# List-wise learning-to-rank with convolutional neural networks for person re-identification

Yiqiang Chen, Stefan Duffner, Andrei Stoian, Jean-Yves Dufour, Atilla Baskurt

**Machine Vision and Applications manuscript No.**
(will be inserted by the editor)

# List-wise Learning-to-rank with Convolutional Neural Networks for Person Re-identification

**Yiqiang Chen · Stefan Duffner · Andrei Stoian · Jean-Yves Dufour · Atilla Baskurt**

Received: 2019-04-12 / Accepted: 2021-02-27

**Abstract** In this paper, we present a novel machine learning-based image ranking approach using Convolutional Neural Networks (CNN). Our proposed method relies on a similarity metric learning algorithm operating on lists of image examples and a loss function taking into account the ranking in these lists with respect to different query images. This comprises two major contributions: (1) Rank lists instead of image pairs or triplets are used for training, thus integrating more explicitly the order of similarity and relations between sets of images. (2) A weighting is introduced in the loss function based on two evaluation measures: the mean average precision and the rank 1 score. We evaluated our approach on two different computer vision applications that are commonly formulated as ranking problems: person re-identification and image retrieval with several public benchmarks and showed that our new loss function outperforms other common functions and that our method achieves state-of-the-art performance compared to existing approaches from the literature.

**Keywords** similarity metric learning · learning-to-rank · person re-identification · Deep Learning · image retrieval

## 1 Introduction

In many applications related to image and video indexation, video surveillance, recommendation systems or mobile robotics, we face the problem of retrieving similar images from a large dataset with known identities or categories. Due to the complexity and variability of the input data, different types of noise and the difficulty to capture the inherent similarities with simple metrics or rules leads to approaches that aim at directly learning this similarity metric from annotated image examples. This allows to construct complex, non-linear metrics that encode

Y. Chen, S. Duffner, A. Baskurt
Université de Lyon, INSA-Lyon, CNRS, LIRIS, F-69621, Villeurbanne, France

Andrei Stoian, Jean-Yves Dufour
Thales Services, ThereSIS, Palaiseau, France

relations on a higher semantic level and that are robust to the various types of noise that are commonly present in realistic image data. In this work, we focus on the challenging problem of person re-identification from images that have been captured by different surveillance cameras with non-overlapping fields of view. In this setting, face recognition and other fine biometric cues are not always available due to the low image resolution. Therefore, the appearance of the whole person (*e.g.* clothing, hair style or other physical traits) is mainly exploited for this task. Moreover, person re-identification is necessary for many critical applications such as cross-camera tracking [1], multi-camera behaviour analysis [2] and forensic search [3].

However, this problem remains challenging due to the large variations of viewpoint and background. The images from the same individual can have very different appearance, and, different individuals may look similar in appearance. It can be difficult even for humans given the severe lighting and pose differences between images.

To tackle these problems, existing person re-identification approaches generally either build a discriminative feature representation [4–8] or learn a distance metric [9–12]. The extracted features should be robust to challenging factors like pose variation while preserving the identity information. Metric learning approaches try to learn the semantic similarities (and dissimilarities) between images from the same and from different persons. They either learn a distance metric or a projection of the features into a subspace, where intra-class distances are minimised and inter-class distances are maximised.

Many recent person re-identification approaches are based on Convolutional Neural Networks (CNN). The advantage of these models is that they learn feature representations and a distance metric jointly in an integrated framework. To train these neural networks, different loss functions have been proposed such as contrastive loss, triplet loss or quadruplet loss. However, minimising such loss functions based on pairs, triplets or quadruplets only take into account constraints between few (two to four) examples and do not incorporate any information about a more global ranking or order of the instances. Also, the exact ranking order is usually not known or only partially defined. For example, a good match must be before an incorrect one, but there is not necessarily an order among good matches. To address these issues, in this work, we propose a novel list-wise loss function which we call the Rank-Triplet loss. It is based on the predicted and ground truth ranking of a list of instances with respect to a query image. To incorporate partial order constraints, our approach specifically focuses on mis-ranked instances.

Furthermore, existing deep learning methods are solely based on the minimisation of a loss function defined on a certain similarity metric between different examples. However, the final evaluation measures are computed on the overall ranking accuracy. Inspired by the learning-to-rank method LambdaRank [13], our optimisation approach directly incorporates these evaluation measures in the loss function. During training, each image in the training batch is used as probe image in turn and the rest as gallery. For each query, the mean average precision and rank 1 score are calculated. Triplets are formed by the probe image and a pair of mis-ranked true and false correspondence.

The loss of one triplet is weighted by the improvement of these evaluation measures by swapping the rank positions of the true and false correspondences, as shown in Fig. 1. This evaluation measure-based weighting makes better use of

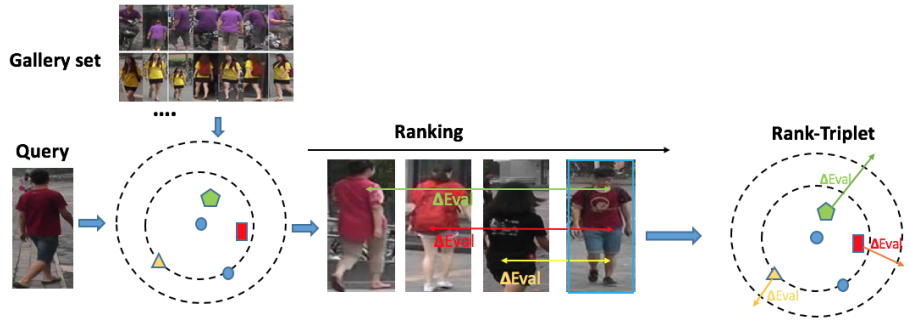


Fig. 1: Schematic illustration of Rank-Triplet. An image of a person of interest on the left (the query) is used to rank images from a gallery according to how closely they match that person. The correct match, highlighted in a blue box, can be difficult to find given the similar negative images, pose and viewpoint variations and occlusions. During training, we propose to estimate the importance of misranked pairs by the gain of the evaluation measure incurred by swapping the rank positions and to weight the loss according to their importance. In this example, swapping the falsely ranked (positive) image on the right with the leftmost one would lead to the biggest improvement ($\Delta$Eval).

difficult triplets which can bring a larger rank improvement and are more effective for the learning, and, at the same time, keep the learning stable by using all misranked pairs. Only using the hardest examples can, in practice, lead to bad local minima early during training.

The main contributions of this paper can be summarised as follows:

- A novel list-wise loss function and training strategy for neural networks that combine the triplet loss and LambdaRank, where, in each training iteration, triplets are formed effectively according to the rankings in the given list.
- A new weighting term in the loss function, based on the combination of two ranking measures: the mean average precision and the rank 1 score. This new loss function considers image retrieval and re-identification problems in a conceptually more natural way than previous work by directly taking into account the ranking evaluation scores.
- A thorough experimental evaluation showing that the proposed loss function outperforms other common functions and that our approach achieves state-of-the-art results on three challenging person re-identification datasets as well as an image retrieval dataset.

The rest of the paper is organised as follows. Section 2 introduces related work in the literature. In Section 3, we explain the proposed method in detail, and, in Section 4, we present the results of an extensive experimental evaluation of our approach and a comparison with state-of-the-art algorithms. Finally, conclusions are drawn in Section 5.

## 2 Related work

### 2.1 Learning-to-rank

Learning-to-rank is a class of techniques that learns a model for optimal ordering of a list of items. It is widely applied in information retrieval and natural language processing. Many learning-to-rank methods of different categories have been proposed in the literature.

The *pairwise* learning-to-rank approaches try to compare the relevance of every two documents, then rank all the documents based on all these comparison results. For example, RankSVM [14] seek to learn a ranking function in a higher dimensional feature space where true matches and wrong matches become more separable than the original feature space via the kernel trick. Prosser *et al.* [15] reformulated the person re-identification problem as a ranking problem. Their method learns a set of weak RankSVMs, each computed on a small set of data, and then combines them to build a stronger ranker using ensemble learning. And Ranknet [16] is the first neural network based learning-to-rank method. Query dependent features are extracted as the inputs of the network. To learn the model, the cross entropy cost function is minimised. It penalises the deviation of the model output probabilities from the desired probabilities: let $\overline{P}_{ij} = \{-1, 1\}$ be the known probability that training $x_i$ should be ranked higher than training $x_j$. Then the loss function is

$$L_{ranknet} = -\overline{P}_{ij} log P_{ij} - (1 - \overline{P}_{ij}) log(1 - P_{ij}) \ . \tag{1}$$

The list-wise learning-to-rank approaches tries to directly compare the relevance of list of documents, instead of trying to get one ranking score for each document as pointwise methods do. It is motivated by the fact that the objective of pairwise learning is formalised as minimising errors in ranking document pairs, rather than minimising errors in ranking the document list. For example, LambdaRank [13] is the improved and list-wise version of RankNet. ListNet [17] defined loss function as cross entropy between predicted and ground truth parameterised probability distributions of permutations. ListMLE [18] maps a list of similarity scores to a probability distribution, then utilises the negative log likelihood of ground truth permutations as the loss function. Later, Wang *et al.* [19] applied the ListMLE method to the person re-identification problem.

### 2.2 Person re-identification

Classical person re-identification approaches focus on two key points: developing a powerful feature for image representation and learning an effective metric that represents semantic similarities and dissimilarities between same and different persons.The features used for re-identification are mainly variants of colour histograms [7,8], Local Binary Patterns (LBP) [7,8] or Gabor features [7]. For example, Farenzena *et al.* [4] partitioned the human body into meaningful parts exploiting the asymmetry and symmetry in images. On each part, the weighted colour histogram, the maximally stable colour regions and the recurrent high-structured patches are computed. Ma *et al.* [5] used local descriptors based on colour and gradient information and encode them using high-dimensional Fisher vectors. Mignon *et al.* [6] used a feature vector composed of a mixture of colour

(RGB, HSV and YCbCr) and texture (LBP) from six horizontal stripe regions. Liao *et al*. [12] analyse the horizontal occurrence of local features and maximise this occurrence to improve the robustness of features. The main metric learning methods include Mahalanobis-like metrics like KISSME [9], Local Fisher discriminant Analysis (LFDA) [10], Marginal Fisher Analysis(MFA) [11] and Cross-view Quadratic Discriminant Analysis (XQDA) [12].

Recently, deep learning approaches have achieved state-of-the-art results for person re-identification. Deep learning approaches for person re-identification learn visual feature representations and a similarity metric jointly. Recently, these approaches try to leverage geometric and semantic knowledge that helps the model to focus on specific images regions (e.g. head, torso, legs, feet) by means of semantic segmentation [20, 21] or other attention mechanisms [22–25]. Other approaches improve on the re-identification performance by generating additional artificial training samples (e.g. by Generative Adversarial Networks) [26, 27] and effectively combining them with the real images. Also, some works propose neural architectures and training strategies combining the semi-supervised training with some sort of additional supervision (e.g. from person identities or other factors) [28, 29].A These recently proposed improvements are mostly independent and could eventually be combined with our similarity metric learning approach.

Here we mainly review the related deep learning methods that learn a nonlinear projection into a feature space in which the similarity of pedestrian is well represented. In this regard, several loss functions are proposed or applied in person re-identification. Yi *et al*. [30] first proposed to use a Siamese network to person re-identification. And Ahmed *et al*. [31] and Li *et al*. [32] consider the re-identification task as an image pair classification problem deciding whether an image pair is from the same person or not. One disadvantage of pairwise approaches is the data imbalance since there are much more possible negative pairs than positive pairs in the training set. This may lead to over-fitting when using a pairwise loss function. A class weighting can be applied to solve the problem, but it adds a free parameter that may be different for different datasets. Some approaches use the triplet loss or its variants. The triplet loss forces the similarity between positive matching pairs to be larger than that of negative matching pairs. Ding *et al*. [33] first applied the triplet loss to train a CNN for person re-identification. Cheng *et al*. [34] proposed an improved variant of the triplet loss function by combining the contrastive loss and a CNN network processing parts and the entire body. Chen *et al*. [35] applied a quadruplet loss which samples four images from three identities and minimises the difference between a positive pair from one identity and a negative pair from two different identities and they combine this quadruplet loss with the triplet loss.

Using triplets reduces the imbalance problem in Siamese networks, but another drawback of both Siamese and triplet loss learning is that the trivial pairs or triplets become inactive at a later training stage. To tackle this problem, some methods exploit hard example mining to enhance convergence and overall performance. Ahmed *et al*. [31], for example, used the difference of feature maps to measure the similarity and performed hard negative example pair mining. Shi *et al*. [36] proposed to perform moderate positive and negative example mining to ensure a stable training process and avoid perturbing the manifold learning by using hard examples. On the contrary, Hermans *et al*. [37] proposed to use the hardest positive and negative examples in each training batch to perform an effective triplet learning.

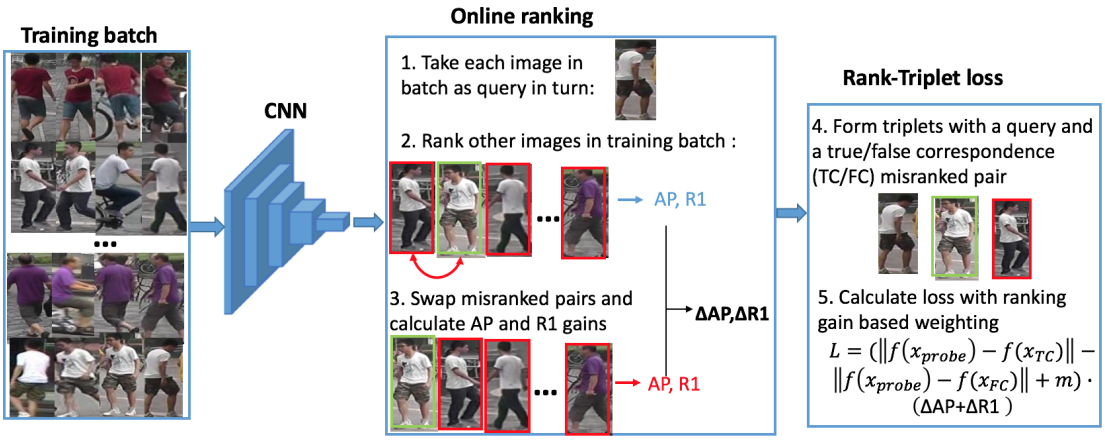


Fig. 2: Overview of the training procedure of the proposed Rank-Triplet approach.

Our approach is inspired by these strategies by selecting mis-ranked examples from the global ordered retrieval results. But instead of simply focusing or retraining on these errors, we principally incorporate the ranking score gain or loss in the global minimisation framework.

## 3 Proposed Method

In the following, we will first outline the basic learning-to-rank method LambdaRank and its limitations in our context. Then, we will present the ranking evaluation measures that we integrate into the training algorithm. And finally, we will describe the proposed training algorithm based on our Rank-Triplet loss and list-wise ranking. An overview of our approach is shown in Fig. 2.

### 3.1 Learning-to-rank with LambdaRank

RankNet [16] uses a neural network model that is trained using the cross entropy cost. Thus, it is minimising the number of pairwise errors and does not consider other information retrieval measures. To directly optimise these measures is difficult as they are not differentiable, thus leading to a non-convex problem that cannot be solved with gradient descent-based algorithms commonly used for neural network models. To tackle this problem, Burges *et al.* [13] proposed LambdaRank which, at each training iteration, simply scales the gradient of the loss function by the difference of the document retrieval evaluation measure Normalised Discounted Cumulative Gain (NDCG) incurred by swapping the rank positions of two items, as shown in Eq. 2. The authors showed that this approach improves the overall ranking performance.

$$\lambda = \frac{\partial L_{ranknet}}{s_{ij}} \cdot \Delta NDCG \,. \tag{2}$$

The triplet learning has shown good performance on verification problems like image classification [38], face recognition [39] and person re-identification [33,34, 37]. However, in triplet learning for person re-identification, we face a similar problem to RankNet. The classical triplet loss is defined on the *partial order* relations among identities, However, the final ranking performance is calculated on the *global* order. That means that the triplet loss iteratively enforces pairwise order relationships w.r.t. reference examples. It ignores the fact that ranking is a prediction task on list of objects. So it is difficult to generalise this approach for optimising the global order. In this regard, a list-wise ranking is a better approximation of this global order relation, and we adapted it to the person re-identification problem, as explained in Section 3.3.

### 3.2 Person re-identification evaluation measure

Cumulated Matching Characteristics (CMC) and mean average precision (mAP) are widely used performance measures for person re-identification. CMC evaluates the top n nearest images in the gallery set w.r.t. one probe image. If a correct match of a query image is at the $k^{th}$ position (k$\leqslant$n), then this query is considered a success of rank n. In most cases, we look at the success of rank 1 (R1), *i.e.* the person has been correctly re-identified. The CMC curve shows the probability that a query identity appears in different-sized ordered candidate lists. As for mAP, for each query, we calculate the area under the Precision-Recall curve, which is known as average precision (AP):

$$AP = \int_0^1 P(R)\,\mathrm{d}R\,, \tag{3}$$

where $P(R)$ is the precision for a given recall $R$. Then, the mean value of $AP$ of all queries, *i.e.* $mAP = \frac{1}{n}\sum_n^{i=1} AP_i$, where $n$ is the number of queries, which considers both precision and recall of an algorithm, thus providing a more suitable evaluation for the setting in which there are several true correspondences in gallery set.

Since $P(\cdot)$ and $R(\cdot)$ are discrete functions, the area under the precision-recall curve is approximated as [40]:

$$AP = \sum_{k=1}^{N} \frac{p(k) + p(k-1)}{2}[r(k) - r(k-1)]\,, \tag{4}$$

where $k$ is the rank in the sequence of retrieved items. $p(k)$ and $r(k)$ are respectively the precision and recall at the rank $k$ position. We define also $p(0) = 1$ and $r(0) = 0$. $N$ is the number of images in the gallery set.

Since in our method the $AP$ is calculated at each iteration during training, we propose to simplify this computation. In ranking problems, recall is the fraction of the items that are relevant to the query that are successfully retrieved, the variation $r(k)-r(k-1)$ is different from zero only when a relevant item is retrieved through the sequence of retrieved items. We only need to take into account the true correspondence ranking position and the variation of recall equals always $\frac{1}{M}$,

where $M$ is the number of the true correspondences of a query. Thus AP can be calculated as:

$$AP = \frac{1}{2M}[1 + p(\pi_1) + \sum_{i=2}^{M} p(\pi_i) + p(\pi_{i-1}))] \ , \tag{5}$$

where $\pi_i$ is the rank index of the $i^{th}$ true correspondence. Precision is defined as the proportion of retrieved non-relevant items out of all non-relevant items available. Thus the precision at ranking position $\pi_i$ is: $p(\pi_i) = \frac{i}{\pi_i}$. We can further simplify the equation to:

$$AP = \frac{1}{M}\sum_{i=1}^{M}[\frac{i}{\pi_i}] - \frac{1}{2\pi_M} + \frac{1}{2M} \ . \tag{6}$$

### 3.3 Rank-Triplet loss

The triplet loss uses triplets of examples to train the network with an anchor image $a$, a positive image $p$ from the same person as $a$ and a negative image $n$ from a different person. Training imposes that the projection of the positive example is placed closer to the anchor than the projection of the negative example. This constraint is defined as:

$$\|f(a_i) - f(p_i)\|_2^2 < \|f(a_i) - f(n_i)\|_2^2 \ . \tag{7}$$

The weights of the network for the three input images are shared, and to train the network, the constraint of Eq. 7 is formulated as the minimisation of the following triplet loss function is minimised:

$$E_{triplet} = -\frac{1}{N}\sum_{i=1}^{N}[max(\|f(a_i) - f(p_i)\|_2^2 - \|f(a_i) - f(n_i)\|_2^2 + m, 0)] \ , \tag{8}$$

where $N$ is the number of triplets, $f$ is the projection of the network, and $m$ is a margin. With the triplet loss function, the network learns a semantic distance metric by "pushing" the negative image pairs apart and "pulling" the positive images closer in the feature space.

In order to update the weights of the network, it is crucial to select triplets that violate the triplet constraint in Eq. 7. However, in practice, the majority of the triplets does not violate the constraint at a later learning stage. Hard triplet mining is an effective way to tackle this problem, but some too hard triplets may distort the manifold [16]. We propose to select the triplets according to the ranking order, *i.e.* only mis-ranked matches will be selected. Given a query and a ranking, an example is declared *mis-ranked* if a wrong match is ranked before it (*i.e.* considered more similar to the query). Not only using the hardest examples stabilises the training, and weighting the triplets according to their contribution makes the learning more effective.

The overall training procedure is presented in Algorithm 1. In order to optimise directly the AP and R1 scores, we estimate the gain for AP and R1 of the triplets from the ranking within a training batch. A training batch is formed by M images of N identities. For each example in the batch, we preform a ranking among the

**Algorithm 1:** Similarity learning with Rank-Triplet loss

**Input:** Training image set, identity label set, learning rate $\lambda$
**Output:** The network weights W

Initialise W **for** $t = 1...T$ **do**
  Randomly sample M identities
  Randomly sample K images for each identity
  Form the training batch with images and identity labels $X = \{x_i\}_{i=1}^{KM}, ID = \{Id_i\}_{i=1}^{KM}$
  Forward pass to obtain image embeddings $Y = \{y_i\}$:
  $Y = f_w(X)$
  L = 0
  **for** $i = 1...KM$ **do**
    $\mathcal{D} \leftarrow dist(y_i, y_{j=1...KM, j\neq i})$
    **foreach** $j$ *that* $Id_i = Id_j$ **do**
      $\mathcal{D}_j \leftarrow \mathcal{D}_j + margin$
    **end**
    $\mathcal{R} \leftarrow sort(\mathcal{D})$
    $AP, R1 \leftarrow calculateAPR1(\mathcal{R})$
    **foreach** $j$ *with* $Id_i = Id_j$ **do**
      **foreach** $k$ *with* $\mathcal{R}_k < \mathcal{R}_j$ *and* $Id_k \neq Id_j$ **do**
        $\mathcal{R}' \leftarrow swap(\mathcal{R}_k, \mathcal{R}_j)$
        $AP', R1' \leftarrow calculateAPR1(\mathcal{R}')$
        $eval\_gain \leftarrow AP' - AP + R1' - R1$
        $L \leftarrow L + (\mathcal{D}_j - \mathcal{D}_k) \times eval\_gain$
      **end**
    **end**
  **end**
  $Loss \leftarrow \frac{L}{N}$
  $W^t \leftarrow W^{t-1} - \lambda \frac{\partial Loss}{\partial W}$
**end**
Return W

rest of images in the batch. For the sake of a robust metric, we add a margin $m$ to the distance of ranking positions between the true correspondences and the probe before ranking. The AP and R1 scores are computed for each query ranking. Then, with respect to one probe, we form all possible mis-ranked pairs (false correspondences ranked before the true correspondence), and we re-calculate the new AP and R1 scores by swapping positions of the pair in the ranking and thus obtain the gains $\Delta AP$ and $\Delta R1$, respectively. The loss of each triplet is weighted by the sum of these gains. The final Rank-Triplet loss is calculated as follows:

$$E_{Rank-Triplet} = \frac{1}{MN}\sum_{i=1}^{MN}\frac{1}{K_i}\sum_{j\in TC_i}\sum_{\substack{k\in FC_i \\ r_k^i < r_j^i}}[\|f(x_i) - f(x_j)\|_2^2 - \|f(x_i) - f(x_k)\|_2^2 + m] \cdot (\Delta AP_{jk}^i + \Delta R1_{jk}^i), \quad (9)$$

where $x_i$ is the $i^{th}$ training example in a training batch, $K_i$ is the number of mis-ranked pairs w.r.t. the $i^{th}$ example as query, and $r_j^i$ is the rank of the $j^{th}$ example w.r.t. the $i^{th}$ image as query. $TC_i/FC_i$ is the true/false correspondence set of the $i^{th}$ example. $\Delta AP_{jk}^i$ is the gain of AP by swapping the $j^{th}$ and $k^{th}$ examples w.r.t. the $i^{th}$ example as query and analogously for R1.

With our evaluation based weighting, we make a trade-off between the moderate hard examples and hardest examples, i.e. more weight is given to the hardest examples to make the learning efficient, and, at the same time, the less hard example are used to stabilise the training.

## 4 Experiments and results

In this section, we report the experiment results carried out on the person re-identification datasets Market-1501 [40], DukeMTMC-Reid [41] and CUHK03 [32] to compare our approach with the state-of-the-art approaches. We also perform a comparison with other loss functions commonly used in the literature. We further perform a more detailed analysis of the proposed method in several aspects, like its convergence behaviour and training time. Finally, to show the genericity of our approach, we applied it to an image retrieval task. We performed experimental evaluations on the Holidays dataset and compared it to the state-of-the-art methods and results.

### 4.1 Person re-identification

#### *4.1.1 Datasets*

The Market-1501 dataset [40] is one of the largest publicly available datasets 130 for human re-identification with 32,668 annotated bounding boxes of 1501 subjects. All images are resized to $128 \times 48$. The dataset is split into 751 identities for training and 750 identities for testing as in [40].

The DukeMTMC-Reid dataset [41] is collected with 8 cameras and used for cross-camera tracking. It contains 36,411 total bounding boxes from 1,404 identities. Half is used for training and the rest for testing. In total, it has 36,411 total bounding boxes including 16,522 training images, 2,228 queries, and 17,661 gallery images.

The CUHK03 dataset [32] is a challenging dataset collected in the CUHK campus with 13,164 images of 1,360 identities from two camera views. Each identity is captured by two disjoint camera views and has an average of 4.8 images 140 in each view. There are two settings: labelled with human-annotated bounding boxes and the more challenging detected with automatically generated bounding boxes. Our experiments will be conducted on the detected version of CUHK03 which is a more realistic scenario. We followed the new test protocol proposed in [42] which splits the CUHK03 dataset into training set and testing set similar to that of Market-1501, which consist of 767 identities and 700 identities respectively.

Some examples images of the datasets are shown in Fig 3. All the three datasets follow the same test protocol. The authors randomly select one image from each camera as the query for each identity and use the rest of images to construct the gallery set. In evaluation, true matched images captured from the same camera as the query are not considered. Thus, these images have no influence on the re-identification accuracy.

(a) Market-1501

(b) DukeMTMC-Reid

(c) CUHK03

Fig. 3: Some image examples from different person re-identification datasets

*4.1.2 Implementation Details*

We used Alexnet [43] and Resnet-50 [44] as the model architecture and the weights pre-trained on the ImageNet dataset are used as initialisation. We replaced the final layer of the models by a fully-connected layer with 256 output dimensions. Each input image is resized to 256 $\times$128 pixels. Data augmentation is performed by randomly flipping the images and cropping central regions with random perturbation. Adam optimiser is used and the initial learning rate is set to $10^4$. Each 80 epochs the learning rate is decreased by a factor of 0.1. The weight decay is set to 0.0005. The training is performed in 200 epochs. And the batch size is set to 128 from 32 identities with 4 images each. The 32 identities are randomly selected without replacement. In principle, for better balance in a training batch, we recommend to use the same number of images per person. We also require at least 2 images per person, in order to form the positive pairs in the loss function. The 4 images provide more flexibility in forming positive pairs while still giving enough possibilities for wrong rankings (negative pairs).

*4.1.3 Experimental results*

**Comparison of different neural network models**. To demonstrate the effectiveness of our approach, the well-known Alexnet and Resnet models have been

| | Resnet | | Alexnet | |
|---|---|---|---|---|
| | R1 | mAP | R1 | mAP |
| Hardbatch | 81.0 | 63.9 | - | - |
| Baseline | 82.1 | 66.5 | 70.9 | 47.3 |
| Rank-Triplet | **83.6** | **67.3** | **72.7** | **49.1** |

Table 1: Re-identification performance on the Market-1501 dataset in terms of rank 1 (R1) and mean average precision (mAP) (in %) for different loss functions and neural network models.

| Loss function | R1 | mAP |
|---|---|---|
| Classification loss | 74.3 | 51.0 |
| Siamese loss | 62.9 | 46.6 |
| Triplet loss | 74.3 | 56.5 |
| Quadruplet loss | 74.9 | 58.1 |
| Hardbatch* | 81.0 | 63.9 |
| Baseline | 82.1 | 66.5 |
| Rank-Triplet | **83.6** | **67.3** |

Table 2: Re-identification results(in %) on Market-1501 with different loss functions.*: The result of our re-implementation of [37]. To notice that we did not use the same training parameters and fc layer settings as [37] and in [37], the test data augmentation is performed.

used in our experiments as they have been proven very successful for various computer vision applications. For training both models, we implemented the Hardbatch triplet loss (Eq. 11), a baseline that is our method without (Eq. 9) the ranking-based weighting term and our complete approach, called Rank-Triplet (cf. Eq. 9). Table 1 shows the re-identification results on the Market-1501 dataset. The Resnet50-based model trained with Rank-Triplet shows a better performance than the Alexnet-based model by a margin of 9.9% point for R1 and 18.2% points for mAP. Integrating our evaluation measure gain weighting of Rank-Triplet raised the mAP by 1.8% points and 0.8% points and R1 by 1.8% points and 1.5% points with Alexnet and Resnet respectively. The Hardbatch approach with Alexnet cannot converge on the Market-1501 dataset. This demonstrates that hard example mining can make the learning more effective, but only using the hardest examples may severely perturb the learning process.

**Comparison of different loss functions**. We conducted experiments with different loss results are shown in Table 2. For the supervised classification with identity labels, the softmax cross entropy loss is used. The margin in the Siamese loss and triplet loss is fixed to the default value m =1. For the pairwise Siamese learning the contrastive loss is used, we generate all possible pairs of images within a batch. The loss is calculated as follows:

$$L_{contrastive} = \frac{1}{N}\sum_{i=1}^{N} l\|f(x_i)-f(x_j)\|_2^2+(1-l)\max(m-\|f(x_i)-f(x_j)\|_2^2,0)\ , \quad (10)$$

where $N$ is the number of pairs, $x_i, x_j$ are feature embeddings for two images, $l$ is 1 for images from the same person (positive pairs) and 0 for images from different persons (negative pair).

The triplet loss is calculated according to Eq. 8. And the Hardbatch triplet loss is computed using only the hardest positive image and negative image with respect to a query $x_i$:

$$L_{Hardbatch} = \frac{1}{N}\sum_{i=1}^{N} max(\max_{j\in TC_i} \|f(x_i) - f(x_j)\|_2^2 - \min_{k\in FC_i)} \|f(x_i) - f(x_k)\|_2^2 + m, 0) \ , \quad (11)$$

where $N$ is the number of triplets, $TC_i/FC_i$ is the true/false correspondence set of the $i$-th example.

The quadruplet loss in [35] propose to, based on triplets, pushes away also negative pairs from positive pairs w.r.t different probe images. The loss is formulated as:

$$E_{quadruplet} = -\frac{1}{N}\sum_{i=1}^{N}[max(\|f(x_i) - f(x_j)\|_2^2 - \|f(x_i) - f(x_k)\|_2^2 + m_1, 0) + max(\|f(x_i) - f(x_j)\|_2^2 - \|f(x_k) - f(x_l)\|_2^2 + m_2, 0)], \quad (12)$$

where $x_j$ is the feature embeddings of an image from the same identity as $x_i$ and $x_k$,$x_l$ are from different identities. As [35], we set the $m_1 = 1, m_2 = 0.5$.

Finally, we implement a baseline with the Rank-Triplet loss function (Eq. 9) without the term of evaluation gain weighting. But the triplet selection is still based on online ranking orders.

Rank-Triplet achieved the best performance among these loss functions. The Rank-Triplet improves the baseline by a margin of 1.5% points for R1 and 0.8% points for mAP. This shows the effectiveness of the list-wise evaluation measure-based weighting. The Rank-Triplet gives also better results than the Hardbatch. This shows that using moderate difficult examples and weighting them helps the metric learning. In fact, Hardbatch is a particular weighting with weight=1 given to the hardest examples and 0 given to the rest. Also Hardbatch shows better performance than the normal triplet loss, confirming the effectiveness of the hardest example mining in [31,37]. Using the quadruplet loss also slightly improves the performance with respect to triplets. This could eventually be combined with our loss.

**Comparison with state-of-the-art methods**. We compared our method with state-of-the-art methods on the three benchmark datasets. The results are shown in Table 3. We followed the same standard evaluation protocol for all the compared methods and datasets.

Our method Rank-Triplet achieves better results than most of the other methods on the three benchmarks. Only on Market-1501, HA-CNN obtains a better R1 score with a lower mAP, and DPFL and HA-CNN achieve a better R1 score on DukeMTMCReid with a much lower mAP also. This could be due to the multi-scale approach of DPFL and an effective attention-mechanism in HA-CNN. Note that these techniques could also easily be applied to our model. Also the fact that the R1 score is slightly higher but not the mAP indicates the overall ranking produced by our method is generally more consistent and accurate. For the CUHK03 dataset, the performance of these methods is clearly below the one of our Rank-Triplet approach. This may be explained by the smaller number of images per

| | Market-1501 | | DukeMTMC-Reid | | CUHK03-NP | |
|---|---|---|---|---|---|---|
| Methods | R1 | mAP | R1 | mAP | R1 | mAP |
| Hardbatch triplet loss [37] | 81.0 | 63.9 | 62.8 | 42.7 | 46.4 | 50.6 |
| Baseline | 82.1 | 66.5 | 72.4 | 52.0 | 45.3 | 48.9 |
| Rank-Triplet loss | 83.6 | 67.3 | 74.3 | 55,6 | 47.8 | 52.4 |
| Rank-Triplet+re-rank [42] | 86.2 | **79.8** | 78.6 | **71.4** | **60.4** | **60.8** |
| LOMO+XQDA [12] | 43.8 | 22.2 | 30.8 | 17.0 | 12.8 | 11.5 |
| LSRO [41] | 78.1 | 56.2 | 67.7 | 47.1 | - | - |
| Divide and fuse [45] | 82.3 | 72.4 | - | - | 30.0 | 26.4 |
| K-reciprocal re-rank [42] | 77.1 | 63.6 | - | - | 34.7 | 37.4 |
| ACRN[46] | 83.6 | 62.6 | 72.6 | 52.0 | - | - |
| SVDNet [47] | 82.3 | 62.1 | 76.7 | 56.8 | 41.5 | 37.3 |
| JLML [48] | 85.1 | 65.5 | - | - | - | - |
| DPFL [49] | 88.6 | 72.6 | 79.2 | 60.6 | 40.7 | 37.0 |
| MGCAM [22] | 83.6 | 74.3 | - | - | 46.7 | 46.9 |
| AACN [24] | 85.9 | 66.9 | - | - | - | - |
| HA-CNN [23] | **91.2** | 75.7 | **80.5** | 63.8 | 41.7 | 38.6 |

Table 3: Comparison with state-of-the-art methods on person re-identification.

person in CUHK03 (4.8 images an average), which may not be enough for training these large models. Note that, as many state-of-the-art approaches, we employed re-ranking [42, 45] which uses information from nearest neighbours in the gallery and significantly improves the performance.

On the CUHK03 benchmark, our methods achieves the superior results. Since EDPF and SVDNet are based on classification loss. The CUHK03 datasets contains less images per person. That is not enough to train a good classifier. However, triplet loss is not much affected because we could still form a large number of triplets even there's less images per person. Since the main contribution of these two state-or-the-art methods focus on the network architecture, their methods could eventually combine with our loss function.

The Hardbatch triplet learning on DukeMTMC-Reid had difficulty to converge with an initial learning rate of $10^{-4}$. The convergence improved when the learning rate was reduced to $2 \times 10^{-5}$. But it still gave an inferior final performance.

Some representative successful and failed top-10 Rank-Triplet results are shown in Fig. 4. As can be seen, most of the errors are due to the high clothing similarity among pedestrians and to some partial occlusion. Even for a human, it is difficult to decide if they represent a real match or not.

#### *4.1.4 Analysis of the proposed method*

**Evolution of R1 and mAP during training**. The online calculation of R1 and mAP could be also used as a training index to observe the progression of the training. We further analysed the R1 and mAP values computed for each batch during training. Further, a separate validation batch is formed with 128 random images from 32 persons that were not used for training We evaluate the mAP, R1, number of mis-ranked pairs and loss for each epoch. The curves showing the evolution of these measures during training are shown in Fig. 5.

We can observe that the R1 and mAP computed on the training batches converge to 1 and the number of mis-ranked pairs almost converges to 0. The validation mAP and R1 also increase and converge during training. The number of mis-ranked

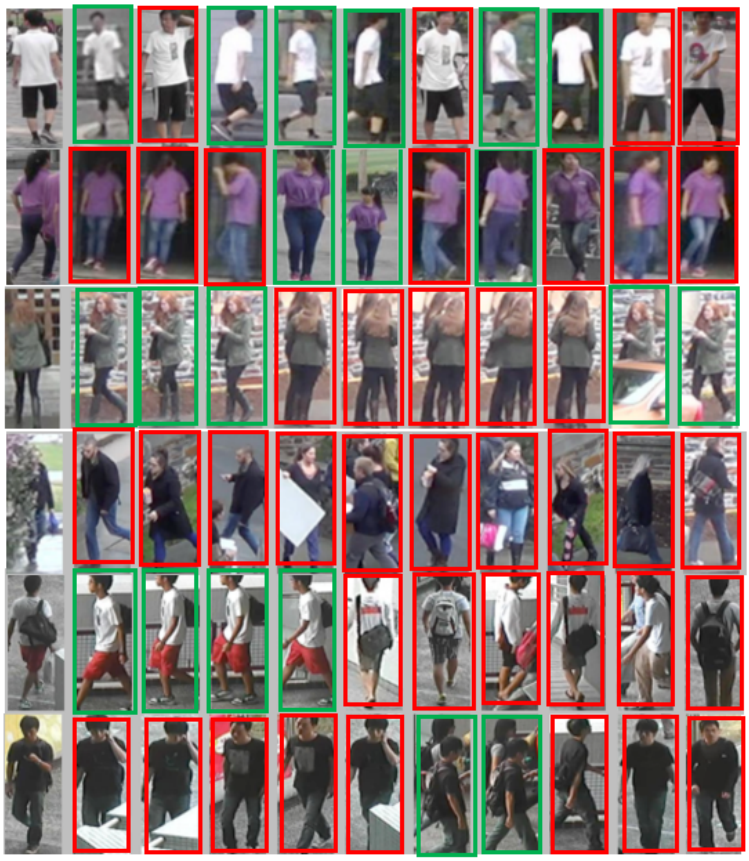

Fig. 4: Some ranking results. The query image is on the left, the true matches are surrounded by green boxes and false matches are surrounded by red boxes. The two top rows are from Market-1501, the tow middle rows are from DukeMTMC-Reid, and the two bottom rows are from CUHK03.

pairs decrees with fluctuation, and finally converges. The validation loss naturally increases because the loss is the average among the mis-ranked pairs, and after solving the simple cases, it remains only the mis-ranked pairs giving a high loss with a large evaluation weighting. Thus, validation loss is not suitable to observe the training process, as discussed In [37]. The online validation mAP and R1 that we proposed to observe can be better training index.

**Training time analysis**. In order to analyse the complexity of Rank-Triplet, we compared its training time with the one of Hardbatch. The results are shown in Table 4. All algorithms are implemented in Pytorch. The training was performed with Intel i7-5930K 3.50GHZ CPU and 2 Nvidia GTX Titan Maxwell GPUs. The Hardbatch triplet loss has first been implemented with Eq. 11. Surprisingly, this implementation takes about 1 hour more for training than our Rank-Triplet. A probable explanation is that the Hinge function slowed down the training. That means, even if the triplets do not violate the constraint, the gradient is still calcu-

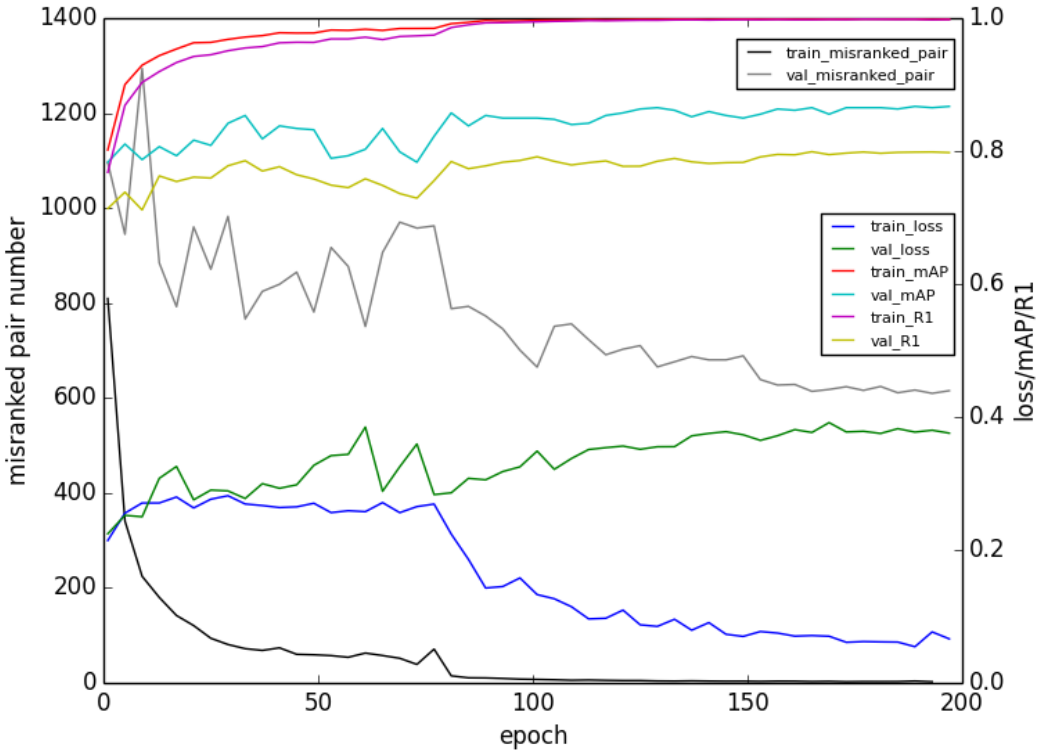


Fig. 5: Evolution of training and validation mAP, R1 and number of mis-ranked pairs.

| Loss | Training time |
|---|---|
| Hardbatch hinge triplet loss | 6h10min |
| Hardbatch triplet loss with condition flow | 4h25min |
| Rank-Triplet loss | 5h04min |

Table 4: Training time on Market-1501.

lated on these triplets. Then we optimise the Hardbatch code by adding a condition flow in order to calculate the loss only for the necessary triplets. The training time for Rank-Triplet is considerably reduced. After this optimisation, the Hardbatch takes about 40 minutes less than the Rank-Triplet loss. That difference roughly corresponds to the time of ranking, evaluation measure computation and the use of more triplets at each iteration. However, we consider that this is a reasonable extra cost given the overall performance improvement.

| Method | mAP (in%) |
|---|---|
| Neural codes [51] | 75.9 |
| R-MAC [53] | 85.2 |
| NetVLAD [54] | 83.1 |
| Cross-dimension weighting [55] | 84.9 |
| Hard Siamese [56] | 82.5 |
| ROI-Triplet [52] | **90.7** |
| Baseline | 85.1 |
| Rank-Triplet | 85.8 |

Table 5: Experimental evaluation on the Holidays dataset.

### 4.2 Image retrieval

To further evaluate our method, we tested our Rank-Triplet loss on a more general content-based image retrieval problem, where the task is to retrieve images from a gallery set that belong to the same category as the probe image or are similar to it. As with the person re-identification task, the challenges are translation, rotation and scaling transformations of the objects of interest in the images and also illumination changes. We use the INRIA Holidays [50] dataset to perform the evaluation. Images are considered from the same category/class, *i.e.* relevant to a specific query, if they are taken in the same scene or showing the same object under different viewpoints. The dataset contains 500 queries and 991 corresponding relevant images. For the training, we used the landmark dataset as in [51,52]. However, we were only able to use a subset of the dataset due to broken URLs. In total, we used 28777 images of 560 landmarks for training. For the training and the test, the input images are randomly cropped to 320×320 from 362×362. Since there is a high variance of translation and scale of relevant objects inside the images, we replaced the last global average pooling by a global max-pooling as in [53]. All other experimental settings remain the same. Table 5 shows the comparison with the state-of-the-art methods and with the baseline. Our method performs slightly better than the baseline and is superior to most state-of-the-art results. The ROI-triplet method uses also a triplet network and integrates a pre-trained ROI pooling to localise the salient image content. This technique could also be integrated in our model to further improve the instance retrieval performance.

## 5 Conclusion

In this paper, we proposed a new learning-to-rank approach to perform similarity learning with images using Convolution Neural Networks. We introduced a novel list-wise loss function directly integrating ranking evaluation measures inspired by the idea of LambdaRank. An online ranking within training batches is performed to evaluate the importance of different triplets composed of probe, mis-ranked true and false correspondences and to weight the loss with the rank improvement for a given query. We experimentally showed that taking into account the evaluation measures during training and calculating the loss in a list-wise way improves the overall ranking and recognition performance on the given task of person re-identification. Further, our proposed loss function outperforms other common functions in the literature and achieved state-of-the-art results on three different

benchmarks. Finally, we applied the proposed approach to a more general image retrieval problem with photographs of very diverse content. Without any major modifications, our algorithm outperformed most state-of-the-art methods on the Holiday benchmark showing the general applicability of our approach.

## References


1. Beyer, L., Breuers, S., Kurin, V., Leibe, B.: Towards a principled integration of multi-camera re-identification and tracking through optimal bayes filters. In: International Conference on Computer Vision Workshops (2017)
2. Chaaraoui, A.A., Padilla-López, J.R., Ferrández-Pastor, F.J., Nieto-Hidalgo, M., Flórez-Revuelta, F.: A vision-based system for intelligent monitoring: Human behaviour analysis and privacy by context. Sensors **14**(5), 8895–8925 (2014)
3. Vezzani, R., Baltieri, D., Cucchiara, R.: People reidentification in surveillance and forensics: A survey. ACM Computing Surveys (CSUR) **46**(2), 29 (2013)
4. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person re-identification by symmetry-driven accumulation of local features. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 2360–2367. IEEE (2010)
5. Ma, B., Su, Y., Jurie, F.: Local descriptors encoded by fisher vectors for person re-identification. In: Proceedings of the European Conference on Computer Vision, pp. 413–422 (2012)
6. Mignon, A., Jurie, F.: Pcca: A new approach for distance learning from sparse pairwise constraints. In: Computer Vision and Pattern Recognition (CVPR), pp. 2666–2672. IEEE (2012)
7. Li, W., Wang, X.: Locally aligned feature transforms across views. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 3594–3601 (2013)
8. Zhao, R., Ouyang, W., Wang, X.: Person re-identification by salience matching. In: Proceedings of the International Conference on Computer Vision, pp. 2528–2535 (2013)
9. Koestinger, M., Hirzer, M., Wohlhart, P., Roth, P.M., Bischof, H.: Large scale metric learning from equivalence constraints. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 2288–2295 (2012)
10. Pedagadi, S., Orwell, J., Velastin, S., Boghossian, B.: Local fisher discriminant analysis for pedestrian re-identification. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 3318–3325 (2013)
11. Xiong, F., Gou, M., Camps, O., Sznaier, M.: Person re-identification using kernel-based metric learning methods. In: Proceedings of the European Conference on Computer Vision, pp. 1–16 (2014)
12. Liao, S., Hu, Y., Zhu, X., Li, S.Z.: Person re-identification by local maximal occurrence representation and metric learning. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (2015)
13. Burges, C.J., Ragno, R., Le, Q.V.: Learning to rank with nonsmooth cost functions. In: Neural Information Processing Systems, pp. 193–200 (2007)
14. Herbrich, R.: Large margin rank boundaries for ordinal regression. Advances in large margin classifiers pp. 115–132 (2000)
15. Prosser, B.J., Zheng, W.S., Gong, S., Xiang, T., Mary, Q.: Person re-identification by support vector ranking. In: Proceedings of the British Machine Vision Conference (BMVC) (2010)
16. Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., Hullender, G.: Learning to rank using gradient descent. In: Proceedings of the International Conference on Machine Learning, pp. 89–96 (2005)
17. Cao, Z., Qin, T., Liu, T.Y., Tsai, M.F., Li, H.: Learning to rank: from pairwise approach to listwise approach. In: Proceedings of the International Conference on Machine Learning, pp. 129–136 (2007)
18. Xia, F., Liu, T.Y., Wang, J., Zhang, W., Li, H.: Listwise approach to learning to rank: theory and algorithm. In: Proceedings of the International Conference on Machine Learning, pp. 1192–1199 (2008)

19. Wang, J., Wang, Z., Gao, C., Sang, N., Huang, R.: Deeplist: Learning deep features with adaptive listwise constraint for person reidentification. IEEE Transactions on Circuits and Systems for Video Technology **27**(3), 513–524 (2017)
20. Kalayeh, M.M., Basaran, E., Gökmen, M., Kamasak, M.E., Shah, M.: Human semantic parsing for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
21. Tian, M., Yi, S., Li, H., Li, S., Zhang, X., Shi, J., Yan, J., Wang, X.: Eliminating background bias for robust person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
22. Song, C., Huang, Y., Ouyang, W., Wang, L.: Mask-guided contrastive attention model for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
23. Li, W., Zhu, X., Gong, S.: Harmonious attention network for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
24. Xu, J., Zhao, R., Zhu, F., Wang, H., Ouyang, W.: Attention-aware compositional network for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2119–2128 (2018)
25. Si, J., Zhang, H., Li, C.G., Kuen, J., Kong, X., Kot, A.C., Wang, G.: 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 (2018)
26. Huang, Y., Xu, J., Wu, Q., Zheng, Z., Zhang, Z., Zhang, J.: Multi-pseudo regularized label for generated samples in person re-identification. IEEE Transactions on Image Processing **28**(3), 1391–1403 (2019)
27. Zheng, Z., Yang, X., Yu, Z., Zheng, L., Li, Y., Kautz, J.: Joint discriminative and generative learning for person reidentification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2019)
28. Zheng, Z., Zheng, L., Yang, Y.: A discriminatively learned CNN embedding for person reidentification. ACM Transactions on Multimedia Computing, Communications and Applications **14**(1) (2017)
29. Chang, X., Hospedales, T.M., Xiang, T.: Multi-level factorisation net for person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
30. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Deep metric learning for person re-identification. In: Proceedings of the IEEE International Conference on International Conference on Pattern Recognition, pp. 34–39 (2014)
31. Ahmed, E., Jones, M., Marks, T.K.: An improved deep learning architecture for person re-identification. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 3908–3916 (2015)
32. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid:deep filter pairing neural network for person re-identification. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 152–159 (2014)
33. Ding, S., Lin, L., Wang, G., Chao, H.: Deep feature learning with relative distance comparison for person re-identification. Pattern Recognition **48**(10), 2993–3003 (2015)
34. Cheng, D., Gong, Y., Zhou, S., Wang, J., Zheng, N.: Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1335–1344 (2016)
35. Chen, W., Chen, X., Zhang, J., Huang, K.: Beyond triplet loss: a deep quadruplet network for person re-identification. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, vol. 2 (2017)
36. Shi, H., Yang, Y., Zhu, X., Liao, S., Lei, Z., Zheng, W., Li, S.Z.: Embedding deep metric for person re-identification: A study against large variations. In: European Conference on Computer Vision, pp. 732–748 (2016)
37. Hermans, A., Beyer, L., Leibe, B.: In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737 (2017)
38. Wang, J., Leung, T., Rosenberg, C., Wang, J., Philbin, J., Chen, B., Wu, Y., et al.: Learning fine-grained image similarity with deep ranking. In: CVPR (2014)
39. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 815–823 (2015)

40. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: International Conference on Computer Vision, pp. 1116–1124 (2015)
41. Zheng, Z., Zheng, L., Yang, Y.: Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In: International Conference on Computer Vision (2017)
42. Zhong, Z., Zheng, L., Cao, D., Li, S.: Re-ranking person re-identification with k-reciprocal encoding. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (2017)
43. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)
44. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
45. Yu, R., Zhou, Z., Bai, S., Bai, X.: Divide and fuse: A re-ranking approach for person re-identification. In: Proceedings of the British Machine Vision Conference (BMVC) (2017)
46. Schumann, A., Stiefelhagen, R.: Person re-identification by deep learning attribute-complementary information. In: Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1435–1443. IEEE (2017)
47. Sun, Y., Zheng, L., Deng, W., Wang, S.: Svdnet for pedestrian retrieval. In: International Conference on Computer Vision (2017)
48. Li, W., Zhu, X., Gong, S.: Person re-identification by deep joint learning of multi-loss classification. In: International Joint Conference on Artificial Intelligence (2017)
49. Chen, Y., Zhu, X., Gong, S.: Person re-identification by deep learning multi-scale representations. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 2590–2600 (2017)
50. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: European conference on computer vision, pp. 304–317 (2008)
51. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural codes for image retrieval. In: European Conference on Computer Vision, pp. 584–599 (2014)
52. Gordo, A., Almazán, J., Revaud, J., Larlus, D.: Deep image retrieval: Learning global representations for image search. In: European Conference on Computer Vision, pp. 241–257 (2016)
53. Tolias, G., Sicre, R., Jégou, H.: Particular object retrieval with integral max-pooling of cnn activations. In: International Conference on Learning Representations (2016)
54. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5297–5307 (2016)
55. Kalantidis, Y., Mellina, C., Osindero, S.: Cross-dimensional weighting for aggregated deep convolutional features. In: European Conference on Computer Vision, pp. 685–701 (2016)
56. Radenović, F., Tolias, G., Chum, O.: Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In: European Conference on Computer Vision, pp. 3–20 (2016)