after the third convolution block with a

94.7% *Rank-1* and 88.0% *mAP*.

IE Branch Position	*mAP*	*Rank-1*
baseline (only global branch)	84.7	94.2
After block 1	87.0	94.3
After block 2	87.4	94.6
After block 3	**88.0**	**94.7**

Table 1: Influence of IE-Branch position ($\tau = 0.8$).

Multiple IE-Branches: In this paragraph we evaluate whether the incorporation of multiple IE-branches into Deep Miner improves the results compared to a single IE-branch. Table 2 shows that adding IE-branches after the convolution block 3 and the convolutions block 1 or 2 improves the re-identification performance. Indeed, multiple IE-branches ensure that Deep Miner discovers new features at different semantics levels of the initial backbone. In particular, the optimal scores are obtained by adding IE-branches after the blocks 2 and 3. However, adding IE-branches after three different convolutional blocks does not yield a significant improvement compared to the 2 IE-branches architecture.

IE Branch Positions	*mAP*	*Rank-1*
After block 3	88.0	94.7
After blocks 2 and 3	**88.5**	**95.3**
After blocks 1 and 3	88.3	95.1
After blocks 1,2,3	88.2	95.0

Table 2: Influence of multiple IE-Branches ($\tau = 0.8$).

Local Branch: Now we evaluate the effect of adding a local branch. Table 3 shows that adding this branch to the optimal Multiple IE-branches (obtained by the previous ablation study, i.e., adding IE-branches after blocks 2 and 3) significantly improves the *mAP* (89.4% versus 88.5%).

Local Branch	*mAP*	*Rank-1*
No	88.5	**95.3**
Yes	**89.4**	95.2

Table 3: Influence of the Local Branch. IE-branches are placed after blocks 2 and 3 ($\tau = 0.8$).

Attention Modules: As previously discussed, attention modules are shown to increase the performance of modern person re-identification models. We therefore evaluate in this part the effect of such modules on Deep Miner. Table 4 shows that the attention module described in Subsection 3.4 helps the proposed model to mine relevant features with a 1.0% absolute *mAP* improvement.

In the next section, we consider the optimal Deep Miner model obtained through this ablation studies which we com-

Attention Module	*mAP*	*Rank-1*
No	89.4	95.2
SAM	90.2	95.5
CHAM	89.4	95.5
SAM & CHAM	**90.4**	**95.7**

Table 4: Influence of the Attention Module. IE-branches are placed after blocks 2 and 3 ($\tau = 0.8$) and a Local branch is used.

Method	Backbone	*mAP*	*Rank-1*
Deep Miner	Resnet50	**90.40**	**95.70**
*+ PLR OSNet [38]	OSNet	88.90	95.60
~HOReID [35]	Resnet50	84.9	94.2
*+ SCSN [8]	Resnet50	88.50	**95.70**
* ABDNet [7]	Resnet50	88.28	95.60
+ Pyramid [43]	Resnet101	88.20	**95.70**
DCDS [1]	Resnet101	85.80	94.81
*+ MHN [5]	Resnet50	85.00	95.10
+ MGN [19]	Resnet50	86.9	95.70
BFE [10]	Resnet50	86.20	95.30
*+CASN [45]	Resnet50	82.80	94.40
*+AANet [33]	Resnet152	83.41	93.93
* IANet [13]	Resnet50	83.10	94.40
*+ VPM [29]	Resnet50	80.80	93.00
~PSE+ECN [25]	Resnet50	80.50	90.40
+ PCB+RPP [31]	Resnet50	81.60	93.80
+ PCB [31]	Resnet50	77.40	92.30
~Pose-transfer [20]	DenseNet169	56.90	78.50
~SPReID [16]	Resnet152	83.36	93.68
* RGA-SC [42]	Resnet50	88.4	96.1
SNR [15]	Resnet50	84.70	94.40
OSNet [47]	OSNet	84.9	94.8
Tricks [21]	SEResNet101	87.30	94.60
* Mancs [34]	Resnet50	82.30	93.10
PAN [46]	Resnet50	63.40	82.80
SVDNet [30]	Resnet50	62.10	82.30

*Attention related, +Stripes Related, ~Pose or human pose related

Table 5: Comparison with state-of-the-art person Re-ID methods on the Market1501 dataset.

pare to State-of-the-art methods in terms of *mAP* and *Rank-1* score. Specifically, we define the Deep Miner model as a Resnet50 with two IE-branches placed after blocks 2 and 3, a local branch and attention modules (see Figure 1).

4.5. Comparison with State-of-the-art Methods

Now we compare our proposed Deep Miner model with recent SOTA methods to demonstrate its effectiveness and robustness compared to more advanced methods.

Market1501: Table 5 shows the results on the **Market1501** dataset. We divide the methods into two groups: methods that uses local features (top of the table) and methods that uses only global features (bottom). Deep Miner significantly outperforms previous methods in terms of the *mAP* score and has the same *Rank-1* score as the

Method	Backbone	mAP	Rank-1
Deep Miner	Resnet50	**81.80**	91.20
∗+ PLR OSNet [38]	OSNet	81.20	**91.60**
~HOReID [35]	Resnet50	75.60	86.90
∗+ SCSN [8]	Resnet50	79.00	91.00
∗ ABDNet [7]	Resnet50	78.60	89.00
+ Pyramid [43]	Resnet101	79.00	89.00
DCDS [1]	Resnet101	75.50	87.50
∗+ MHN [5]	Resnet50	77.20	89.10
+ MGN [19]	Resnet50	78.40	88.70
BFE [10]	Resnet50	75.90	88.90
∗+CASN [45]	Resnet50	73.70	87.70
∗+AANet [33]	Resnet50	74.29	87.65
∗ IANet [13]	Resnet50	73.40	83.10
∗+ VPM [29]	Resnet50	72.60	83.60
~PSE+ECN [25]	Resnet50	75.70	84.50
+ PCB+RPP [31]	Resnet50	69.20	83.30
~Pose-transfer [20]	Densenet169	56.90	78.50
~SPReID [16]	Resnet152	73.34	85.95
SNR [15]	Resnet50	72.9	84.4
OSNet [47]	OSNet	73.50	88.60
Tricks [21]	SeResnet101	78.00	87.50
∗ Mancs [34]	Resnet50	71.80	84.90
PAN [46]	Resnet50	51.51	71.59
SVDNet [30]	Resnet50	56.80	76.70

*Attention related, +Stripes Related, ~Pose or human pose related

Table 6: Comparison with state-of-the-art person Re-ID methods on the DukeMTMC-ReID dataset.

Method	Backbone	Labeled		Detected	
		mAP	Rank-1	mAP	Rank-1
Deep Miner	Resnet50	**84.7**	86.6	**81.4**	83.5
∗+ PLR OSNet [38]	OSNet	80.5	84.6	77.2	80.4
∗+ SCSN [8]	Resnet50	84.00	**86.80**	81.00	**84.70**
+ Pyramid [43]	Resnet101	76.90	78.90	74.80	78.90
∗+ MHN [5]	Resnet50	72.40	77.20	65.40	71.70
+ MGN [19]	Resnet50	67.40	68.00	66.00	68.00
BFE [10]	Resnet50	76.60	79.40	73.50	76.40
∗+CASN [45]	Resnet50	68.00	73.70	64.40	71.50
+ PCB+RPP [31]	Resnet50	-	-	57.50	63.70
OSNet [47]	OSNet	-	-	67.8	72.3
Tricks [21]	SeResnet101	70.40	72.00	68.00	69.60
∗ Mancs [34]	Resnet50	63.90	69.00	60.50	65.50

*Attention related, +Stripes Related,

Table 7: Comparison with state-of-the-art person Re-ID methods on the the CUHK03 dataset with the 767/700 split.

SOTA methods SCSN and Pyramid while being conceptually much simpler than the former. We also stress that Deep Miner surpasses methods that use stronger backbones while we only consider Resnet50. In fact, authors in [43] and [1] use Resnet101 while authors in [33] and [16] use a Resnet152. Moreover, Deep Miner – with a local branch that uses only 4 stripes – surpasses more complex methods like Pyramid that uses a pyramidal feature set and a large number of stripes. Overall, the proposed Deep Miner model surpasses all SOTA methods of the global features group demonstrating its ability to mine more diverse and richer features for person re-identification.

DukeMTMC-ReID: Similar to **Market1501**, results in

Method	Backbone	mAP	Rank-1
Deep Miner	Resnet50	**67.30**	**85.60**
∗+ SCSN [8]	Resnet50	58.50	83.80
∗ ABDNet [7]	Resnet50	60.80	82.30
BFE [10]	Resnet50	51.50	78.80
∗ IANet [13]	Resnet50	46.80	75.50
GLAD[36]	Resnet50	34.00	61.40
PDC[28]	GoogLeNet	29.70	58.00

*Attention related, +Stripes Related

Table 8: Comparison with state-of-the-art person Re-ID methods on the MSMT17 dataset.

Table 6 show that Deep Miner achieves optimal *mAP* scores compared to all SOTA methods on this dataset.

CUHK03: The results are shown in Table 7. Note that this dataset is more challenging than the previous ones in the sense that it contains fewer number of samples and has limited viewpoint variations. The proposed Deep Miner model is shown to achieve the optimal *mAP* score compared to SOTA methods. Indeed, Deep Miner exceeds PLR OSNet by 4.2% in *mAP* on the labeled dataset and exceeds SCSN by 0.7%. On the detected dataset, it surpasses PLR OSNet by 3.2% and SCSN by 0.4%. These experimental results notably express the benefits of the feature mining method, even under the condition of limited training samples.

MSMT17: Lastly, Table 8 shows the results on the **MSMT17** dataset, which is the more recent and largest dataset. Note that Deep Miner significantly outperforms all SOTA methods in terms of both *mAP* and *Rank-1* scores. Notably, we obtain a *significant gain* of **6.5%** *mAP* compared to the best SOTA method on this dataset.

5. Conclusion

This paper introduced Deep Miner, a multi-branch network that mines rich and diverse features for person re-identification. This model is composed of three types of branches that are all complementary to each other: the global branch extracts general features of the person; the Input-Erased branches mine richer and more diverse features; the Local branch looks for fine grained details. Specifically, our main insight is to add to a given backbone Input-Erased Branches which take as input partially erased features maps (through a simple erasing strategy) and find new features ignored by the backbone branch. Extensive experiments have demonstrated the effectiveness of the proposed Deep Miner model and its superiority in terms of performance compared to much more complex SOTA methods. It is worth noting that the proposed Deep Miner model makes **6.5%** *mAP* improvement on the **MSMT17** dataset – the largest and more complex existing person re-identification dataset.

References

[1] Leulseged Tesfaye Alemu, Marcello Pelillo, and Mubarak Shah. Deep constrained dominant sets for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9855–9864, 2019. 7, 8

[2] Sabrina Narimene Benassou, Wuzhen Shi, Feng Jiang, and Abdallah Benzine. Hierarchical complementary learning for weakly supervised object localization. *arXiv preprint arXiv:2011.08014*, 2020. 3

[3] Binghui Chen and Weihong Deng. Energy confused adversarial metric learning for zero-shot image retrieval and clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8134–8141, 2019. 2, 3

[4] Binghui Chen and Weihong Deng. Hybrid-attention based decoupled metric learning for zero-shot image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2750–2759, 2019. 2, 3

[5] Binghui Chen, Weihong Deng, and Jiani Hu. Mixed high-order attention network for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 371–381, 2019. 2, 3, 7, 8

[6] Dapeng Chen, Dan Xu, Hongsheng Li, Nicu Sebe, and Xiaogang Wang. Group consistent similarity learning via deep crf for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8649–8658, 2018. 3

[7] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 8351–8361, 2019. 2, 3, 5, 7, 8

[8] Xuesong Chen, Canmiao Fu, Yong Zhao, Feng Zheng, Jingkuan Song, Rongrong Ji, and Yi Yang. Salience-guided cascaded suppression network for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3300–3310, 2020. 3, 4, 7, 8

[9] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2219–2228, 2019. 3

[10] Zuozhuo Dai, Mingqiang Chen, Xiaodong Gu, Siyu Zhu, and Ping Tan. Batch dropblock network for person re-identification and beyond. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3691–3701, 2019. 7, 8

[11] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008. 6

[12] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 3

[13] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen. Interaction-and-aggregation network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9317–9326, 2019. 3, 7, 8

[14] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 5

[15] Xin Jin, Cuiling Lan, Wenjun Zeng, Zhibo Chen, and Li Zhang. Style normalization and restitution for generalizable person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3143–3152, 2020. 7, 8

[16] Mahdi M Kalayeh, Emrah Basaran, Muhittin Gökmen, Mustafa E Kamasak, and Mubarak Shah. Human semantic parsing for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1062–1071, 2018. 7, 8

[17] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 152–159, 2014. 2, 6

[18] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2018. 3

[19] Yutian Lin, Liang Zheng, Zhedong Zheng, Yu Wu, Zhilan Hu, Chenggang Yan, and Yi Yang. Improving person re-identification by attribute and identity learning. *Pattern Recognition*, 95:151–161, 2019. 7, 8

[20] Jinxian Liu, Bingbing Ni, Yichao Yan, Peng Zhou, Shuo Cheng, and Jianguo Hu. Pose transferrable person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4099–4108, 2018. 7, 8

[21] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 5, 6, 7, 8

[22] Ruijie Quan, Xuanyi Dong, Yu Wu, Linchao Zhu, and Yi Yang. Auto-reid: Searching for a part-aware convnet for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3750–3759, 2019. 3

[23] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. 6

[24] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer, 2016. 2, 3, 6

[25] M Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 420–429, 2018. 7, 8

[26] Yantao Shen, Hongsheng Li, Tong Xiao, Shuai Yi, Dapeng Chen, and Xiaogang Wang. Deep group-shuffling random

walk for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2265–2274, 2018. 3

[27] Jianlou Si, Honggang Zhang, Chun-Guang Li, Jason Kuen, Xiangfei Kong, Alex C Kot, and Gang Wang. Dual attention matching network for context-aware feature sequence based person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5363–5372, 2018. 3

[28] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. Pose-driven deep convolutional model for person re-identification. In *Proceedings of the IEEE international conference on computer vision*, pages 3960–3969, 2017. 8

[29] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 393–402, 2019. 7, 8

[30] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang. Svdnet for pedestrian retrieval. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3800–3808, 2017. 3, 7, 8

[31] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 480–496, 2018. 2, 7, 8

[32] Youssef Tamaazousti, Hervé Le Borgne, Céline Hudelot, Mohamed El Amine Seddik, and Mohamed Tamaazousti. Learning more universal representations for transfer-learning. *arXiv:1712.09708*, 2019. 2, 3

[33] Chiat-Pin Tay, Sharmili Roy, and Kim-Hui Yap. Aanet: Attribute attention network for person re-identifications. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7134–7143, 2019. 3, 7, 8

[34] Cheng Wang, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Mancs: A multi-task attentional network with curriculum sampling for person re-identification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 365–381, 2018. 7, 8

[35] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6449–6458, 2020. 7, 8

[36] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning discriminative features with multiple granularities for person re-identification. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 274–282, 2018. 3, 8

[37] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 79–88, 2018. 2, 6

[38] Ben Xie, Xiaofu Wu, Suofei Zhang, Shiliang Zhao, and Ming Li. Learning diverse features with part-level resolution for person re-identification. *arXiv preprint arXiv:2001.07442*, 2020. 1, 3, 4, 5, 7, 8

[39] Jing Xu, Rui Zhao, Feng Zhu, Huaming Wang, and Wanli Ouyang. Attention-aware compositional network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2119–2128, 2018. 3

[40] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Deep metric learning for person re-identification. In *2014 22nd International Conference on Pattern Recognition*, pages 34–39. IEEE, 2014. 2

[41] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1325–1334, 2018. 3

[42] Zhizheng Zhang, Cuiling Lan, Wenjun Zeng, Xin Jin, and Zhibo Chen. Relation-aware global attention for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3186–3195, 2020. 7

[43] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8514–8522, 2019. 3, 7, 8

[44] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015. 2, 6

[45] Meng Zheng, Srikrishna Karanam, Ziyan Wu, and Richard J Radke. Re-identification with consistent attentive siamese networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5735–5744, 2019. 7, 8

[46] Zhedong Zheng, Liang Zheng, and Yi Yang. Pedestrian alignment network for large-scale person re-identification. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(10):3037–3045, 2018. 7, 8

[47] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-scale feature learning for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3702–3712, 2019. 7, 8