This WACV 2021 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Set Augmented Triplet Loss for Video Person Re-Identification

Pengfei Fang[1,2], Pan Ji[3*], Lars Petersson[2], Mehrtash Harandi[4]
[1]Australian National University, [2]DATA61-CSIRO, [3]OPPO US Research Center, [4]Monash University
Pengfei.Fang@anu.edu.au, peterji1990@gmail.com
Lars.Petersson@data61.csiro.au, mehrtash.harandi@monash.edu

Abstract

*Modern video person re-identification (re-ID) machines are often trained using a metric learning approach, supervised by a triplet loss. The triplet loss used in video re-ID is usually based on so-called clip features, each aggregated from a few frame features. In this paper, we propose to model the video clip as a set and instead study the distance between sets in the corresponding triplet loss. In contrast to the distance between clip representations, the distance between clip sets considers the pair-wise similarity of each element (*i.e.*, frame representation) between two sets. This allows the network to directly optimize the feature representation at a frame level. Apart from the commonly-used set distance metrics (*e.g.*, ordinary distance and Hausdorff distance), we further propose a hybrid distance metric, tailored for the set-aware triplet loss. Also, we propose a hard positive set construction strategy using the learned class prototypes in a batch. Our proposed method achieves state-of-the-art results across several standard benchmarks, demonstrating the advantages of the proposed method.*

1. Introduction

Person re-identification (re-ID) has drawn an increasing amount of attention in academia and industry due to its great potential in research and real-world applications [43]. A person re-ID machine is trained to regress a non-linear function which maps the pedestrian images to a semantically meaningful embedding space. In such an embedding space, feature vectors extracted from images belonging to the same identity (ID) are clustered, thereby retrieving correct matches for unseen query images of persons in the gallery. In the past decades, image re-ID has achieved significant improvements via learning discriminative representations from a single image [28, 29, 4, 12]. Recently, video person re-ID has attracted a growing interest as videos provide richer cues in terms of encoding video representations for person retrieval [15, 41, 6, 26, 3]. In this paper, we aim to create compact yet discriminative features from videos for accurate video re-ID.

The pipeline of training a typical video re-ID machine consists of first extracting the frame-level features with the help of a deep network backbone and then aggregating them to a clip-level feature. In video re-ID, the ranking task (*i.e.*, triplet loss) is a popular choice to supervise the network to learn an embedding space, w.r.t. the clip-level features. This, however, could lead to sub-optimal learning of the video embedding space, as the aggregation operation to frame features will result in loss of information of the original frame features. Specifically, in the video-based applications, the triplet loss considers the distance between the clip representations (*i.e.*, d^{an} and d^{ap} in Fig. 1(a)), which only indirectly penalizes the hard frames between the clips (*i.e.*, hard positive frames and hard negative frames in Fig. 1(a)). This observation motivates us to directly leverage the frame features, to decrease the hard positive distance (*i.e.*, $\leftrightarrow$ in Fig. 1(a)) and increase the hard negative distance (*i.e.*, $\leftrightarrow$ in Fig. 1(a)) for frame features.

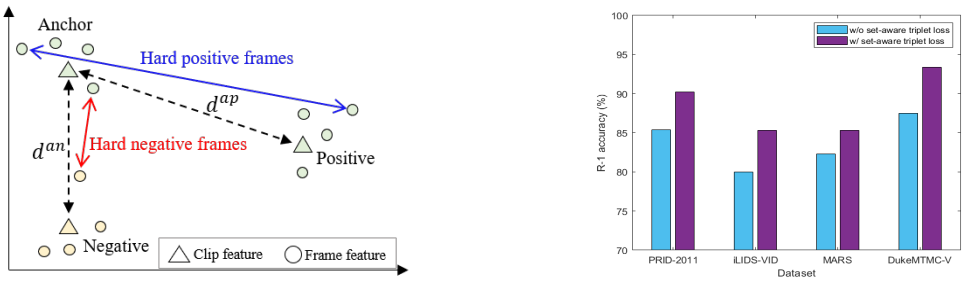
In video re-ID, we often aggregate the frame features (*i.e.*, $\{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_t\}, \boldsymbol{f}_i \in \mathbb{R}^c, i = 1, \ldots, t$) to a clip-level representation (*i.e.*, $\hat{\boldsymbol{f}} \in \mathbb{R}^c$) using an aggregation function (*i.e.*, $\mathrm{Agg}(\cdot)$). This processing can be summarized as:

$$\hat{\boldsymbol{f}} = \mathrm{Agg}(\{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_t\}) = \phi\big(\sum_{i=1}^{t}(\omega_i \boldsymbol{f}_i)\big), \qquad (1)$$

where $\phi(\cdot)$ and $\{\omega_1, \ldots, \omega_t\} \in \mathbb{R}^t$ denote non-linear mapping and aggregation weights, respectively. Due to the summation operator in Eqn. (1), the clip feature (*i.e.*, $\hat{\boldsymbol{f}}$) is invariant to the order of frame features, indicating that the aggregation function is temporally invariant. In other words, the aggregation function acts on sets, in the sense that the response of the aggregation function is "insensitive" to the ordering of elements in the input [38]. With this intuition, we aim to use the theory of sets to make better use of the frame features within each video clip.

In this paper, we propose to model the frame features within a clip as a set and propose to use the distance be-

*Work done while at NEC Laboratories America



(a) (b)

Figure 1. (a): Geometry interpretation of the distance metrics for clip representation and frame representation. The color represents the class of samples. d^{ap} and d^{an} denote the distance from positive pair and negative pair in a clip level. However, those two distances cannot reveal the original distribution of frame features, thereby ignoring the distance between hard frames (*i.e.*, $\leftrightarrow$ for hard negative pair and $\leftrightarrow$ for hard positive pair). (b): The comparison of R-1 accuracy from the networks trained without set-aware triplet loss and with set-aware triplet loss, across four datasets. The backbone network is ResNet-50, pre-trained on ImageNet. In the set-aware triplet loss, we use the proposed hybrid set distance metric to calculate the distance of anchor-positive pair and anchor-negative pair.

tween sets in the triplet loss. Different from the L_2 distance between the aggregated clip features (see Fig. 2(a)), the distance between sets considers every pair-wise distance in two sets and explores more information of the frame features. In set theory, the distance between sets is usually measured by ordinary distance (see Fig. 2(b)) or Hausdorff distance (see Fig. 2(c)). However, these set distance measures cannot fully utilize hard frames (*i.e.*, hard positive and hard negative) in a triplet. To construct an effective set triplet loss, we further propose a hybrid distance metric (see Fig. 2(d)), where the hard frames for anchor-positive and anchor-negative sets are considered explicitly. In essence, our hybrid distance metric aims at penalizing the hard frames between sets (*i.e.*, $\leftrightarrow$ and $\leftrightarrow$ in Fig. 1(a)). Fig. 1(b) shows the comparison of retrieval accuracies from video re-ID models, trained *without* our set-aware triplet loss, and *with* our set-aware triplet loss, across four video re-ID datasets. We further apply the class prototypes to frame-level features to construct hard sets by comparing the similarity between the class prototype and frame feature with the same instance. Then the constructed set acts as a hard positive set.

Contributions. The contributions of this work are summarized as follows:

- We model the video clip as a set[1], and employ the distance metric between sets to construct the triplet loss. Furthermore, we propose a new hybrid set distance metric, which is tailored for the set triplet loss.
- We further model the weights in the last classification layer as class prototypes, to construct a hard positive set, w.r.t. each anchor set with the same identity.
- Our algorithm achieves state-of-the-art performance across four standard video person re-ID datasets (*i.e.*, PRID-2011 [10], iLIDS-VID [32], MARS [42] as well as DukeMTMC-VideoReID [35]), showing the effectiveness of the proposed set augmented triplet loss.

[1]In the remainder of this paper, we will use "clip" and "set" interchangeably

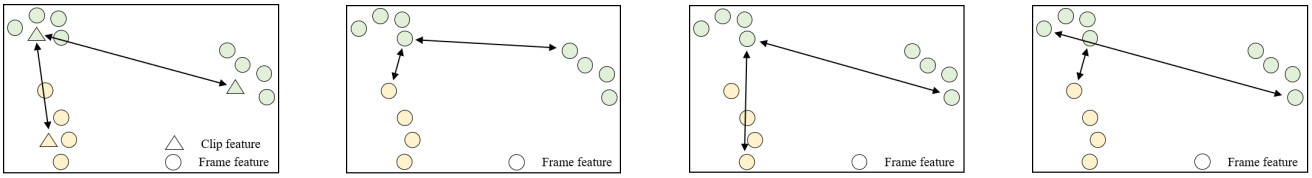
2. Related Work

2.1. Sets

The concept of modeling the training data as a set has appeared in many applications, *e.g.*, point cloud classification [38], image tagging [38], object localization [22] *et al.* In general, the response of set functions is insensitive to the order of the elements in the set and the work in [38] studies the structure of such functions. The most popular function is the pooling operation (*i.e.*, max pooling, average pooling) across the elements of its input. For example, deep Convolutional Neural Networks (CNNs) use pooling layers to summarize the features in a patch [8]. In the point cloud classification task [21], a non-linear function extracts the latent representation of point coordination and the pooling function further summarizes the features of objects. Attention using non-local connections also acts as a set function as the attention weights are produced by pairwise similarities of pixel features [33]. In [22], the locations of objects are estimated by training a detector which minimizes the set distance between the prediction and ground truth of objects.

2.2. Metric Learning

Deep metric learning aims to project images to a low dimensional embedding space, in which the images with similar semantics are clustered together [27, 23, 4]. The most popular paradigm is to employ the triplet loss to penalize the positive pair or negative pair or both of them within a triplet [25]. However, the possible number of triplets is ex-



(a) Clip distance metric (b) Ordinary distance metric (c) Hausdorff distance metric (d) Hybrid distance metric

Figure 2. Geometry interpretation of different distance metrics. (a), (b), (c), and (d) denote L_2 distance metric between clip representation, ordinary distance metric, Hausdorff distance metric, and hybrid distance metric between sets. The color represents the class of samples.

ponential to the number of samples in a mini-batch, leading to a prohibitive computational cost. Much effort has gone into mining the triplets efficiently [9, 4, 27]. For example, the hard mining strategy only selects the hard positive and hard negative for an anchor sample [9]. However, a hard mining strategy often leads to getting caught in local minima during optimization [9]; thus the semi-hard mining method is further proposed to make use of more negative pairs [4]. Beyond mining the triplets in a mini-batch, the work in [27] employs the class signatures to mine hard negative classes for an anchor class in the whole dataset.

2.3. Person Re-identification

Most popular solutions for person re-ID mainly focus on learning an appearance-discriminative representation [43]. In general, the person representation is often encoded by the holistic appearance feature [31], or part features [29] or both of them [4]. Beyond an image-based person re-ID, a video-based re-ID system can make use of additional temporal cues within a couple of frames, thereby encoding a robust video representation [15, 26, 36]. Various temporal modeling methods have been studied extensively to effectively fuse the frame features to encode a discriminative and robust video representation. In [19, 36], a clip-level person representation is modeled by average/max temporal pooling of frame-level features; thereafter, frame features are regressed by a Recurrent Neural Network (RNN), whose final hidden state encodes the representation of the target. The temporal modeling also utilizes the attention mechanism, which aggregates frame features according to individual importance [7, 15]. In [7], individual importance is generated to aggregate frame features in a weighted sum fashion.

In contrast to existing works, our work utilizes the original frame features by modeling the video clip as a set, and employ the distance between sets to optimize the hard frame features. In the remainder of this paper, we will present our set triplet loss and empirically verify the superior performance of the proposed method.

3. Method

3.1. Set Theory Revisited

A function $f(\cdot)$, which maps its domain $\mathcal{X}$ to its range $\mathcal{Y}$, is considered as a function of sets if it is permutation invariant to the order of elements in the input. In other words, given a set (*i.e.*, $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s\}$) as input, the function f holds that $f(\boldsymbol{X}) = f(\boldsymbol{P}\boldsymbol{X})$ for any permutation matrix $\boldsymbol{P}$. In this case, the domain of $f(\cdot)$ is the power set of $\boldsymbol{X}$, *i.e.*, $\mathcal{X} = \wp(\boldsymbol{X})$.

Let $(\boldsymbol{X}, d)$ be a metric space. The distance between two nonempty sets $\boldsymbol{A}$ and $\boldsymbol{B}$ in $\wp(\boldsymbol{X})$ (*i.e.* $D : \wp(\boldsymbol{X}) \setminus \emptyset \times \wp(\boldsymbol{X}) \setminus \emptyset \to \mathbb{R}$) measures the similarity of two sets. The ordinary distance between sets (see Fig. 2(b)) is defined as:

$$D^o(\boldsymbol{A}, \boldsymbol{B}) = \inf_{\boldsymbol{a}\in\boldsymbol{A}, \boldsymbol{b}\in\boldsymbol{B}} d(\boldsymbol{a}, \boldsymbol{b}), \tag{2}$$

where $\inf$ denotes the infimum function. The ordinary distance metric could be interpreted as the minimum pair-wise distance between two sets.

Another well-known set distance metric is the Hausdorff distance, which is defined as:

$$\begin{aligned} D^h(\boldsymbol{A}, \boldsymbol{B}) &= \max\Big\{ \sup_{\boldsymbol{a}\in\boldsymbol{A}} d(\boldsymbol{a}, \boldsymbol{B}), \sup_{\boldsymbol{b}\in\boldsymbol{B}} d(\boldsymbol{b}, \boldsymbol{A})\Big\} \\ &= \max\Big\{ \sup_{\boldsymbol{a}\in\boldsymbol{A}} \inf_{\boldsymbol{b}\in\boldsymbol{B}} d(\boldsymbol{a}, \boldsymbol{b}), \sup_{\boldsymbol{b}\in\boldsymbol{B}} \inf_{\boldsymbol{a}\in\boldsymbol{A}} d(\boldsymbol{a}, \boldsymbol{b})\Big\}, \end{aligned} \tag{3}$$

where $\sup$ represents the supremum function. As shown in Fig. 2(c), the geometrical interpretation of the Hausdorff distance can be understood as the greatest of all the distances from an element in one set to the closest element in the other set.

3.2. Triplet Loss

When training a deep video feature extractor, we first sample a mini-batch, which contains P different classes and K video clips for each class, with each video clip having T frames. The network first extracts the frame features, denoted by $\boldsymbol{A}_i = \{\boldsymbol{a}_{i1}, \ldots, \boldsymbol{a}_{iT}\}, i = 1, \ldots, PK$. Then the network aggregates the frame features to a clip feature as $\hat{\boldsymbol{a}}_i = \mathrm{Agg}(\boldsymbol{A}_i)$. Given an anchor clip representation $\hat{\boldsymbol{a}}_i^a$, one possible triplet is formed as $\{\hat{\boldsymbol{a}}_i^a, \hat{\boldsymbol{a}}_i^p, \hat{\boldsymbol{a}}_i^n\}$, where the

positive pair (*i.e.*, $\{\hat{\boldsymbol{a}}_i^a, \hat{\boldsymbol{a}}_i^p\}$) shares the same label, while the negative pair (*i.e.*, $\{\hat{\boldsymbol{a}}_i^a, \hat{\boldsymbol{a}}_i^n\}$) does not. The triplet loss aims to penalize the triplet in which the distance between the positive pair is not sufficiently smaller than that between the negative pair. The triplet loss with hard triplet mining is given by

$$\mathcal{L}_{\text{ctri}}^{\text{hm}} = \frac{1}{PK}\sum_{i=1}^{PK} \max(0, d_i(\hat{\boldsymbol{a}}_i^a, \hat{\boldsymbol{a}}_i^p) - d_i(\hat{\boldsymbol{a}}_i^a, \hat{\boldsymbol{a}}_i^n) + \eta), \tag{4}$$

where η is a task-specific margin. Existing video re-ID machines [6, 7] only optimize the clip representation (see Fig. 2(a)) and it has never been considered to optimize the frame features within each video clip.

3.3. Set-aware Triplet Loss

The nature of the triplet loss is to penalize the positive pairs with a large distance and negative pairs with a small distance. It works well in image re-ID where the triplets are constructed from the image features. However, in video re-ID, the distance measure is hampered by the aggregation operation, as shown in Fig. 1(a). To overcome this issue, we directly enforce the constraint of the triplet loss on the frame features. We first model the frame features within a video clip as a set and employ set theory to calculate the distance between sets. Eqn. (2) and Eqn. (3) formulate the commonly used set distance metrics. However, the geometry interpretation of Eqn. (2) and Eqn. (3) (see Fig. 2(b) and Fig. 2(c)) indicates that those two distance metrics cannot distinguish the distances from the hard positive frames ($\leftrightarrow$ in Fig. 1(a)) and hard negative frames ($\leftrightarrow$ in Fig. 1(a)) simultaneously. Thus, we further propose a hybrid distance metric tailored to the nature of the triplet loss.

Given a triplet, *i.e.*, $\{\boldsymbol{A}^a, \boldsymbol{A}^p, \boldsymbol{A}^n\}$, the hybrid distance metric is defined using the anchor-positive distance and anchor-negative distance individually, as follows:

$$D^{hd+}(\boldsymbol{A}^a, \boldsymbol{A}^p) = \sup_{\boldsymbol{a}^a \in \boldsymbol{A}^a, \boldsymbol{a}^p \in \boldsymbol{A}^p} d(\boldsymbol{a}^a, \boldsymbol{a}^p), \tag{5}$$

and

$$D^{hd-}(\boldsymbol{A}^a, \boldsymbol{A}^n) = \inf_{\boldsymbol{a}^a \in \boldsymbol{A}^a, \boldsymbol{a}^n \in \boldsymbol{A}^n} d(\boldsymbol{a}^a, \boldsymbol{a}^n), \tag{6}$$

where D^{hd+} and D^{hd-} denote the positive pair distance and negative pair distance, respectively. Fig. 2(d) shows the geometrical interpretation of the hybrid distance metric. This formulation allows the loss to penalize the hard frames in each set with the set-aware triplet loss:

$$\mathcal{L}_{\text{stri}}^{\text{hm}} = \frac{1}{PK}\sum_{i=1}^{PK} \max(0, D_i^{hd+} - D_i^{hd-} + \eta). \tag{7}$$

3.4. Hard Positive Set Construction

The network is also supervised by a cross-entropy loss to minimize the within-class variance. Once the network aggregates the frame features to a clip feature as $\hat{\boldsymbol{a}}_i = \text{Agg}(\boldsymbol{A}_i)$. A following fully connected (FC) layer, parameterized by $\boldsymbol{W}$, is used to predict the identity of the video, normalized by the softmax function, as $\boldsymbol{p} = \text{softmax}(\boldsymbol{W}^\top \hat{\boldsymbol{a}}_i)$. A cross-entropy loss is employed to maximize the log likelihood of $\hat{\boldsymbol{a}}_i$ with respect to its label c as follows:

$$\mathcal{L}_{\text{ce}} = \frac{1}{PK}\sum_{i=1}^{PK} -\log\big(p(y_i = c|\hat{\boldsymbol{a}}_i)\big). \tag{8}$$

In Eqn. (8), it holds that $p(y_i = c|\hat{\boldsymbol{a}}_i) \propto \boldsymbol{w}_c^\top \hat{\boldsymbol{a}}_i$. The optimization will maximize $p(y_i = c|\hat{\boldsymbol{a}}_i)$, thereby maximizing the similarity between $\boldsymbol{w}_c$ and $\hat{\boldsymbol{a}}_i$. Thus $\boldsymbol{w}_c$ can be understood as a prototype feature for the class c. Given K sets containing the same class c in one mini-batch, we can further approximate the probability of each frame feature belonging to its label as: $p(y_j = c|\boldsymbol{a}_j), j = 1, \ldots, KT$. For each class, we continue to mine T frame features $\hat{\boldsymbol{A}} = \{\boldsymbol{a}_r : r \in \boldsymbol{i}'\}$, where $\boldsymbol{i}'$ satisfies

$$\boldsymbol{i}' = \{r : \underset{r=1,\cdots,KT}{\arg\min}\ p_r;\ \text{s.t.}|\boldsymbol{i}'| = T\}, \tag{9}$$

and this set is summarized to a set representation (*i.e.*, $\hat{\boldsymbol{a}} = \text{Agg}(\hat{\boldsymbol{A}})$), acting as a hard positive with respect to the original set features $\{\hat{\boldsymbol{a}}_1, \ldots \hat{\boldsymbol{a}}_K\}$ in the batch, where $\hat{\boldsymbol{a}}_i = \text{Agg}(\boldsymbol{A}_i)$. Finally, we could form hard positive pairs as $\{\hat{\boldsymbol{a}}_i, \hat{\boldsymbol{a}}\}, i = 1, \ldots, K$. The hard positive pairs are also minimized by the triplet loss. Besides the hard positive set, we mine a hard negative clip representation to form a valid triple loss, denoted by $\mathcal{L}_{\text{ctri}}^{\text{hpsc}}$. Algorithm 1 summarizes the process of constructing hard positives.

3.5. Network and Optimization

Fig. 3 shows the architecture of the deep network. The network receives a batch of video clips as input and produces frame representations. The original frame features are used to model the set and supervised by the set-aware triplet loss. We further use our proposed hard positive set construction to form hard positive pairs. Then average pooling is used to summarize the clip features. A vanilla triplet loss with hard mining and a triplet loss with hard positive set construction are utilized to supervise the clip features. An additional classifier is further used to train the network. The network is trained to update the parameters by jointly minimizing the multiple triplet losses and cross-entropy loss. The total loss function is formally formulated as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{ce}} + \lambda_2 \mathcal{L}_{\text{ctri}}^{\text{hm}} + \lambda_3 \mathcal{L}_{\text{ctri}}^{\text{hpsc}} + \lambda_4 \mathcal{L}_{\text{stri}}^{\text{hm}}, \tag{10}$$

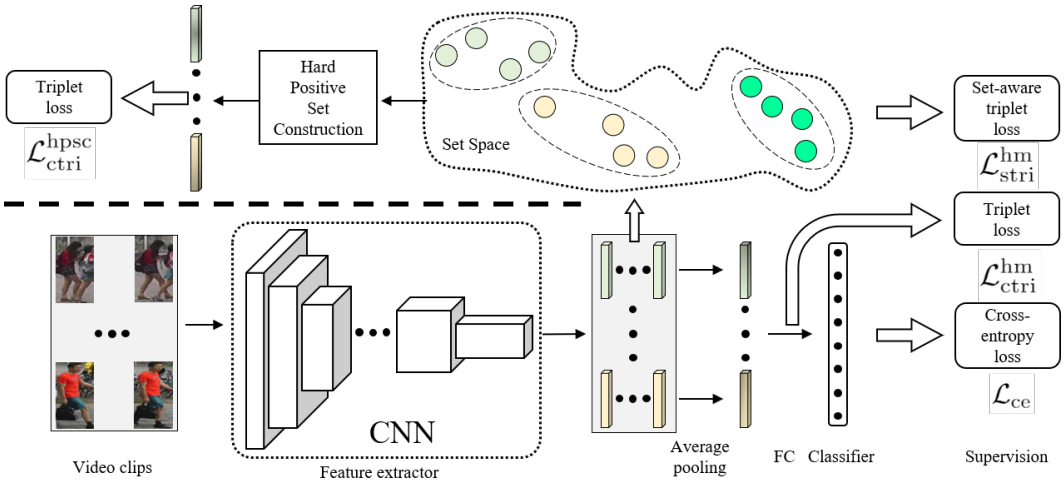


Figure 3. The architecture of the proposed approach. The network receives frame images as input and produces the frame features.

Algorithm 1 Hard Positive Set Construction

Input: K: Number of sets; T: Number of frame features in each set with same class; $\boldsymbol{A}_i = \{\boldsymbol{a}_{i1}, \ldots, \boldsymbol{a}_{iT}\}$: A set of frame features; $\hat{\boldsymbol{a}}_i$: Set feature; $\boldsymbol{W} = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n\}$: Class prototypes; c: Class of sets

Output: $\{\hat{\boldsymbol{a}}_i, \hat{\boldsymbol{a}}\}, i = 1, \ldots, K$: Hard positive pairs

1: Merging all sets with the same class: $\boldsymbol{A} = \{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_T\} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{TK}\}$

2: Calculate the probability of predicting class c for each frame:

$$p(y_j = c|\boldsymbol{a}_j) = \frac{\exp(\boldsymbol{w}_c^\top \boldsymbol{a}_j)}{\sum_{m=1}^{n} \exp(\boldsymbol{w}_m^\top \boldsymbol{a}_j)}, \quad j = 1 \ldots TK$$

3: Pick T frame features with the lowest probability, satisfying

$$\boldsymbol{i}' = \{r : \underset{r=1,\cdots,KT}{\arg\min}\ p_r;\ \text{s.t.}|\boldsymbol{i}'| = T\}$$

4: Construct a hard positive set: $\hat{\boldsymbol{A}} = \{\boldsymbol{a}_r : r \in \boldsymbol{i}'\}$

5: Summarize to hard positive set feature: $\hat{\boldsymbol{a}} = \text{Agg}(\hat{\boldsymbol{A}})$

6: Form hard positive pairs: $\{\hat{\boldsymbol{a}}_i, \hat{\boldsymbol{a}}\}, i = 1, \ldots, K$

where $\mathcal{L}_{\text{ce}}$, $\mathcal{L}_{\text{ctri}}^{\text{hm}}$, $\mathcal{L}_{\text{ctri}}^{\text{hpsc}}$ and $\mathcal{L}_{\text{stri}}^{\text{hm}}$ denote cross entropy loss, clip-feature triplet loss with hard mining, clip-feature triplet loss with hard positive set construction, and set-aware triplet loss with hard mining. The loss terms are weighted by the factors $[\lambda_1, \lambda_2, \lambda_3, \lambda_4]$.

4. Experiments

4.1. Datasets and Evaluation Protocol

We evaluate our method on four popular video person re-identification benchmarks, including PRID-2011 [10], iLIDS-VID [32], MARS [42] and DukeMTMC-VideoReID [35], with samples shown in Fig. 4. The PRID-2011 consists of 200 identities, each with 2 video sequences, amounting to 400 video sequences in total. Both the train and test sets contain 100 person identities. The person trajectories are captured by two disjoint, static cameras. In each frame/image, the person bounding box is manually annotated. Similar to PRID-2011, iLIDS-VID is also a small scale dataset, which contains 600 video sequences of 300 identities, recorded by two cameras in an airport. Each of the train and test sets has 150 person identities. The main challenge of this dataset is the occlusion of the target person. MARS is one of the large-scale video datasets. It has $1,261$ identities and $20,715$ video sequences captured by 6 separate cameras. In this dataset, each video sequence is generated by the GMMCP tracker [2], and the bounding box of each frame is automatically detected by DPM [5]. In this dataset, the train and test sets contain 631 and 630 person identities, respectively. The DukeMTMC-VideoReID is another large video re-ID dataset. This manually labeled dataset contains 702 pedestrians for training, 702 pedestrians for testing. Additionally, this dataset further employs 408 extra pedestrians as distractors. Those 1812 identities have 4832 video sequences. Mean average precision (mAP) and cumulative matching characteristic (CMC) metrics are used to evaluate the proposed method. We report R-1, R-5, R-10 and R-20 values in the CMC metric.

4.2. Implementation Details

4.2.1 Network and Data Organization

We implement all experiments using the PyTorch [20] machine learning package. We use ResNet-50 [8], SE-ResNet-50 [13] and GLTR [15] as baseline networks to evaluate our approach. Noted that the GLTR is self implemented ver-

Table 1. Comparison with state-of-the-art approaches on PRID-2011, iLID-VID and MARS datasets. The 1st best in **bold font**. † indicates the self-implemented network.

Methods	PRID-2011					iLIDS-VID					MARS				
	R-1	R-5	R-10	R-20	mAP	R-1	R-5	R-10	R-20	mAP	R-1	R-5	R-10	R-20	mAP
Chen *et al*. [1]	88.6	99.1	-	-	90.9	79.8	91.8	-	-	82.6	81.2	92.1	-	-	69.4
+ Optcal flow	93.0	99.3	100.0	100.0	94.5	85.4	96.7	98.8	99.5	87.8	86.3	94.7	-	98.2	76.1
QAN [17]	90.3	98.2	99.3	100.0	-	68.0	86.8	-	97.4	-	73.7	84.9	-	91.6	51.7
Li *et al*. [16]	93.2	-	-	-	-	80.2	-	-	-	-	82.3	-	-	-	65.8
PBR [28]	-	-	-	-	-	-	-	-	-	-	83.0	92.8	95.0	96.8	72.2
SCAN [39]	92.0	98.0	100.0	100.0	-	81.3	93.3	96.0	98.0	-	86.6	94.8	-	98.1	76.7
+ Optical flow	95.3	99.0	100.0	100.0	-	88.0	96.7	98.0	100.0	-	87.2	95.2	-	98.1	77.2
STIM-RRU [18]	92.7	98.8	-	99.8	-	84.3	96.8	-	100.0	-	84.4	93.2	-	96.3	72.7
COSAM [26]	-	-	-	-	-	79.6	95.3	-	-	-	84.9	95.5	-	97.9	79.9
STAR+Optical flow [34]	93.4	98.3	100.0	100.0	-	85.9	97.1	98.9	99.7	-	85.4	95.4	96.2	97.3	76.0
STA [6]	-	-	-	-	-	-	-	-	-	-	86.3	95.7	-	98.1	80.8
VRSTC [11]	-	-	-	-	-	83.4	95.5	97.7	99.5	-	88.5	96.5	**97.4**	-	82.3
Zhao *et al*. [41]	93.9	99.5	-	100.0	-	86.3	97.4	-	99.7	-	87.0	95.4	-	98.7	78.2
GLTR [15]	95.5	100.0	-	-	-	86.0	98.0	-	-	-	87.0	95.7	-	98.2	78.4
MG-RAFA [40]	95.9	99.7	-	100.0	-	**88.6**	98.0	-	99.7	-	88.8	97.0	-	98.5	**85.9**
STGCN [37]	-	-	-	-	-	-	-	-	-	-	**89.9**	96.4	-	98.3	83.7
ResNet-50	85.4	98.9	98.9	98.9	91.0	80.0	95.3	98.7	99.3	87.1	82.3	93.9	95.8	97.2	76.2
+ Set Triplet Loss (Ours)	90.2	99.6	100.0	100.0	93.6	85.3	96.0	98.6	99.4	90.4	85.3	95.4	97.1	98.2	81.8
SE-ResNet-50	89.9	98.9	100.0	100.0	94.3	84.0	96.0	98.7	99.3	89.5	85.2	95.3	97.0	97.8	80.0
+ Set Triplet Loss (Ours)	**96.6**	**100.0**	**100.0**	**100.0**	**97.2**	**88.6**	**98.6**	98.7	**100.0**	**92.9**	87.9	**97.2**	97.1	**98.9**	83.2
GLTR†	94.4	99.7	100.0	100.0	95.3	85.2	96.7	97.3	99.7	91.1	86.4	95.4	96.9	97.7	78.8
+ Set Triplet Loss (Ours)	**96.6**	**100.0**	**100.0**	**100.0**	96.9	88.0	98.0	**99.3**	100.0	92.5	87.8	95.5	97.0	97.9	82.2

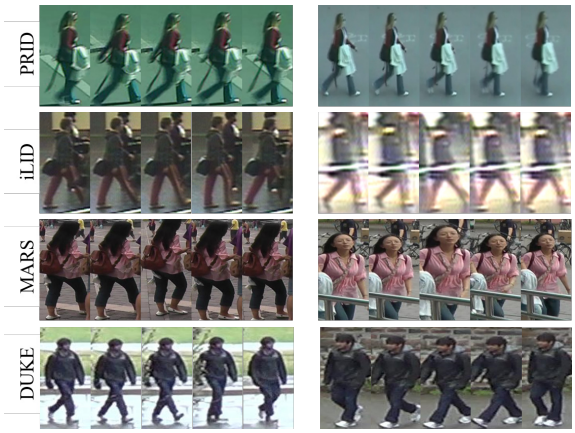


Figure 4. Frames sampled from the pedestrian video sequences across four datasets. Each row shows two sequences of the same person captured by different cameras.

sion. All baselines are pre-trained on ImageNet [24]. The baseline network extracts each frame feature to the dimension of 2048 and we further project them to a lower dimensional space of dimension 1024. Thereafter, a set of frame features are fused to a clip-level video representation and a linear-transformation layer is further utilized to predict the class of the video representation. In each video clip, T is chosen as 4 in all experiments and 4 frames are *randomly* sampled from a video sequence. The frames are first resized to 288×144, and then randomly cropped to 256×128. The data augmentations used in our experiments include randomly flipping in the horizontal direction and random erasing (RE) [44] during training. In the test phase, no data augmentation and re-ranking are used.

4.2.2 Optimization Details

We train the network using the Adam [14] optimizer with default momentum (*i.e.*, $[\beta_1, \beta_2] = [0.9, 0.999]$). The learning rate is initialized to 3e-4 for PRID-2011 and iLIDS-VID datasets, and 4e-4 for MARS and DukeMTMC-VideoReID datasets. During training, the learning rate is decayed by a fixed factor of 1e-1 at the 200^{th} and 400^{th} epoch for the PRID-2011 and iLIDS-VID , and the 100^{th}, 200^{th} and 500^{th} epoch for the MARS and DukeMTMC-VideoReID, respectively. The batch size is set to 16 for the PRID-2011 and iLIDS-VID datasets and 32 for the MARS and DukeMTMC-VideoReID datasets, respectively. In a mini-batch, both P and K are set to 4 for the PRID-2011 and iLIDS-VID, whereas $P = 8$, $K = 4$ for the MARS and DukeMTMC-VideoReID. The margin in Eqn. (4) and Eqn. (7), *i.e.*, η, is set to 0.3 for all datasets. $[\lambda_1, \lambda_2, \lambda_3, \lambda_4] = [1, 0.5, 0.5, 0.5]$. In § 4.4, we will verify each loss component in the total loss function. We report the results of the network at its 800^{th} epoch without any post processing tricks to boost the accuracy, *i.e.*, re-ranking.

4.3. Results

We first compare our method to existing state-of-the art algorithms, as shown in Table 1 and Table 2.

Evaluation on PRID-2011. PRID-2011 is an old video re-ID dataset; thus only a few methods report the mAP value.

To show the superiority of our method, we report both metrics for comparison in Table 1. Our method outperforms MG-RAFA [40] by 0.7% on the R-1 value. Our approach also outperforms the state-of-the-art mAP value in [1] by 2.7%.

Evaluation on iLIDS-VID. Same as for the PRID-2011 dataset, we report the CMC accuracy and mAP value in Table 1. On the iLIDS-VID dataset, our method also achieves state-of-the-art performance. In particular, our network has the same R-1 value with MG-RAFA [40] and outperforms the state-of-the art mAP values by 5.1% in [1].

Evaluation on MARS. Compared with MG-RAFA [40], the state-of-the-art algorithm on the MARS dataset (see Table 1), our method improves the R-5 and R-20 by 0.2% and 0.4% and achieves competitive performance on the R-1 and mAP value.

Evaluation on DukeMTMC-VideoReID. We further evaluate our method on the DukeMTMC-VideoReID dataset. Table 2 compares the performance between our network and existing state-of-the-art algorithms and demonstrates that our method outperforms the STGCN [37] by 0.2% in mAP. Our methods also outperform STA [6] by $0.2\%/0.8\%$ and GLTR by $0.1\%/2.0\%$ in R-1/mAP respectively.

Table 2. Comparison with the state-of-the-art approaches on the DukeMTMC dataset. The 1st best in **bold font**. † indicates the self-implemented network.

Methods	DukeMTMC-VideoReID				
	R-1	R-5	R-10	R-20	mAP
ETAP-Net [35]	83.6	94.6	-	97.6	78.3
STAR+Optical flow [34]	94.0	99.0	99.3	99.7	93.4
VRSTC [11]	95.0	99.1	99.4	-	93.5
STA [6]	96.2	99.3	-	99.7	94.9
GLTR [15]	96.3	99.3	-	99.7	93.7
STGCN [37]	**97.3**	99.3	-	99.7	95.7
ResNet-50	87.5	96.5	97.2	98.3	86.2
+ Set Triplet Loss (Ours)	93.4	98.4	99.8	99.2	91.9
SE-ResNet-50	90.2	97.3	98.0	98.9	89.7
+ Set Triplet Loss (Ours)	96.8	**99.4**	**99.9**	**99.9**	**95.9**
GLTR†	96.0	99.2	99.3	99.5	93.5
+ Set Triplet Loss (Ours)	97.1	**99.4**	99.8	**99.9**	95.4

4.4. Ablation Study

In this section, we will conduct extensive experiments to evaluate the effectiveness of each component in this work.

Effect of set-aware triplet loss. We first evaluate the effectiveness of set-aware triplet loss with different set distance metrics. In this study, we use the SE-ResNet-50 as the backbone network and employ all three distance metrics for the set-aware triplet loss. As shown in Table 3, the set-aware triplet loss indeed helps the network to learn a discriminative person description. Compared with the commonly-used set distance metrics (*i.e.*, ordinary distance, Hausdorff distance), the proposed hybrid distance metric brings the largest performance gain, showing that the optimization to hard frames of anchor-positive pairs and anchor-negative leads the network to create a discriminative video representation.

Table 3. Effect of set-aware triplet loss across the iLIDS-VID and DukeMTMC-VideoReID datasets. SATL: set-aware triplet loss, D^o: ordinary distance, D^h: Hausdorff distance, D^{hd}: Hybrid distance.

Model	iLIDS-VID		DukeMTMC-VideoReID	
	R-1	mAP	R-1	mAP
SE-ResNet-50	84.0	89.5	90.2	89.7
SATL w/ D^o	86.8	90.6	92.8	91.7
SATL w/ D^h	87.6	91.1	94.1	92.9
SATL w/ D^{hd}	**88.3**	**91.9**	**94.9**	**93.7**

Effect of hard positive set construction. We continue to verify the effectiveness of our hard positive set construction method. We still use the SE-ResNet-50 as the backbone network. Table 4 shows that our network benefits from the hard positive set construction method across two datasets. A reasonable explanation for this improvement is that the hard positive sample helps the network minimize the intra-class variance, thereby improving the performance of the network.

Table 4. Effect of hard positive set construction across the iLIDS-VID and DukeMTMC-VideoReID datasets. HPSC: hard positive set construction.

Model	iLIDS-VID		DukeMTMC-VideoReID	
	R-1	mAP	R-1	mAP
SE-ResNet-50	84.0	89.5	90.2	89.7
HPSC	**86.2**	**91.4**	**92.4**	**91.9**

Effect of each loss component. In the study above, we have shown that our network achieves a performance gain from the set-aware triplet loss and the hard positive set construction method. In this study, we will verify each component in the total loss function. SE-ResNet-50 is also used here as the backbone network. The total loss function has four components (*i.e.*, $\mathcal{L}_{\text{ce}}$, $\mathcal{L}_{\text{ctri}}^{\text{hm}}$, $\mathcal{L}_{\text{ctri}}^{\text{hpsc}}$ and $\mathcal{L}_{\text{stri}}^{\text{hm}}$). Table 4 shows the effectiveness of each loss term. In this study, the baseline model is trained by cross-entropy loss (*i.e.*, (i)). The rows in (ii), (iii), and (iv) show that each of the triple losses provides complementary cues to optimize the network. In addition, the terms $\mathcal{L}_{\text{ctri}}^{\text{hpsc}}$ and $\mathcal{L}_{\text{stri}}^{\text{hm}}$ will further improve the performance of the network. In summary, this study reveals that our method helps the network to learn complementary information when encoding the person representation.

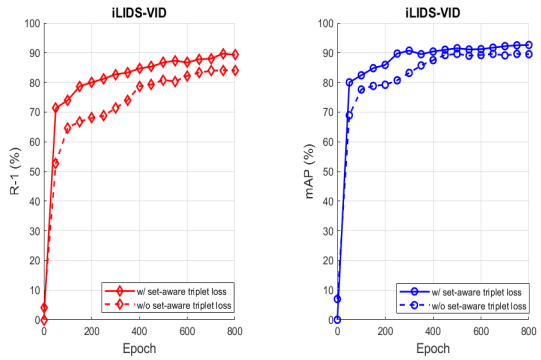
Visualization of hard positive set construction. We further visualize the hard positive set construction by Algorithm 1 on the iLIDS-VID dataset. The original and constructed video clips/sets are framed by black and red lines, respectively. As shown in Fig. 5, we can observe that the frames with occlusions or distractors will be easily selected

Figure 5. Example of hard positive set construction via Algorithm 1 on the iLIDS-VID dataset. The original and constructed video clips/sets are framed by black and red lines, respectively. The constructed clip indicates that the frames with occlusions or distractors will be easily selected as hard samples by our algorithm. Images are sampled from two video sequences from different pedestrians.

Table 5. Effect of each loss component across the iLIDS-VID and DukeMTMC-VideoReID datasets. $[\lambda_1, \lambda_2, \lambda_3, \lambda_4]$ denote the weights assigned to each loss term in Eqn. (10).

	$[\lambda_1, \lambda_2, \lambda_3, \lambda_4]$	iLIDS-VID		DukeMTMC-VideoReID	
		R-1	mAP	R-1	mAP
(i)	$[1, 0, 0, 0]$	74.7	82.5	80.2	79.6
(ii)	$[1, 0.5, 0, 0]$	84.0	89.5	90.2	89.7
(iii)	$[1, 0, 0.5, 0]$	82.0	87.6	87.3	85.2
(iv)	$[1, 0, 0, 0.5]$	84.7	88.9	89.2	88.3
(v)	$[1, 0.5, 0.5, 0]$	85.2	90.4	91.4	90.9
(vi)	$[1, 0.5, 0.5, 0.5]$	**89.3**	**92.9**	**96.8**	**95.9**

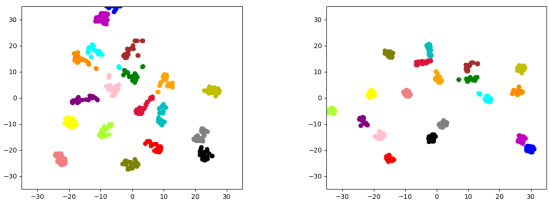
as hard samples by our algorithm. This observation is also in line with our intuition that the hard set is constructed from the hard frames in a batch.



(a) (b)

Figure 6. The training process of the network without set-aware triplet loss and with set-aware triplet loss on the iLIDS-VID dataset. (a): The R-1 value along the training process. (b): The mAP value along the training process.

Training convergence and feature embedding. In this part, we continue to demonstrate the superior performance of set-aware triplets by studying the training convergence and feature embedding of networks. In this study, we also use SE-ResNet-50 as the baseline network. Fig. 6(a) and Fig. 6(b) show the training curves of the network with our set-aware triplet loss and without our set-aware triplet loss w.r.t. the R-1 value and mAP value respectively. Fig. 7(a) and Fig. 7(b) visualize the features extracted by the network, trained without set-aware triplet loss, and with set-aware triplet loss. Both figures clearly show that the set-aware triplet loss indeed helps the network to learn a discriminative embedding space, in which the within-class variance is minimized and the between-class variance is maximized jointly.



(a) (b)

Figure 7. T-SNE visualization [30] of learned features by the network (a) w/o set-aware triplet loss and (b) w/ set-aware triplet loss on the iLIDS-VID dataset. We select 20 people from the query set and visualize the frame features. Points with the same color denote the features of the same person. (Best viewed in color)

5. Conclusion

In this paper, we construct a triplet loss to optimize the frame features of the video person re-ID task, by modeling the video clip as a set. We employ the commonly-used distance metric to measure the distance between sets, *i.e.*, ordinary distance and Hausdorff distance. Considering the hard pairs in the triplets, we further propose a new hybrid distance metric, which is defined for the anchor-positive pair and the anchor-negative pair separately. In addition, we also propose a hard positive set construction algorithm to decrease the within-class variance. Extensive experiments are conducted to verify the superior performance of the proposed method across the standard video person re-ID datasets.

Future work includes employing the set distances to other general metric learning applications or other video-related applications.

References

[1] Dapeng Chen, Hongsheng Li, Tong Xiao, Shuai Yi, and Xiaogang Wang. Video person re-identification with competitive snippet-similarity aggregation and co-attentive snippet embedding. In *CVPR*, 2018.

[2] Afshin Dehghan, Shayan Modiri Assari, and Mubarak Shah. Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking. In *CVPR*, 2015.

[3] Pengfei Fang, Pan Ji, Jieming Zhou, Lars Petersson, and Mehrtash Harandi. Channel recurrent attention networks for video pedestrian retrieval. In *ACCV*, 2020.

[4] Pengfei Fang, Jieming Zhou, Soumava Kumar Roy, Lars Petersson, and Mehrtash Harandi. Bilinear attention networks for person retrieval. In *ICCV*, 2019.

[5] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010.

[6] Yang Fu, Xiaoyang Wang, Yunchao Wei, and Thomas Huang. Sta: Spatial-temporal attention for large-scale video-based person re-identification. In *AAAI*, 2019.

[7] Jiyang Gao and Ram Nevatia. Revisiting temporal modeling for video-based person reid. *arXiv:1805.02104*, 2018.

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

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

[10] Martin Hirzer, Csaba Beleznai, Peter M. Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Image Analysis*, 2011.

[11] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen. Vrstc: Occlusion-free video person re-identification. In *CVPR*, 2019.

[12] Bin Hu, Jiwei Xu, and Xinggang Wang. Learning generalizable deep feature using triplet-batch-center loss for person re-identification. *Science China Information Sciences*, 2020.

[13] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

[15] Jianing Li, Jingdong Wang, Qi Tian, Wen Gao, and Shiliang Zhang. Global-Local Temporal Representation For Video Person Re-Identification. In *ICCV*, 2019.

[16] Shuang Li, Slawomir Bak, Peter Carr, and Xiaogang Wang. Diversity regularized spatiotemporal attention for video-based person re-identification. In *CVPR*, 2018.

[17] Yu Liu, Yan Junjie, and Wanli Ouyang. Quality aware network for set to set recognition. In *CVPR*, 2017.

[18] Yiheng Liu, Zhenghang Yuan, Wengang Zhou, and Houqiang Li. Spatial and temporal mutual promotion for video-based person re-identification. In *AAAI*, 2019.

[19] Niall McLaughlin, Jesus Martinez del Rincon, and Paul Miller. Recurrent convolutional network for video-based person re-identification. In *CVPR*, 2016.

[20] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NeurIPS*, 2017.

[21] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv:1612.00593*, 2016.

[22] Javier Ribera, David Güera, Yuhao Chen, and Edward J. Delp. Locating objects without bounding boxes. In *CVPR*, 2019.

[23] Soumava Kumar Roy, Mehrtash Harandi, Richard Nock, and Richard Hartley. Siamese networks: The tale of two manifolds. In *ICCV*, 2019.

[24] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[25] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[26] Arulkumar Subramaniam, Athira Nambiar, and Anurag Mittal. Co-segmentation inspired attention networks for video-based person re-identification. In *ICCV*, 2019.

[27] Yumin Suh, Bohyung Han, Wonsik Kim, and Kyoung Mu Lee. Stochastic class-based hard example mining for deep metric learning. In *CVPR*, 2019.

[28] Yumin Suh, Jingdong Wang, Siyu Tang, Tao Mei, and Kyoung Mu Lee. Part-aligned bilinear representations for person re-identification. In *ECCV*, 2018.

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

[30] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

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

[32] Taiqing Wang, Shaogang Gong, Xiatian Zhu, and Shengjin Wang. Person re-identification by discriminative selection in video ranking. *TPAMI*, 2016.

[33] Xiaolong Wang, Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2017.

[34] Guile Wu, Xiatian Zhu, and Shaogang Gong. Spatio-temporal associative representation for video person re-identification. In *BMVC*, 2019.

[35] Yu Wu, Yutian Lin, Xuanyi Dong, Yan Yan, Wanli Ouyang, and Yi Yang. Exploit the unknown gradually: One-shot video-based person re-identification by stepwise learning. In *CVPR*, 2018.

[36] Yichao Yan, Bingbing Ni, Zhichao Song, Chao Ma, Yan Yan, and Xiaokang Yang. Person re-identification via recurrent feature aggregation. In *ECCV*, 2016.

[37] Jinrui Yang, Wei-Shi Zheng, Qize Yang, Ying-Cong Chen, and Qi Tian. Spatial-temporal graph convolutional network for video-based person re-identification. In *CVPR*, 2020.

[38] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In *NeurIPS*, 2017.

[39] Ruimao Zhang, Hongbin Sun, Jingyu Li, Yuying Ge, Liang Lin, Ping Luo, and Xiaogang Wang. Scan: Self-and-collaborative attention network for video person re-identification. *arXiv:1807.05688*, 2018.

[40] Zhizheng Zhang, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Multi-granularity reference-aided attentive feature aggregation for video-based person re-identification. In *CVPR*, 2020.

[41] Yiru Zhao, Xu Shen, Zhongming Jin, Hongtao Lu, and Xian-sheng Hua. Attribute-driven feature disentangling and temporal aggregation for video person re-identification. In *CVPR*, 2019.

[42] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *ECCV*, 2016.

[43] Liang Zheng, Yi Yang, and Alexander G. Hauptmann. Person re-identification: Past, present and future. *arXiv:1610.02984*, 2016.

[44] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020.