# Searching Parameterized Retrieval & Verification Loss for Re-identification

Xin Yuan, *Student Member, IEEE*, Xin Xu, *Member, IEEE*, Zheng Wang, *Member, IEEE*, Kai Zhang, *Member, IEEE*, Wei Liu, *Student Member, IEEE*, Ruimin Hu, *Member, IEEE*

***Abstract*—The goal of re-identification (re-ID) is to find an object (*e.g.*, person or vehicle) of interest across cameras. In re-ID, designing suitable and effective loss functions plays an essential and imperative role in learning identifiable features. Regardless of the significant success achieved by using retrieval- or verification-based loss functions due to re-ID can be formulated as a retrieval or verification task, the model performance might be degraded owing to the inconsistency between the loss functions and evaluation metrics. Moreover, current hand-designed loss functions based on evaluation metrics require great expertise and a significant workforce, which is often sub-optimal and laborious. To this end, we propose to automatically design loss functions with specific evaluation metrics for re-ID. Specifically, we propose Parameterized Retrieval & Verification (RV) Loss, which jointly optimizes RV tasks while introducing parameterized functions to replace non-differentiable operations in RV evaluation metrics. Different evaluation metric approximations are thus represented in a single formula by a family of parameterized functions. Then, an automated parameter search algorithm is used to conduct the parameter search. Experimental results indicate that the proposed Parameterized RV Loss can improve the performance of the state-of-the-art re-ID methods, thus demonstrating its effectiveness and superiority over other relevant loss functions on the public person re-ID and vehicle re-ID benchmarks.**

***Index Terms*—Re-identification, Deep metric learning, Loss function search**

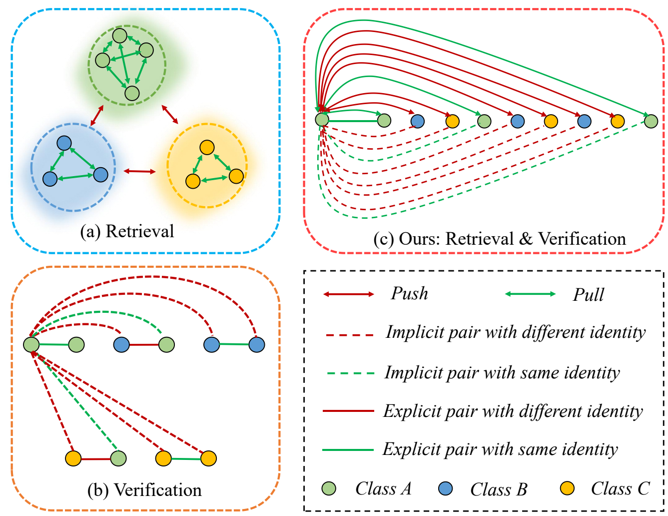


Figure 1. Motivation of the proposed Parameterized RV Loss. Solid circles with same color represent the same class. (a) indicates that re-ID is formulated as a retrieval task. (b) is another manner, *i.e.*, verification task (only negative/positive pairs connected to the first sample are shown). (c) is our solution that combine both the strengths of retrieval and verification tasks (only top-1 samples in ranking list connected to the query are represented). Noteworthy, green and red edges represent whether the sample pair depicts the same identity or not. Dotted edges represent implicit relationships and solid edges represent explicit relationships. Differently, our Parameterized RV Loss combines the strengths of the two tasks. *Best viewed in color.*

## I. Introduction

RE-IDENTIFICATION (re-ID) attempts to find the same identity of object (*e.g.*, person/vehicle) in the surveillance systems [1]–[3]. With the powerful capability of deep neural networks, the past decade has witnessed the significant breakthrough in re-ID. In particular, the design of loss functions with task-specific heuristics plays an inevitable role in model training. Meanwhile, re-ID is usually investigated as a retrieval task [4], [5] or a verification task (also called binary classification task) [6]–[8]. Therefore, the retrieval- or verification-based loss functions are directly adopted to train the re-ID models.

However, there are still some limitations when using retrieval- or verification-based loss functions ***alone***. Listed below are the main reasons: 1) For the retrieval-based loss, it is inevitably challenging to find a true positive with a less similarity [5], [9] (Figure 1(b)); 2) For the verification-based loss, it may result in a local optimum for the learned model [6], [7] (Figure 1(a)). Unlike these works, recent efforts have been proposed to take both retrieval and verification tasks into account in re-ID [10]–[15], *i.e.*, jointly optimizing Retrieval & Verification (RV) tasks simultaneously (Figure 1(c)). For the first limitation, Chen *et al.* [10] potentially rectified this problem by introducing the retrieval-based loss function (*i.e.*, triplet loss). Meanwhile, there is an obvious and easily neglected problem, *i.e.*, the sample pairs are prejudicial, with far more negative sample pairs than positive ones. To tackle the imbalance of sample pairs, a re-weighting strategy derived from the triplet loss was proposed to improve the popular verification loss [13]. For the second limitation, Khatun *et al.* [11], [16] introduced a verification loss function to improve the generalization of the re-ID methods by jointly optimizing both RV tasks. As these two tasks deal with re-ID from different perspectives, impressive results have been achieved by optimizing both the retrieval- and verification-based loss functions in training re-ID models according to their own advantages [10], [11], [13], [16], further demonstrating

Manuscript received August 31, 2022; revised August XX, 2022.
Xin Xu is the corresponding author. This work was supported in part by the NSFC (No. U1803262, 62171325, 62176191).
Xin Yuan, Xin Xu, Kai Zhang, and Wei Liu are with Wuhan University of Science and Technology, Wuhan, Hubei 430065, China (email: xinyuan@wust.edu.cn, xuxin@wust.edu.cn, zhangkai@wust.edu.cn, liuwei@wust.edu.cn).
Zheng Wang and Ruimin Hu are with Wuhan University, Wuhan, Hubei 430072, China (email: wangzwhu@whu.edu.cn, hrm@whu.edu.cn).

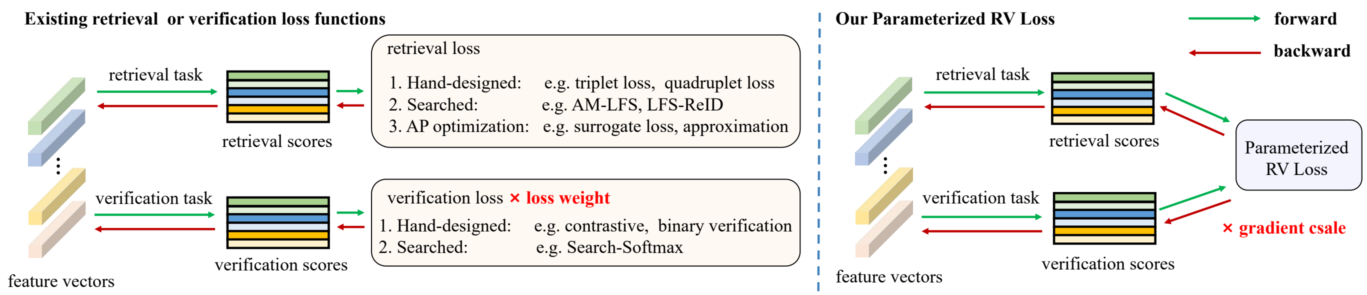


Figure 2. Illustration of comparison between existing loss functions and our Parameterized RV Loss. Different from existing loss functions, we take the retrieval task and the verification task into account. Our Parameterized RV Loss is to constitute a compact search space as well as provide the consistency between training objectives and evaluation metrics from the evaluation metric perspective.

their joint benefit and mutual complementary effects.

In our previous works, we focused on improving the generalization ability of the re-ID model from both global and local features together [17], and investigated two strategies of sampling and re-weighting frames for unsupervised video person re-ID [18]. Although the above-discussed methods and our previous efforts have yielded progressive results, the model performance might be degraded owing to the inconsistency between the loss functions and evaluation metrics [19], [20]. To mitigate this inconsistency problem, a direct and effective solution is to approximate re-ID evaluation metrics (*e.g.*, commonly used Average Precision (AP)) during the model training. Because the AP metric is non-differentiable, many works have manually designed smooth AP approximations according to the mathematical formulation of AP metric [21]–[23]. Besides, we alleviated positive-negative class imbalance during re-ID training process by designing a rank-in-rank loss based on AP [1]. Despite the improvements, these methods of simultaneously optimizing RV tasks are built upon two independent loss functions (see Figure 2). Meanwhile, they depend on hand-designed heuristics that require tremendous effort from the professional experts to exploit the expansive loss function design space, which is often sub-optimal and extremely laborious in practice [24]–[28].

In recent years, with the development of AutoML, its powerful capability of automatically learning has surpassed the hand-designed methods in many fields, such as data augmentation [29]–[31] and networks architecture [32]–[34]. In particular, AutoML-based loss function search methods have emerged and attracted attention recently. Existing AutoML-based loss function search methods can be divided into two categories depending on whether the fixed form of the loss function can be obtained. In the first category, only the final trained model needs to be cared about, instead of directly searching for the fixed form of the loss function [24]–[26]. As a result, these methods are highly coupled to the model and dataset, *i.e.*, the expensive search procedure needs to be repeated when the model or dataset changes. The second category of methods is to search for the loss function, through some primitive operations [27], [28], [35] or the surrogate loss function with parameterized functions [36], [37]. More importantly, the second category of methods can be transferable for models and datasets in similar tasks, without repeating the search procedure. The second category of methods meets our goal, and AutoLoss-GM [27] is the first work to search the generalized margin-based softmax (GMS) loss function with a fixed form for person re-ID. Although AutoLoss-GM [27] has achieved good results, it essentially searches for better loss functions in the GMS loss function space. It is worth noting that the GMS loss function is also hand-designed, which is the approximation for evaluation metrics. Therefore, there exists the above-mentioned inconsistency between the loss functions and evaluation metrics, and this issue remains under-investigated.

In this paper, we propose Parameterized Retrieval & Verification (RV) Loss (shown in Figure 2) for re-ID, which is built on the retrieval and verification evaluation metrics to alleviate the inconsistency between training and evaluation objectives. The core idea is that the search space is constructed according to the evaluation metrics with the help of parameterized functions, which enables the automatic search of the optimal parameters. Specifically, the first step is to reformulate the retrieval and verification evaluation metrics for re-ID as a RV function of retrieval and verification scores. Then we employ the parameterized functions to replace the non-differential operations in above reformulated functions. Finally, we search the parameters via a reinforcement learning-based search process to maximize the RV score on the evaluation set for obtaining the loss function with optimal parameters.

To summarize, the main contributions are listed as follows:

- To the best of our knowledge, we are the first to propose a novel AutoML-based loss (*i.e.*, Parameterized RV Loss) from the evaluation metric perspective for re-ID, which jointly optimizes both retrieval and verification tasks.
- We design a novel verification metric named verification precision (VP) that is constructed with *precision* and *recall*, which is integrated into AP to form the RV function by simply replacing the summation region. Instead of hand-designing smooth surrogate losses or approximated ranking step functions, our method automatically searches for the optimal parameters (*i.e.*, parameter-free) via the differentiable parameterized substitutions.
- Our method achieves competitive performance on public person re-ID and vehicle re-ID datasets, showing its effectiveness and superiority over other relevant methods.

The rest of this paper is organized as follows: §II briefly reviews related works. Then, §III introduces the notations definition used in this paper, and overviews evaluation metrics for re-ID. Next, we introduce our method in §IV. In §V, we show extensive experiments to compare with related state-of-the-art methods and comprehensively study important modules in the ablation study. Finally, we make the corresponding conclusions and summaries of this work in §VI.

## II. Related Work

In this section, we provide a brief overview of related research in five fields: hand-designed loss functions for re-ID, hand-designed loss functions for evaluation metrics, searching loss functions for re-ID, direct optimization for the evaluation metric, and hyper-parameter optimization.

### A. Hand-designed Loss Functions for Re-ID

Designing effective loss functions has attracted extensive interests in re-ID for a long time. Identity loss [38], triplet loss [39], and verification loss [6] are widely used. The identity loss investigates re-ID as a classification task. It is usually used with the label smoothing [40], [41] to avoid overfitting to over-confident annotated labels. To learn discriminative and view-invariant features, the attention mechanism [42], [43] and the adversarial training [44] are developed to learn more robust metrics by combining with identity loss, triplet loss, *etc* [45], which can ultimately enhance the robustness of the learned re-ID model. Moreover, there have been many efforts to improve identity loss and achieve better results, *e.g.*, SphereReID [46], CircleLoss [47], and *etc*. The identity loss can be combined with retrieval- or verification-based loss functions to optimize the re-ID model. It is also worth noting that re-ID is usually viewed as a retrieval or verification task. Therefore, many works have been done to design the appropriate loss functions from the perspective of these two tasks. For the retrieval task, the triplet loss and its variants (*e.g.*, quadruplet loss [48]) propose to keep a pre-defined margin of the distance between the positive pair and the negative pair. For the verification task, to optimize the pairwise relationship, the contrastive loss [6] optimizes the distance between pairwise samples, and the binary verification loss [7] determine whether the input sample pair has the same identity or different identities. Considering the strengths and weaknesses of retrieval and verification tasks, some works [10]–[15] have proposed to build a more comprehensive re-ID solution by jointly optimizing the two tasks simultaneously, which obtains impressive results. It is easy to ignore that the loss functions for these two tasks are trained separately, and these loss functions are not aligned with the re-ID evaluation metrics. Consequently, such joint combined optimization may degrade the model performance. Meanwhile, these loss functions are all hand-designed that require great expertise and human energy, which may be usually sub-optimal.

### B. Hand-designed Loss Functions for Evaluation Metrics

Loss functions play a prominent role in model training. Recent years have witnessed the remarkable success of loss functions for many tasks, including object detection [20], [49], image retrieval [21], [22], [50], [51], and re-ID [1], [2]. For object detection, cross entropy loss [38] and L1 loss [49] are widely used for classification and location, while the corresponding evaluation metrics are $precision/recall$ and IoU respectively. For image retrieval and re-ID, cross entropy loss [38] and triplet loss [39] are commonly used for model training, while the AP metric is the standard evaluation metric for measuring retrieval performance. In summary, these loss functions are all hand-designed, *i.e.*, heavily relying on careful design and human expertise. Specifically, these hand-designed loss functions contain some hyper-parameters or weights to be tuned, which require a lot of time and labor costs for tuning the hyper-parameters before effective model training. Therefore, the extensibility of hand-designed loss functions is restricted. Meanwhile, since these hand-designed loss functions are essentially approximate objectives for specific evaluation metrics in various tasks, there thus exists an inconsistency problem between the loss functions and evaluation metrics. This inconsistency between model training and evaluation leads to suboptimization, resulting in the degradation of model performance. To tackle this issue, many hand-designed loss functions have been proposed for evaluation metrics. The key idea of these hand-designed loss functions is to hand-design differentiable approximations or surrogates of evaluation metrics [1], [20]–[22], [49]–[51] (detailed analysis in §II-C). The most relevant work is our proposed DRSL loss [1] for re-ID, which is to minimize the distance between samples of the same category along with the angle between them based on the AP metric. Although the above-mentioned hand-designed loss functions have achieved some success, they rely heavily on human experience to analyze the characteristics of the evaluation metrics for the specific task before completing the design. In contrast, we propose Parameterized RV Loss to automatically search suitable loss functions based on evaluation metrics, to reduce human labor in re-ID tasks.

### C. Direct Optimization for Evaluation Metrics

There have already been many tasks in which the direct optimization for non-differentiable evaluation metrics were investigated, *e.g.*, information retrieval [52], [53], object detection [20], [49], and image retrieval [21], [22], [50], [51]. There are four groups of studies. *First*, it is to design smooth surrogate loss functions for evaluation metrics. The most commonly used surrogate loss functions contain cross entropy loss [38], triplet loss [39], contrastive loss [6], and *etc*. These methods suffer from inconsistency between training and evaluation objectives, and need specific analysis and tricks to be effective. *Second*, the smooth structured hinge-loss upper bound is designed to optimize the upper bound to the loss functions based on evaluation metrics. Several works are based on the structured SVM model [54], [55] to carry out the investigation. However, the construction with a linear SVM model is restricted, so the performance is limited. Approximate gradient methods [56], [57] are proposed to calculate the loss functions with the loss-augmented inference, which has high computational complexity and requires a task-specific design. Using a blackbox optimization [58] to approximate the calculation of loss functions requires careful design and may be challenging. *Third*, the error-driven learning scheme is adopted to update the non-differentiable components in evaluation metrics. Recently, [20], [49], [59], [60] have achieved significant gains using error-driven learning for target detection, but one weakness of such methods is that they are restricted by specific scenarios. *Fourth*, there are some solutions to explore the approximation of loss functions by designing smooth differentiable approximation of the ranking function. [21], [50] apply soft-binning techniques to approximate the loss functions of evaluation metrics. Another methods try to approximate the non-differentiable ranking function in evaluation metrics with LSTM networks [61] or sigmoid functions [1], [22], [23], [51], which may have vanishing gradients and easily fall into local optimum. Although the aforementioned four groups of methods provide consistency between loss functions and evaluation metrics, it is laborious for them to demand a tailored human design of evaluation metrics.

### D. Searching Loss Functions for Re-ID

AutoML for automatic search has long been investigated in many fields (*e.g.*, data augmentation [29]–[31] and networks architecture [32]–[34]) and has achieved incredible results. AutoReID [62] and CDNet [63] propose to automatically search neural network structures (NAS) for re-ID via reinforcement learning. However, the final output of such methods is the network model rather than the loss function. In recent years, AutoML-based loss function search methods have attracted the interest of researchers. There are some attempts to perform loss function search for re-ID [24]–[27]. In these methods, the final loss functions obtained by AM-LFS [24], Search-Softmax [25], and LFS-ReID [26] are all optimal constructions by searching hand-designed cross entropy loss and its variants. However, there are no fixed forms for the loss functions searched by these methods, so it is difficult to directly transfer and apply the gained loss functions to different networks or datasets. Therefore, AutoLoss-GMS [27] utilizes some primitive operations to search the loss functions that have a fixed form for re-ID. AutoLoss-GMS is the first work to automatically search the generalized margin-based softmax (GMS) loss function, and it achieves exciting success. Meanwhile, it could have different effective forms of loss functions (*e.g.*, five loss functions explored in [27]) according to different settings. Although it can obtain a fixed form of the loss functions, this approach requires enough primitive mathematical operations defined by humans and remains inconsistency between the loss functions and evaluation metrics. In this paper, we propose to provide consistency between training and evaluation objectives by combining evaluation metrics while also ensuring that the loss functions of the final searched result have a fixed form.

### E. Hyper-Parameter Optimization

Hyper-parameter optimization (HPO) has been studied to automatically assist search in the machine learning setting for hyper-parameter configurations over the last 20 to 30 years. Random search and grid search [64] are widely used methods for handling hierarchical search spaces. Because the search space has become diverse and complex, various HPO methods have been proposed to perform an effective search. Bayesian Optimization [65]–[68] uses the historical evaluation results to build an evaluation model, which is then used to evaluate and optimize the hyper-parameters. Currently, evolutionary algorithms [69], [70] are widely used to evolve the optimal hyper-parameters through evolutionary models. Notably, reinforcement learning [71] has also been studied to optimize the search space with the specific sampling policy. This paper proposes to search for optimal parameters by employing reinforcement learning, due to its simplicity and efficiency. Alternatively, other effective HPO techniques can also be applied to optimize the search space.

## III. Revisiting Evaluation Metrics

In this section, we first give notations definition used in this paper, and provide an overview of re-ID evaluation metrics. Because re-ID can be studied as either a verification or retrieval task, two groups of evaluation metrics are included, *i.e.*, verification-based and retrieval-based.

### A. Notations

Some notations definitions used in this paper are as follows. Given a query $\mathbf{q}_i \in \mathcal{Q}$, the re-ID systems need to rank all elements in a gallery set $\mathcal{G}$. Note that the retrieval set $\Omega = \{\mathbf{x}_i\}_{i=1}^N$ with $N$ elements consists of $M$ queries $\mathcal{Q} = \{\mathbf{q}_i\}_{i=1}^M$ and $\mathcal{G}$ with $N-M$ elements, *i.e.*, $\Omega = \mathcal{Q} \cup \mathcal{G}$. Meanwhile, for a query $\mathbf{q}_i \in \mathcal{Q}$, each element in $\mathcal{G}$ is assigned a relevant label $\mathbf{y}_j \in \{1, 0\}$, *i.e.*, $\mathbf{y}_j = 1$ if $\mathbf{x}_i$ is relevant to $\mathbf{q}_i$, and otherwise $\mathbf{y}_j = 0$. Therefore, the retrieval set $\Omega$ can be divided into $\mathcal{P}_i = \{\mathbf{x}_i \in \Omega | \mathbf{y}_j = 1\}$ and $\mathcal{N}_i = \{\mathbf{x}_i \in \Omega | \mathbf{y}_j = 0\}$ according to whether it is relevant to query $\mathbf{q}_i$. For each element $\mathbf{x}_i$, it can be encoded to a feature vector with $d$ dimensions $\mathbf{v}_i \in \mathbb{R}^d$, *i.e.*, $\mathbf{V} = \{\mathbf{v}_i\}_{i=1}^N$. In this paper, we use the cosine similarity, and the similarity score between each query $\mathbf{q}_i$ and each element $\mathbf{x}_j$ in $\Omega$ is computed by $s(\mathbf{q}_i, \mathbf{x}_j) = \frac{\mathbf{v}_{\mathbf{q}_i}^T \mathbf{v}_j}{||\mathbf{v}_{\mathbf{q}_i}||^2 ||\mathbf{v}_j||^2}$.

### B. Verification-based evaluation metrics

**Precision and Recall:** Re-ID can be studied as a verification task, *i.e.*, determining whether two elements have the same identity. This task usually requires a threshold to obtain four kinds of values: True-Positive ($TP$), False-Positive ($FP$), True-Negative ($TN$) and False-Negative ($FN$). Similar to the classification task, two evaluation metrics are used for the validation task: $precision$ [72] and $recall$ [73]:

$$precision = \frac{TP}{TP+FP}, \quad recall = \frac{TP}{TP+FN} \tag{1}$$

### C. Retrieval-based evaluation metrics

**CMC and mAP:** With the proposal of large datasets containing multiple cameras, re-ID is also investigated as a retrieval task. Cumulative Match Characteristic (CMC) [74] curve and mean Average Precision (mAP) [75] are the two most widely used evaluation metrics for retrieval task. For each query $\mathbf{q}_i \in \mathcal{Q}$, any element $\mathbf{x}_j \in \mathcal{G}$ is ranked according to the similarity $s(\mathbf{q}_i, \mathbf{x}_j)$. CMC is calculated if the query identity

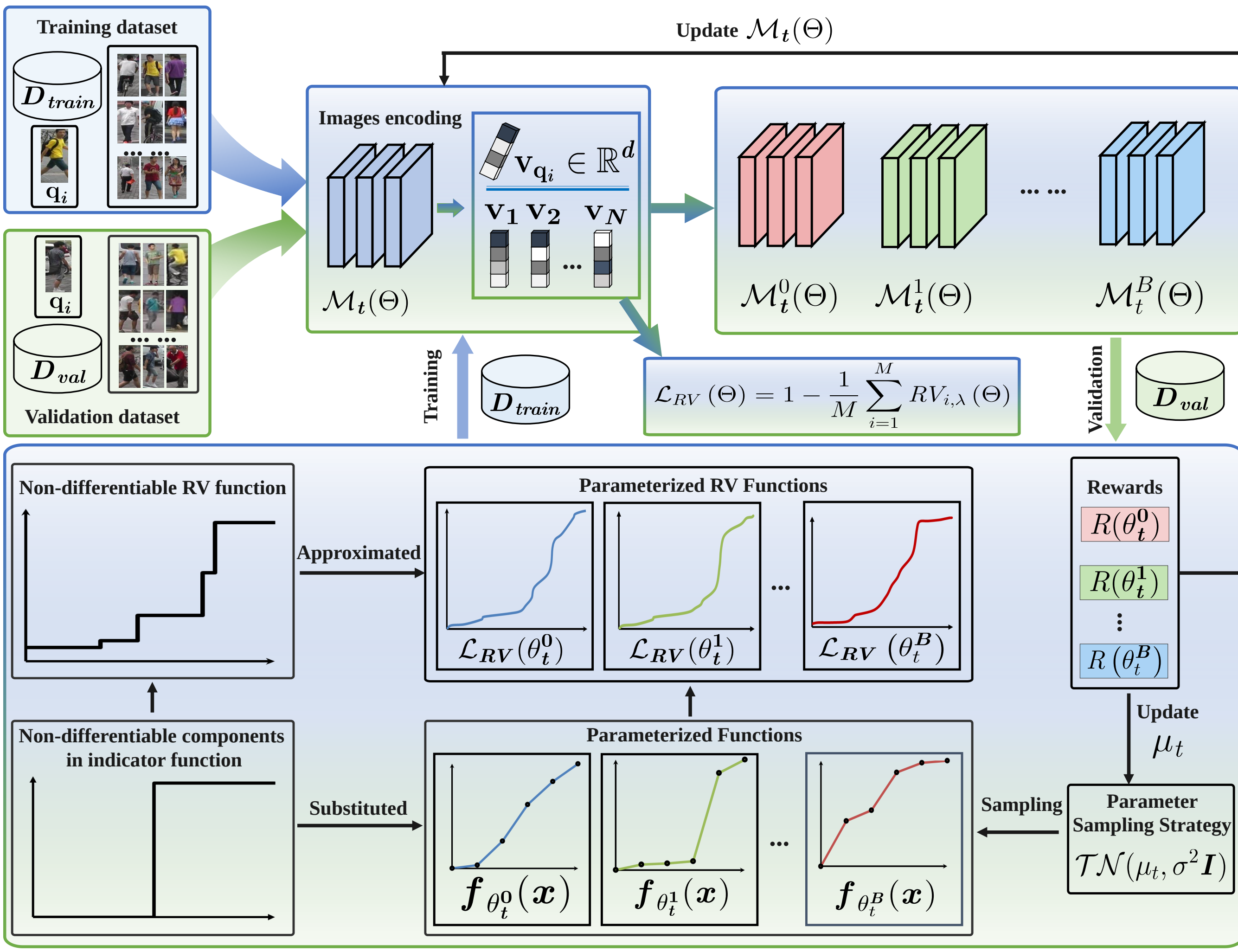


Figure 3. Framework of searching Parameterized RV Loss for re-ID. In each epoch, the hyper-parameter sets $\{\Theta_t^i\}_{i=1}^B$ are sampled through $\Theta \sim \mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$, and using these sampled hyper-parameters to train the network models $\mathcal{M}_t(\Theta)$. After finishing the training, $B$ rewards can be evaluated on the validation dataset, then the parameter sampling strategy and the network models $\mathcal{M}_t(\Theta)$ are updated with these rewards results. *Best viewed in color.*

appears in the top-$k$ ranked gallery items. The right match's rank position is indicated as $R_i^j$. $CMC@k$ can be defined as:

$$CMC@k = \frac{\sum_{i=1}^{M} \mathbb{1}(R_i^j \leq k)}{M}, \tag{2}$$

where $\mathbb{1}$ is the indicator function which equals to 1 only if $R_i^j \leq k$, otherwise 0. The area under the Precision-Recall curve is the average precision (AP) [75]. Thus AP is calculated as

$$AP_i = \sum_k precision_i^k (recall_i^k - recall_i^{k-1}) \tag{3}$$

where $precision_i^k$ and $recall_i^k$ represent $precision$ and $recall$ respectively with respect to query $\mathbf{q}_i$ in the top-$k$ ranked gallery items. Then, mAP evaluates the overall performance:

$$mAP = \frac{1}{M}\sum_{i=1}^{M} AP_i \tag{4}$$

## IV. Methodology

In this section, we present the key components which are included in the parameterized RV loss. The proposed method jointly learns the retrieval and verification tasks in an integrated manner. The framework of our proposed method is illustrated in Figure 3. Firstly, the training set $D_{train}$ and the validation set $D_{val}$ are randomly divided for training and validation respectively. Then in each epoch, the network models $\mathcal{M}_t(\Theta)$ (*i.e.*, $B$ network models initialized from $\{\mathcal{M}(\Theta)_t^i\}_{i=1}^B$ are trained with the hyper-parameters set $\{\Theta_t^i\}_{i=1}^B$ sampled through $\Theta \sim \mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$ on $D_{train}$. Next, after finishing the above epoch training, $B$ rewards are evaluated by $B$ trained network models on $D_{val}$. In this way, we can use these rewards to update the parameters of the parameter sampling strategy $\Theta \sim \mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$. After the above steps are completed, it means that the parameterized search process is finished once. Below, we first introduce the evaluation metrics for the design of loss function. Then we introduce the proposed Parameterized RV Loss, including the design of RV function and the differentiable approximation of RV function. Finally, we describe the parameterized function and the parameter optimization used in our method.

### A. Average Precision & Verification Precision

**Average Precision (AP).** AP is a common evaluation metric for measuring the retrieval performance of the re-ID model. The AP metric is defined as follows.

$$AP_i = \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} precision_i^k \tag{5}$$

where $precision_i^k$ is the precision for the $k$-th relevant element $\mathbf{x}_k$, whose the similarity score is larger than that of $i$-th element. Following [22], [51], Equ. 5 can be further calculated as follows.

$$\begin{aligned} AP_i &= \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} precision_i^k = \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} \frac{Rank(k, \mathcal{P}_i)}{Rank(k, \Omega)} \\ &= \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} \frac{Rank(k, \mathcal{P}_i)}{Rank(k, \mathcal{P}_i) + Rank(k, \mathcal{N}_i)} \\ &= \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} \frac{1 + \sum_{j \in \mathcal{P}_i, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}}}{1 + \sum_{j \in \Omega, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}}} \end{aligned} \tag{6}$$

where $Rank(k, \mathcal{P}_i)$ and $Rank(k, \mathcal{N}_i)$ represent the ranking position of the image $\mathbf{x}_k$ in $\mathcal{P}_i$ and $\mathcal{N}_i$, and $Rank(k, \Omega) = Rank(k, \mathcal{P}_i) + Rank(k, \mathcal{N}_i)$. $\mathbb{1}$ is the indicator function which equals to 1 only if $s_j - s_k > 0$, otherwise 0.

**Verification Precision (VP).** Re-ID can be also investigated as a verification task, *i.e.*, discriminating whether or not two input images have the same identity. The verification task usually needs a special threshold $\Lambda$, *i.e.*, $\Lambda = \{\lambda | s(\mathbf{q}_i, \mathbf{x}_j) \geq \lambda\}$. $precision$ and $recall$ are the most widely used evaluation metrics for evaluating the verification performance of the re-ID model. In this paper, to keep the trade-off between $precision$ and $recall$, we proposed to define a novel verification evaluation metric, *i.e.*, verification precision (VP). The VP under threshold $\lambda$ can be defined as.

$$VP_{i,\lambda} = \frac{recall_{i,\lambda} * precision_{i,\lambda}}{recall_{i,\lambda} + precision_{i,\lambda} - recall_{i,\lambda} * precision_{i,\lambda}} \tag{7}$$

where $precision_{i,\lambda}$ and $recall_{i,\lambda}$ denote the $precision$ and $recall$ under the threshold $\lambda$ with respect to the query $\mathbf{q}_i$. With a further extension, the $VP$ can also be formulated as follows.

$$VP_{i,\lambda} = \frac{Rank(\mathcal{P}_i; \lambda)}{Rank(\mathcal{N}_i; \lambda) + |\mathcal{P}_i|} \tag{8}$$

where $Rank(\mathcal{P}_i; \lambda)$ and $Rank(\mathcal{N}_i; \lambda)$ are the ranking position of the image $\mathbf{x}_j$ in $\mathcal{P}_i$ and $\mathcal{N}_i$ under the threshold $\lambda$, respectively.

### B. Retrieval & Verification Function

One of the major differences from previous works is that we propose to take the retrieval and verification tasks into account, and we introduce the retrieval-based and verification-based metrics as the training objective. In particular, we propose a unified parameterized formulation (*i.e.*, our proposed Parameterized RV Loss) to denote the approximation of the various potential re-ID metrics. Specifically, the steps of our proposed Parameterized RV Loss are as follows. Firstly, the AP metric is transformed to a new form, and our designed VP metric is integrated into the new form of AP to ultimately produce the final Retrieval & Verification (RV) function. Secondly, the non-differentiable indicator functions in the RV function are replaced by parameterized functions, which makes the optimization objective to be a differentiable approximation of the RV function, enabling the network to be back-propagated and trained to perform the optimization of the loss function.

**Retrieval & Verification Function.** Since Equ. 6 only assesses the retrieval scores of re-ID, it requires to be further reformulated as the function that takes the verification score into account. Based on this key insight, It drives us to reformulate the AP by obviously incorporating the verification evaluation metric. Before reformulating the AP metric with our designed VP metric, we first transform Equ. 6 to a new form as follows.

$$AP_i = \frac{1}{|\mathcal{P}_i|} \sum_{k \in \mathcal{P}_i} 1 - \frac{\sum_{j \in \mathcal{N}_i, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}}}{1 + \sum_{j \in \Omega, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}}} \tag{9}$$

Next, we deliberate on how to integrate the VP metric into the AP metric. In Equ. 9, it can be noted that the part associated with the verification task is the summation region (*i.e.*, $\mathcal{P}_i$ and $\mathcal{N}_i$), thus it inspires us to further replace the summation region in Equ. 9 using Equ. 8. After this substitution operation, we can obtain the final reformulated function including retrieval and verification scores for re-ID, *i.e.*, Retrieval & Verification function $RV_{i,\lambda}$.

$$\begin{aligned} RV_{i,\lambda} &= \frac{1}{|\mathcal{P}_i|} \sum_{k \in \Omega} \mathbb{1}_{\{VP_{i,\lambda}(k) > 0\}} - \\ &\frac{\sum_{j \in \Omega, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}} * \left(1 - \mathbb{1}_{\{VP_{i,\lambda}(j) > 0\}}\right)}{1 + \sum_{j \in \Omega, j \neq k} \mathbb{1}_{\{s_j - s_k > 0\}}} * \mathbb{1}_{\{VP_{i,\lambda}(k) > 0\}} \end{aligned} \tag{10}$$

where $\mathbb{1}_{\{VP_{i,\lambda}(\cdot) > 0\}}$ is the indicator function with the verification score as judgment condition, which represents whether the top ranked samples have the same identity with respect to the query $\mathbf{q}_i$ under the threshold $\lambda$. And $VP_{i,\lambda}(k)$ is computed as.

$$VP_{i,\lambda}(k) = \begin{cases} \frac{Rank(k, \mathcal{P}_i; \lambda)}{Rank(k, \mathcal{N}_i; \lambda) + |\mathcal{P}_i|} & k \in \mathcal{P}_i \\ 0 & k \in \mathcal{N}_i \end{cases} \tag{11}$$

where $Rank(k, \mathcal{P}_i; \lambda)$ and $Rank(k, \mathcal{N}_i; \lambda)$ represent the ranking position of the image $\mathbf{x}_k$ in $\mathcal{P}_i$ and $\mathcal{N}_i$ under the threshold $\lambda$. Finally, we multiply each summation term by the approximation of the indicator function to replace the summation region in Equ. 10.

### C. Differentiable Parameterized RV Approximations

To provide a suitable approximation of the RV function in Equ. 10, we need to introduce a differentiable approximated substitution to supersede the non-differentiable indicator function $\mathbb{1}$. The essence of such an operation is ultimately to ensure the consistency between model training and evaluation. To address this problem, there have been many peer efforts to attempt to hand-design some efficient solutions, such as well-designed error-driven update scheme [49], [76], smooth approximation of the rank function [22], [58], [61] and so on. However, these hand-designed methods may unavoidably lead to local optimization of the training process due to the heavy dependence on tedious priors and experiences.

Different from existing hand-designed methods, parameterized functions are proposed to take place the non-differentiable operations in the indicator functions, which is to guide the training process in the optimal direction. Thus, our overall Parameterized RV Loss $\mathcal{L}_{RV}$ is averaged over all queries:

$$\mathcal{L}_{RV}(\Theta) = 1 - \frac{1}{M}\sum_{i=1}^{M} RV_{i,\lambda}(\Theta) \tag{12}$$

where $\Theta$ represents the model parameters, and $RV_{i,\lambda}(\Theta)$ denotes that RV function is parameterized by $\Theta$, *i.e.*, the indicator function $\mathbb{1}$ substituted with parameterized functions. Obviously, our goal is to minimize the Parameterized RV Loss $\mathcal{L}_{RV}$ for each query $\mathbf{q}_i$ in the gallery set $\Omega$. And $RV_{i,\lambda}(\Theta)$ can be further computed as follows.

$$\begin{aligned} RV_{i,\lambda}(\Theta) = &\frac{1}{|\mathcal{P}_i|}\sum_{k\in\Omega} f_{\theta_1}(VP_{i,\lambda}(k)) - \\ &\frac{\sum_{j\in\Omega,j\neq k} f_{\theta_2}(s_j-s_k)*\left(1-f_{\theta_3}(VP_{i,\lambda}(j))\right)}{1+\sum_{j\in\Omega,j\neq k} f_{\theta_4}(s_j-s_k)} * f_{\theta_5}(VP_{i,\lambda}(k)) \end{aligned} \tag{13}$$

where $f_\theta(x)$ parameterized by $\theta$ in Equ. 13 is used to replace the non-differentiable indicator function in Equ. 10. It worth noting that we adopt different parameters $\Theta = \{\theta\}_{i=1}^{5}$ in Equ. 13, and we normalize the input and output ranges of $f_\theta(x)$ to $[0,1]$ in unit norm (see the implementation details in §V-B). The former utilizes different parameters to replace different indicator functions in Equ. 8, which makes the Parameterized RV Loss more flexible and scalable; the latter plays the role of keeping the output value of $f_\theta(x)$ and the indicator function consistent. Experimentally, this separated parameterization has been demonstrated to perform better than that of shared parameters for different indicator functions in §V.

### D. Parameterized Function.

Note that the parameterized function $f_\theta(x)$ can be one of any differentiable functions. The most commonly used parameterized functions are piecewise Bézier curve and piecewise linear functions, respectively. The piecewise Bézier curve is an important parameterized curve in computer graphics, and its shape is determined by control points. In contrast, the piecewise linear function is a simple yet effective parameterized curve verified in [36], and its shape is controlled by the divided points. Furthermore, the number of divided points in the piecewise linear function is twice less than the number of piecewise Bézier curve in the same settings. Importantly, the searched loss functions parameterized with piecewise linear functions can achieve very similar performance with that of using the piecewise Bézier curve for the parameterization in [36].

Therefore, for simplicity, we adopt the piecewise linear function in this paper. In detail, given a positive integer $M^{'}$, we divide the input domain $[0,1]$ into $M^{'}$ different sized intervals, corresponding measure $\Delta_x = \{\delta_{x_1}, \delta_{x_2}, \ldots, \delta_{x_{M'}}\}$ and $\Delta_y = \{\delta_{y_1}, \delta_{y_2}, \ldots, \delta_{y_{M'}}\}$. From left to right, each coordinate of these points can be denoted by $\{(x_{m'}, y_{m'})\}_{0\le m'\le M'}$, where $x_{m'} = \sum_{i=1}^{m'}\delta_{x_i}$ and $y_{m'} = \sum_{j=1}^{m'}\delta_{y_j}$, and the two end points are $(x_0, y_0) = (0,0)$ and $(x_{M'}, y_{M'}) = (1,1)$. To regularize the search space, the parameter constraints for the rest point coordinates are listed as follows.

$$0 \le \frac{\delta_{x_{m'}}}{\sum_{i=m'}^{M'}\delta_{x_i}} < 1 \quad \& \quad 0 \le \frac{\delta_{y_{m'}}}{\sum_{j=m'}^{M'}\delta_{y_j}} < 1 \tag{14}$$

Therefore, the parameterized function $f_\theta(x)$ has $M^{'}$ parts, its $m^{'}$ part is formulated as follows.

$$\begin{aligned} f_\theta^{m^{'}}(x) &= \frac{\delta_{y_{m'+1}}}{\delta_{x_{m'+1}}}\cdot(x - x_{m'}) + y_{m'} \\ x_{m'} \le x < x_{m'+1}, &\quad m^{'} = 0,\ldots,M^{'}-1 \end{aligned} \tag{15}$$

The parameterized function $f_\theta(x)$ is ultimately constrained by the coordinates of these divided points, and the corresponding parameters $\theta = \{(\frac{\delta_{x_{m'}}}{\sum_{i=m'}^{M'}\delta_{x_i}}, \frac{\delta_{y_{m'}}}{\sum_{j=m'}^{M'}\delta_{y_j}})\}_{0\le m'<M'}$ are independent between each other, making the search process easier.

### E. Parameter Optimization

In this subsection, we describe the search process of parameter optimization in our Parameterized RV Loss. Suppose we have a network model $M(\Theta)$ parameterized by the parameter set $\Theta$. In the search process of parameter optimization, the training set is divided into two parts: $D_{train}$ for training and $D_{val}$ for validation respectively. The search process of parameter optimization can be investigated as a standard bi-level optimization problem, and its formulation is as follows.

$$\begin{aligned} &\max_{\Theta} R(\Theta) = RV(\omega^*(\Theta); D_{\text{val}}) \\ &\text{s.t. } \omega^*(\Theta) = \min_{\omega} \mathcal{L}_{RV}(\omega(\Theta); D_{\text{train}}) \end{aligned} \tag{16}$$

where $RV(\omega^*(\Theta); D_{\text{val}})$ calculate the RV metric for the model with weights $\omega^*(\Theta)$ on on the validation dataset $D_{\text{val}}$, and $\mathcal{L}_{RV}$ is our proposed Parameterized RV Loss with the hyper-parameter set $\Theta$. In detail, we trained the model weights $\omega(\Theta)$ by minimizing the Parameterized RV Loss $\mathcal{L}_{RV}(\Theta)$ for each epoch at the inner level. At the outer level, the process will search a optimal loss function hyper-parameters $\Theta$ to make the model weights $\omega^*(\Theta)$ for maximizing the reward $RV(\omega^*(\Theta); D_{\text{val}})$ on the validation dataset $D_{\text{val}}$.

In this paper, following [71], popular hyper-parameter search algorithm is often used to search for the optimal loss function parameters. Specifically, the PPO2 algorithm [77] is applied to optimize the hyper-parameters in training process. And $B$ sets of hyper-parameters $\{\Theta_t^i\}_{i=1}^{B}$ are sampled from a independent truncated Gaussian distribution [78], denoted as $\Theta \sim \mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$ (*e.g.*, $\mu_t$ and $\sigma^2$ are the mean and variance values, respectively), where $B$ indicates the total number of sampled parameter sets, and the truncated range of $\mathcal{TN}$ is $[0,1]$ for satisfying the range constraint of the independent parameters $\theta$.

Suppose we trained the network model $\mathcal{M}(\Theta)$ with $T$ epochs. In the $t$-th epoch, for $i$-th sample of parameter set, the corresponding reward is $R(\theta_t^i)$ at the inner level, and

**Algorithm 1** Parameterized RV Loss

**Input:** Training dataset $D_{train}$, validation dataset $D_{val}$, initialized the network model weights $\omega_0$, initial parameter set distribution $(\mu_0, \sigma^2 I)$, training epochs $T$, and sampling number $B$.
**Output:** Final hyper-parameter set $\Theta^*$

```
for t = 1 to T do
    for b = 1 to B do
        Initialization of the sampled hyper-parameter set Θ_t^b through TN_[0,1](μ_t, σ²I);
        Inner-level model training initialized with the model weights ω_0, ω*(Θ) = min_ω L_RV(ω(Θ); D_train);
        Evaluate the corresponding reward, max_Θ R(Θ) = RV(ω*(Θ); D_eval);
    end for
    Outer-level distribution update, updating μ_{t+1} using Equ. 17;
end for
return Θ* = argmax_Θ R(Θ_t^i)
```

for the next $t$+1-th epoch, we can update the mean value of the truncated Gaussian distribution $\Theta \sim \mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$ by PPO2 algorithm as follows.

$$\mu_{t+1} = \underset{\mu}{\operatorname{argmax}} \frac{1}{B} \sum_{i=1}^{B} \tilde{R}_t^i \left(\mu, \mu_t, \Theta_t^i\right) \tag{17}$$

where the reward$\tilde{R}_t^i \left(\mu, \mu_t, \Theta_t^i\right)$ is computed as follows.

$$\tilde{R}_t^i \left(\mu, \mu_t, \Theta_t^i\right) = \min\left(\frac{p\left(\Theta_t^i; \mu, \sigma^2 I\right)}{p\left(\Theta_t^i; \mu_t, \sigma^2 I\right)} R_t^i, \mathrm{CLIP}\left(\frac{p\left(\Theta_t^i; \mu, \sigma^2 I\right)}{p\left(\Theta_t^i; \mu_t, \sigma^2 I\right)}; 1-\epsilon, 1+\epsilon\right) R_t^i\right) \tag{18}$$

where $p\left(\Theta_t^i; \mu_t, \sigma^2 I\right)$ is the PDF of $\mathcal{TN}_{[0,1]}(\mu_t, \sigma^2 I)$, and the CLIP function described in PPO2 [77] trims the input value between $1-\epsilon$ and $1+\epsilon$ for more stable and efficient search. The training procedure of our proposed Parameterized RV Loss is summarized in Algorithm 1.

## V. Experiments and Comparisons

### A. Experimental Setups

*1) Benchmarks:* To verify the effectiveness of our Parameterized RV Loss, we comprehensively conduct experiments on four large-scale re-ID datasets (see Table I), *i.e.*, 1) three person re-ID datasets: **Market-1501** [75], **MSMT17** [79] and **CUHK03** [80]; 2) two vehicle re-ID datasets: **VeRi-776** [81] and **VehicleID** [82]. Market1501 dataset contains 1,501 identities, 19,732 gallery images and 12,936 training images captured from 6 different cameras by the DPM detector. MSMT17 dataset contains 126,411 person images with 4,101 identities obtained by 15 different camera views. CUHK03 dataset contains 26,264 images of 1367 identities as the training set, and 1928 images of 100 identities as the testing set. We use the novel training/testing protocol proposed in [83], and we compare the dataset in the detected mode of CUHK03 in our experiments. VeRi-776 dataset contains 776 identities, 37,778 images for training and 11,579 images for testing captured in 20 cameras. VehicleID dataset contains 221,736 vehicle images with 26,267 identities obtained in 12 cameras.

Table I
Statistics of training/testing set on person re-ID and vehicle re-ID datasets. The top part is the person re-ID datasets, and the bottom part is the vehicle re-ID datasets.

| Benchmarks | Training | | Testing | |
|---|---|---|---|---|
| | # IDs | # Imges | # Query | # Gallery |
| Market-1501 [75] | 751 | 12,936 | 3,368 | 19,732 |
| CUHK03 [80] | 767 | 14,733 | 2,800 | 10,660 |
| MSMT17 [79] | 1,041 | 30,248 | 11,659 | 82,161 |
| VeRi-776 [81] | 576 | 30,118 | 1,678 | 11,579 |
| VehicleID [82] | 13,164 | 100,182 | 17,638 | 2,400 |

*2) Evaluation Metrics:* We use three kinds of evaluation metrics. Following standard testing protocol, we adopt mean Average Precision (mAP) [75], Cumulative Matching Characteristics (CMC) [74], and *precision* [72] / *recall* [73] (P/R) as our evaluation metrics to evaluate the model performance on the testing set.

### B. Implementation Details

As described in §IV-E, we first randomly split 95% of the images from the original training set as $D_{train}$ and the other images as $D_{val}$. Since our Parameterized RV Loss is investigated as a standard bi-level optimization problem, our implementation settings can be divided into inner level and outer level. In the inner level, we utilize the Adam optimizer to optimize the re-ID model weights $\omega(\Theta)$ according to Algorithm 1 in each epoch training. In the outer level, $B = 4$ samples are sampled during each epoch, the number of search rounds is set to T = 40, and the total number of training epoch is set to 120. we maintain all experimental settings, such as warm-up, learning rate, batch size, *etc.*, consistent with the corresponding original baseline methods on all re-ID benchmarks. Here, we describe the detailed experimental setup in the following three aspects.

**Preparation for Calculating Loss.** The input values of the parameterized functions $f_\theta(x)$ are normalized to range $[0, 1]$. For retrieval score differences $s_j - s_k$, their range is $[-1, 1]$, thus we use min-max re-scaling to map in the range of $[0, 1]$. For the verification score, it is measured with our defined VP in this paper, and its range is $[0, 1]$, so it does not need normalization. The threshold $\lambda$ is set to 0.3. And we set the number of segments $M^{'}$ to 5. In the experiments, the whole retrieval set $\Omega$ consists of all training samples during the current mini-batch, thus the rest positive set $\mathcal{P}_i$ and negative set $\mathcal{N}_i$ are determined by the given query $\mathbf{q}_i$.

**Setting of Parameter Optimization.** The initial mean value $\mu_0$ of the truncated Gaussian distribution is initialized with $f_\theta(x) = x$. The variance $\sigma$ is initialized as 0.2, which always decays linearly to 0 during the search process. In Equ. 18, each module value of $\Theta$ is constrained using the trimming operation. Following PPO2 [77] algorithm, the trimming value $\epsilon$ is set to 0.1. And the warm-up strategy is utilized to bootstrap the model for enhancing performance. Specifically, in the first 10 epochs, linearly increasing the learning rate from $3.5 \times 10^{-5}$ to $3.5 \times 10^{-4}$. The learning rate is decayed to $3.5 \times 10^{-5}$ and $3.5 \times 10^{-6}$ at 40-th and 70-th epochs, respectively.

Table II
PERFORMANCE COMPARISONS WITH STATE-OF-THE-ART METHODS ARE PERFORMED ON THE MARKET-1501, MSMT17 AND CUHK03 DATASETS. THE ABOVE TWO PARTS ARE THE METHODS BASED ON THE RETRIEVAL-BASED LOSS, THE VERIFICATION-BASED LOSS, OR THEIR IMPROVED VARIANT. AND THE BELOW TWO PARTS ARE THE METHODS BASED ON THE NEURAL ARCHITECTURE SEARCH (NAS) AND THE LOSS FUNCTION SEARCH (LFS). R1 REPRESENTS THE CMC AT RANK-1 ACCURACY. THE BOLD FONT INDICATES THE BEST PERFORMANCE. WE REPORT R1 (%) AND MAP (%).

| Methods | Reference | Loss | Market-1501 | | MSMT17 | | CUHK03 | |
|---|---|---|---|---|---|---|---|---|
| | | | R1 | mAP | R1 | mAP | R1 | mAP |
| IV-reID [7] | TOMM'17 | Softmax+Verification | 79.5 | 44.6 | – | – | – | – |
| TriNet [4] | arXiv'17 | Triplet | 84.9 | 69.1 | 56.9 | 26.9 | – | – |
| Support Neighbor [84] | ACMMM'18 | Separation+Squeeze | 88.3 | 73.4 | – | – | – | – |
| SphereReID [46] | JVCIR'19 | Sphere-Softmax | 94.4 | 83.6 | – | – | – | – |
| PCB + RPP [85] | ECCV'18 | Softmax | 93.8 | 81.6 | – | – | 63.7 | 57.5 |
| DATRL-ReID [86] | TCSVT'19 | Labels Interval Extension Loss+Group Loss | 94.4 | 81.5 | 74.2 | 45.3 | 66.4 | 60.6 |
| FGSAM [87] | TIP'20 | Focal Triplet Loss | 91.5 | 85.4 | – | – | – | – |
| ResNet50 + CircleLoss [88] | CVPR'20 | CircleLoss | 94.2 | 84.9 | 76.3 | 50.2 | – | – |
| AutoReID [62] | ICCV'19 | Softmax+Triplet | 94.5 | 85.1 | 78.2 | 52.5 | 73.3 | 69.3 |
| CDNet [63] | CVPR'21 | Softmax+Triplet | 95.1 | 86.0 | 78.9 | 54.7 | – | – |
| SphereReID + AM-LFS [24] | ICCV'19 | AM-LFS | 94.4 | 85.0 | – | – | – | – |
| ResNet50 + Search-Softmax [25] | ICML'20 | Search-Softmax | 94.4 | 85.7 | – | – | 65.8 | 63.1 |
| ResNet50 + AutoLoss-GMS-A [27] | CVPR'22 | AutoLoss-GMS | 94.7 | 87.0 | 79.5 | 55.1 | 70.4 | 68.3 |
| ResNet50 + LFS-ReID [26] | PR'22 | LFS-ReID | 95.2 | 87.4 | 79.9 | 56.4 | 75.4 | 72.3 |
| **ResNet50 + Parameterized RV Loss** | **This work** | **Parameterized RV Loss** | **96.3** | **88.3** | **81.2** | **58.1** | **78.2** | **74.3** |

Table III
PERFORMANCE COMPARISON ON THE VERI-776 DATASET (%). WE REPORT R1 (%) AND MAP (%). THE BEST RESULTS OF PERFORMANCE ARE INDICATED IN BOLD.

| Methods | Reference | Loss | VeRi-776 | |
|---|---|---|---|---|
| | | | R1 | mAP |
| BagofTricks [41] | TMM'20 | Softmax+Triplet | 90.2 | 67.6 |
| VOC-ReID [89] | CVPRW'20 | Circle+Triplet | 95.9 | 78.6 |
| VANet [90] | ICCV'19 | Triplet | 89.8 | 66.3 |
| DMML [38] | ICCV'17 | Softmax | 90.2 | 70.1 |
| PAMTRI(ALL) [91] | CVPR'19 | Softmax+Triplet | 92.8 | 71.8 |
| PRReID [92] | CVPR'19 | Softmax | 93.3 | 72.5 |
| UMTS [93] | AAAI'20 | Softmax+UA-KDL | 95.8 | 75.9 |
| PGAN [94] | TITS'20 | Softmax+Triplet | 96.5 | 79.3 |
| SAVER [95] | ECCV'20 | Softmax+Triplet | 96.4 | 79.6 |
| CFVMNet [96] | ACMMM'20 | Softmax+Triplet | 95.3 | 77.1 |
| SGFD [97] | ICCV'21 | Softmax+Triplet | 96.7 | 81.0 |
| **Our** | **This work** | **Parameterized RV Loss** | **97.3** | **81.2** |

Table IV
PERFORMANCE COMPARISON ON THE VECHICLEID DATASET (%). WE REPORT R1 (%) AND R5 (%). THE BEST RESULTS OF PERFORMANCE ARE INDICATED IN BOLD.

| Methods | VehicleID | | | | | |
|---|---|---|---|---|---|---|
| | Small | | Medium | | Large | |
| | R1 | R5 | R1 | R5 | R1 | R5 |
| Divide [99] | 87.7 | 92.9 | 85.7 | 90.4 | 82.9 | 90.2 |
| MIC [100] | 86.9 | 93.4 | – | – | 82.0 | 91.0 |
| FastAP [21] | 91.9 | 96.8 | 90.6 | 95.9 | 87.5 | 95.1 |
| Cont. w/M [101] | 94.7 | 96.8 | 93.7 | 95.8 | 93.0 | 95.8 |
| Smooth-AP [22] | 94.9 | 97.6 | 93.3 | 96.4 | 91.9 | 96.2 |
| PNP-$D_q$ [23] | 95.5 | 97.8 | 94.2 | 96.9 | 93.2 | 96.6 |
| **Our** | **96.3** | **98.1** | **94.8** | **97.3** | **93.9** | **97.0** |

**Training Details.** After the parameter optimization, *i.e.*, the parameter search of loss function, we need to retrain our model with the search Parameterized RV Loss on $D_{train}$, and evaluate the model performance on $D_{val}$. In all experiments, our implementation is based on Bagoftricks [41] using ResNet50 pretrained on ImageNet [98]. The total batch size is 64. Each mini-batch is formed by sampling $K$ identities and $|P|$ per identity, where $K = 16$ and $|P| = 4$ in the following experiments. Our method is implemented based on the Bagoftricks [41]. All the experiments are conducted on four RTX 2080Ti GPUs.

### C. Comparison with State-of-the-Art Methods

In Tables II, III, IV, and V, our Parameterized RV Loss is compared with state-of-the-art methods on five benchmarks including person re-ID and vehicle re-ID datasets, respectively.

**Results on Person re-ID.** We compare with various state-of-the-art person re-ID methods, *i.e.*, deep metric learning methods [4], [7], [46], [85]–[88], neural architecture search (NAS) methods [62], [63], and loss function (LFS) search methods [24]–[26]. The experimental results are shown in Table II. As shown in Table II, we can observe that Parameterized RV Loss achieves state-of-the-art results on three person re-ID benchmarks. On Market-1501, Parametrized RV Loss outperforms the previous state-of-the-art methods by a relatively small margin (+1.1% R1 and +0.9% mAP). On MSMT17 and CUHK03, Parametrized RV Loss achieves significant performance improvement (+1.3%/+2.8% R1 and +1.7%/+2.0% mAP). Compared with the hand-designed losses, our method can obtain about 2.0/3.0 R1/mAP score gain on Market-1501. It should be noted that NAS (*i.e.*, AutoReID [62] and CDNet [63]) and LFS (*i.e.*, AM-LFS [24], Search-Softmax [25], and LFS-ReID [26]) methods based on AutoML network architectures and AutoML-based loss functions, respectively, which can achieve better performance. Compared with NAS-based and LFS-based methods, our method obtain significant improvement over them. In addition, to assess the verification performance of our method, Table V shows the corresponding results on Market-1501. It is obvious that our method maintains comparable Precision/Recall score performance before the threshold 0.5.

**Results on Vehicle re-ID.** We demonstrate the effectiveness

Table V
COMPARISONS WITH THE STATE-OF-THE-ART RE-ID MODELS ON THE MARKET1501 AND VERI-776 DATASETS (%). THE RESULTS IN BOLD INDICATE THE BEST PERFORMANCE. WE REPORT THE RESULTS OF PRECISION (P) AND RECALL (R) TO EVALUATE THE VERIFICATION PERFORMANCE.

| Methods | Market-1501 | | | | | Methods | VeRi-776 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 | | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| | P/R | P/R | P/R | P/R | P/R | | P/R | P/R | P/R | P/R | P/R |
| BagofTricks [41] | 58.4/22.4 | 7.9/94.1 | 0.2/99.8 | 0.1/100.0 | 0.1/100.0 | BagofTricks [41] | 62.1/25.1 | 6.2/95.1 | 0.1/99.9 | 0.1/100.0 | 0.0/100.0 |
| AGW [102] | 2.0/0.3 | 64.6/31.5 | 62.0/79.8 | 1.3/97.8 | 0.1/100.0 | AGW [102] | 8.9/4.3 | 68.7/36.8 | 64.9/81.3 | 0.7/99.8 | 0.1/100.0 |
| **Our** | **60.3/34.3** | **65.2/98.6** | **64.8/100.0** | **0.1/100.0** | **0.1/100.0** | **Our** | **65.4/28.1** | **71.5/95.3** | **64.6/83.7** | **0.3/99.9** | **0.0/100.0** |

Table VI
EFFECT OF DIFFERENT SUBSTITUTIONS FOR $\mathbb{1}$, INCLUDING HAND-DESIGNED SUBSTITUTIONS (*i.e.*, SIGMOID, SQRT, LINEAR, AND SQUARE) AND OUR SEARCHED PARAMETERIZED FUNCTION.

| Differentiable Substitution | Market-1501 R1 | Market-1501 mAP | VeRi-776 R1 | VeRi-776 mAP |
|---|---|---|---|---|
| Sigmoid | 64.5 | 48.6 | 76.4 | 42.6 |
| Sqrt | 61.5 | 39.4 | 70.8 | 38.1 |
| Linear | 76.8 | 50.3 | 81.6 | 49.2 |
| Square | 90.5 | 81.3 | 92.4 | 73.8 |
| **Searched** | **96.3** | **88.3** | **97.3** | **81.2** |

Table VII
EFFECT OF CORRELATION BETWEEN DIFFERENT INDICATOR FUNCTION $\mathbb{1}$. WE CONDUCTED EXPERIMENTS INCLUDING SHARED AND SEPARATE PARAMETERS.

| Parameters | Market-1501 R1 | Market-1501 mAP | VeRi-776 R1 | VeRi-776 mAP |
|---|---|---|---|---|
| Shared | 94.7 | 84.3 | 95.1 | 77.2 |
| **Separate** | **96.3** | **88.3** | **97.3** | **81.2** |

of our Parameterized RV Loss on vehicle re-ID benchmarks, *i.e.*, VeRi-776 [81] and VehicleID [82]. We compare with current state-of-the-art methods on both datasets to show our performance advantage. Tables III and IV present the experimental results on VeRi-776 and VechicleID, respectively. We report CMC at Rank-1 and mAP on VeRi-776 dataset, as well as CMC at Rank-1 and Rank-5 on VehicleID dataset. From the numbers, we can observe that Parameterized RV Loss achieves state-of-the-art results on two vehicle re-ID benchmarks, especially on the challenging and large-scale VechicleID dataset where the performance improvement is significant. On VeRi-776, Parameterized RV Loss reaches 97.3%/81.2% R1/mAP surpassing SGFD by 0.6%/0.2% R1/mAP. We also conduct experiments on VechicleID in Table IV. It can be observed that our method also achieves state-of-the-art performance on challenging VechicleID. In detail, our method outperforms PNP-$D_q$ by 0.7% R1 for large protocol. These results demonstrate that our method is significantly effective. Like person re-ID, we also conducted experiments to verify the verification performance on VeRi-776. As Shown in Table V, we can observe that our method shows the performance advantage over the other two respective baseline methods.

**Summary.** In this section, we conduct relevant comparative experiments on person re-ID and vehicle re-ID datasets from the retrieval task and the verification task in Tables II, III, IV, and V. The experimental results demonstrate the effectiveness and superiority of our Parameterized RV Loss for the re-ID task. Meanwhile, these results demonstrate that our method can provide the consistency between training and evaluation, hence boosting performance consistently. It is worth noting that since we are the first time to investigate the retrieval and verification tasks simultaneously, the exploration we have done is still in the preliminary stage.

## D. Ablation Study

To study the effectiveness of different experimental designs and hyper-parameter setups, we conducted a series of ablation experiments on the Market-1501 and VeRi-776 datasets to verify the influence of each module or hyper-parameter in this section. In all the following experiments, we only varied one parameter of each ablation experiment at a time and other settings were kept the same for providing a fair understanding of these hyper-parameters.

**Ablation Study of Indicator Function.** To verify the superiority of the searched parameterized function $f_\theta(\cdot)$, we selected some commonly used carefully hand-designed differentiable substitutions in Table VI, including sigmoid, sqrt, linear, and square functions. The results show that the searched parameterized function achieves significantly better performance than the hand-designed substitutions, which also denotes that the hand-designed substitution does not guarantee the optimal results. Also note that although we use linear functions for optimization, there is a further performance improvement after imposing a searched parameterized function (*i.e.*, our designed parameterized function), which also demonstrates that the parameterized search process is effective.

**Ablation Study of Correlation between Different Indicator Function $\mathbb{1}$.** Table VII shows that using separate parameters for the indicator function $\mathbb{1}$ can have a 1.6%/2.2% R1 and 4.0%/4.0% mAP improvement over shared parameters on Market-1501 and VeRi-776, respectively. These results show that separate parameters is very flexible and different components of Equ. 13 used to search for the parameterized function can be closer to the desired loss function.

**Ablation Study of Different Measurements for Verification Score.** Table VIII shows the results of using Precision, Recall, and VP function for the verification score. Precision and Recall have similar results, and both have appropriate performance improvements with respect to Baseline (*i.e.*, ResNet50). Particularly, using our VP outperforms that of using

Table VIII
EFFECT OF DIFFERENT MEASUREMENTS FOR THE VERIFICATION SCORE. WE SELECTED THREE MEASUREMENTS: PRECISION, RECALL, AND VP DEFINED IN THIS PAPER.

| Measurement | Market-1501 | | VeRi-776 | |
|---|---|---|---|---|
| | R1 | mAP | R1 | mAP |
| Precision | 95.7 | 85.9 | 96.4 | 78.6 |
| Recall | 94.2 | 85.1 | 95.8 | 78.2 |
| **VP** | **96.3** | **88.3** | **97.3** | **81.2** |

Table IX
EFFECT OF DIFFERENT NUMBER OF SEGMENTS IN THE PIECEWISE FUNCTION $f_\theta(\cdot)$.

| Segments | Market-1501 | | VeRi-776 | |
|---|---|---|---|---|
| | R1 | mAP | R1 | mAP |
| 3 | 90.6 | 83.6 | 91.4 | 75.4 |
| 4 | 94.9 | 87.6 | 96.6 | 78.7 |
| **5** | **96.3** | **88.3** | **97.3** | **81.2** |
| 6 | 96.1 | 88.1 | 96.9 | 80.8 |
| 7 | 95.7 | 87.6 | 96.4 | 80.1 |

Table X
EFFECT OF DIFFERENT BATCH SIZE DURING TRAINING. THE NUMBER OF BATCH SIZE BY SETTING AS 32, 64, 128, 256 ON MARKET-15 AND VERI-776.

| Batch Size | Market-1501 | | VeRi-776 | |
|---|---|---|---|---|
| | R1 | mAP | R1 | mAP |
| 32 | 95.7 | 87.8 | 96.7 | 80.2 |
| 64 | 96.3 | 88.3 | 97.3 | 81.2 |
| 128 | 96.2 | 88.4 | 97.1 | 81.0 |
| **256** | **96.4** | **88.4** | **97.5** | **81.4** |

Precision or Recall, which is that VP combines the advantages of Precision and Recall.

**Ablation Study of Different Number of Segments in $f_\theta(\cdot)$.** To explore the impact of different number of segments on performance, the results are shown in Table IX. On the one hand, too less segments will lead to restricted expression of the diversity of $f_\theta(\cdot)$, dropping the performance significantly. On the other hand, too many segments may increase the search complexity, leading to performance shake. Experiments show that 5 is the optimal number of segments.

**Ablation Study of Different Batch Size during Training.** Table X shows that compared to other numbers, 64 is the optimal batch size to achieve optimal performance. It also shows that the performance of the model has reached saturation when the batch size is 64 rather than larger. This is expected, as the larger the batch size, the more diverse the samples in the min-batch, which increase the probability of getting hard-negative samples in the batch. In Table X, there have achieved the best results when the batch size is 256. It can be noticed that the performance of a batch size of 256 has only a slight improvement compared to 64, *i.e.*, +0.1%/+0.1% R1/mAP on Market-1501 and 0.2%/0.2% R1/mAP VeRi-776, respectively. From the comprehensive perspective of performance and computer resource utilization, the paper chooses a batch size of 64 to conduct experiments and comparisons.

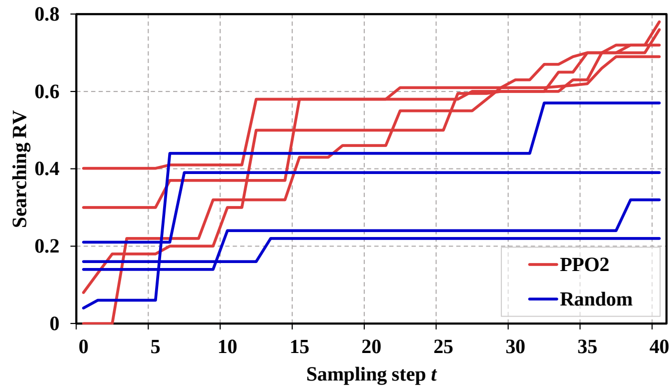


Figure 4. Ablation study of different HPO algorithms including PPO2 and random search on Market-1501. Each curve shows the highest average RV reward score up to the $t$-th round during one search procedure. We conducted the search process four times in total.

Table XI
COMPARISON OF THE COMPUTATIONAL COST (RTX 2080TI GPU) ON MARKET-1501 AND VERI-776 USING THE RESNET50 BACKBONE.

| Datasets | Methods | Number of GPU | Total Time |
|---|---|---|---|
| Market-1501 | AM-LFS [24] | 1 | 16.26 hours |
| | Search-Softmax [25] | 1 | 16.44 hours |
| | AutoLoss-GMS-A [27] | 5 | 61.44 hours |
| | **Our** | **4** | **15.12 hours** |
| VeRi-776 | AM-LFS [24] | 1 | 37.38 hours |
| | Search-Softmax [25] | 1 | 37.80 hours |
| | AutoLoss-GMS-A [27] | 5 | 163.44 hours |
| | **Our** | **4** | **34.74 hours** |

**Ablation Study of Different Search Algorithm.** As shown in Figure 4, PPO2 achieves better parameters compared to random search, which confirms that searching the loss function is difficult, but reinforcement learning can speed up the loss function search process.

### *E. Further Analysis*

**Analysis of computational cost.** Table XI gives the comparison of the computational cost with three AutoML-based loss functions on Market-1501 and VeRi-776, including AM-LFS [24], Search-Softmax [25] and AutoLoss-GMS-A [27]. The backbone of the compared methods are all used ResNet50 and the experimental hardware resources keep the same as our method. We can observe that the computational cost on Market-1501 is less than that on VeRi-776, due to the fact that VeRi-776 has more training data than Market-1501. Among them, the computational cost of AutoLoss-GMS-A [27] is the largest, 61.44 hours and 163.44 hours on Market-1501 and VeRi-776, respectively. In addition, AutoLoss-GMS-A uses a total of 5 GPUs. Compared to the rest of the AutoML-based methods, it can be observed that the computational cost is similar. Although the total time of our method is the shortest compared to other methods, our method requires 4 GPUs. The main reason is that our method needs to search for better parameters in the search space, which requires more hardware resources like AutoLoss-GMS-A.

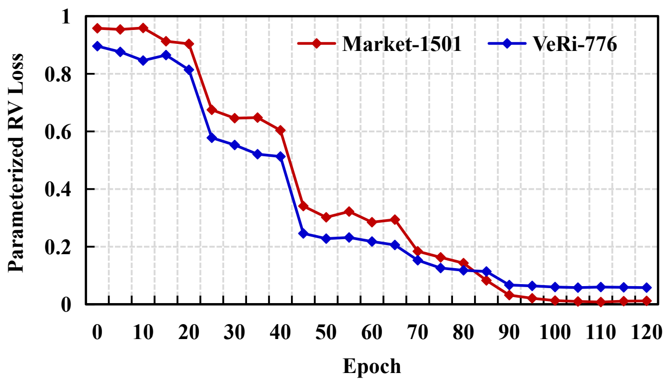


Figure 5. Convergence analysis: Illustration of Parameterized RV Loss curves on Market-1501 and VeRi-776, repectively.

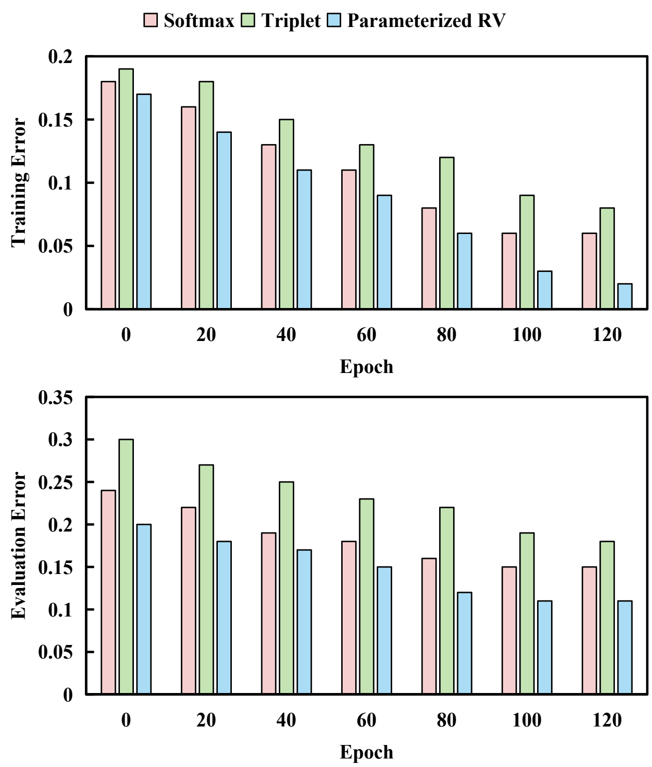


Figure 6. Inconsistency analysis: Comparison of training error and evaluation error on Market-1501. We have selected commonly used softmax loss and triplet loss in re-ID versus our proposed Parameterized RV Loss.

**Analysis of convergence.** We also study the convergence of Parameterized RV Loss. It can be seen from Figure 5 that Parameterized RV Loss tends to be close to 0, which indicates that our method is well trained on both Market-1501 and VeRi-776. We can conclude that our designed Parameterized Loss matches the re-ID task, and it is able to be fully trained and finally converged. Therefore, Figure 5 indicates that our searched parametric RV loss can be effectively adapted to the re-ID task, which provides a new insight for re-ID studies.

**Analysis of inconsistency.** To depict that the inconsistency of existing loss functions may lead to performance degradation, we conduct the inconsistency analysis following [19]. Specifically, we have selected the commonly used softmax loss [38] and triplet loss [39] in re-ID, and compared the training and evaluation errors together with our Parameterized RV Loss on the training set $D_{train}$ and the validation set $D_{val}$, as shown in Figure 6. Compared to softmax loss and triplet loss, the corresponding training and evaluation errors of Parameterized RV Loss are the minimum. It can be seen that our Parameterized RV Loss improves the training error and the evaluation error, which demonstrates that Parameterized RV Loss indeed mitigates the inconsistency between the loss functions and evaluation metrics. Therefore, it can significantly improve the discrimination of the learned feature.

## VI. Conclusion

In this paper, we propose a novel AutoML-based loss named Parameterized RV Loss for re-ID, which jointly optimizes both RV tasks and alleviates the inconsistency between the loss functions and evaluation metrics. Parameterized RV Loss achieves the automatic searching of loss functions, *i.e.*, parameter-free. Parameterized RV loss is a single unified loss function taking both retrieval and verification tasks into account, and it consistently outperforms existing loss functions on several re-ID methods, including some hand-designed loss functions and loss function search-based loss functions. Although we have demonstrated that the parameterized search process is effective, the piecewise linear functions used in our work are not necessarily optimal differentiable substitution functions. Meanwhile, there are still open problems worth studying, such as the design of evaluation metrics and the optimization of searching strategy, which are interesting and challenging works in the future.

**Limitations.** With the experimental results, it can be observed that Parameterized RV Loss still requires lots of time and hardware resources for searching better parameters, and the model performance may degrade without enough search time. Future work may explore more efficient HPO methods to reduce the search time. We will also plan to design and optimize the searching framework on more relevant tasks to further verify the effectiveness of our method.

## References

[1] X. Xu, X. Yuan, Z. Wang, K. Zhang, and R. Hu, "Rank-in-rank loss for person re-identification," *ACM Trans. Multimedia Comput. Commun. Appl. (TOMM)*, pp. 1–20, 2022.
[2] Z. Wang, X. Yuan, T. Yamasaki, Y. Lin, X. Xu, and W. Zeng, "Re-identification= retrieval+ verification: Back to essence and forward with a new metric," *arXiv preprint arXiv:2011.11506*, 2020.
[3] Z. Wan, X. Xu, Z. Wang, T. Yamasaki, X. Zhang, and R. Hu, "Efficient virtual data search for annotation-free vehicle reidentification," *Int. J. Intell. Syst. (IJIS)*, vol. 37, no. 5, pp. 2988–3005, 2022.
[4] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737v4*, 2017.
[5] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: a deep quadruplet network for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 403–412.
[6] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 994–1003.
[7] Z. Zheng, L. Zheng, and Y. Yang, "A discriminatively learned cnn embedding for person reidentification," *ACM Trans. Multimedia Comput. Commun. Appl. (TOMM)*, vol. 14, no. 1, pp. 1–20, 2017.

[8] X. Xu, L. Liu, X. Zhang, W. Guan, and R. Hu, "Rethinking data collection for person re-identification: active redundancy reduction," *Pattern Recogn.*, vol. 113, p. 107827, 2021.

[9] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016, pp. 1335–1344.

[10] W. Chen, X. Chen, J. Zhang, and K. Huang, "A multi-task deep network for person re-identification," in *AAAI Conf. Artif. Intell. (AAAI)*, 2017, pp. 3988–3994.

[11] A. Khatun, S. Denman, S. Sridharan, and C. Fookes, "A deep four-stream siamese convolutional neural network with joint verification and identification loss for person re-detection," in *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2018, pp. 1292–1301.

[12] C. Song, Y. Huang, W. Ouyang, and L. Wang, "Mask-guided contrastive attention model for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 1179–1188.

[13] H. Tian, X. Zhang, L. Lan, and Z. Luo, "Person re-identification via adaptive verification loss," *Neurocomputing*, vol. 359, pp. 93–101, 2019.

[14] K. Zheng, C. Lan, W. Zeng, Z. Zhang, and Z.-J. Zha, "Exploiting sample uncertainty for domain adaptive person re-identification," in *AAAI Conf. Artif. Intell. (AAAI)*, 2021, pp. 3538–3546.

[15] H. Lee, S. Eum, and H. Kwon, "Negative samples are at large: Leveraging hard-distance elastic loss for re-identification," in *Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 1–19.

[16] A. Khatun, S. Denman, S. Sridharan, and C. Fookes, "Joint identification-verification for person re-identification: A four stream deep learning approach with improved quartet loss function," *Comput. Vision Image Understanding (CVIU)*, vol. 197-198, p. 102989, 2020.

[17] X. Xu, W. Liu, Z. Wang, R. Hu, and Q. Tian, "Towards generalizable person re-identification with a bi-stream generative model," *Pattern Recogn.*, p. 108954, 2022.

[18] P. Xie, X. Xu, Z. Wang, and T. Yamasaki, "Sampling and re-weighting: Towards diverse frame aware unsupervised video person re-identification," *IEEE Trans. Multimedia (TMM)*, pp. 1–13, 2022.

[19] C. Huang, S. Zhai, W. Talbott, M. B. Martin, S.-Y. Sun, C. Guestrin, and J. Susskind, "Addressing the loss-metric mismatch with adaptive loss alignment," in *Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 2891–2900.

[20] K. Oksuz, B. C. Cam, E. Akbas, and S. Kalkan, "Rank & sort loss for object detection and instance segmentation," in *Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 3009–3018.

[21] F. Cakir, K. He, X. Xia, B. Kulis, and S. Sclaroff, "Deep metric learning to rank," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 1861–1870.

[22] A. Brown, W. Xie, V. Kalogeiton, and A. Zisserman, "Smooth-ap: Smoothing the path towards large-scale image retrieval," in *Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 677–694.

[23] Z. Li, W. Min, J. Song, Y. Zhu, L. Kang, X. Wei, X. Wei, and S. Jiang, "Rethinking the optimization of average precision: Only penalizing negative instances before positive ones is enough," in *AAAI Conf. Artif. Intell. (AAAI)*, 2022, pp. 1518–1526.

[24] C. Li, X. Yuan, C. Lin, M. Guo, W. Wu, J. Yan, and W. Ouyang, "Am-lfs: Automl for loss function search," in *Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 8410–8419.

[25] X. Wang, S. Wang, C. Chi, S. Zhang, and T. Mei, "Loss function search for face recognition," in *Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 10 029–10 038.

[26] H. Gu, J. Li, G. Fu, M. Yue, and J. Zhu, "Loss function search for person re-identification," *Pattern Recogn.*, vol. 124, p. 108432, 2022.

[27] H. Gu, J. Li, G. Fu, C. Wong, X. Chen, and J. Zhu, "Autoloss-gms: Searching generalized margin-based softmax loss function for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 4744–4753.

[28] H. Li, T. Fu, J. Dai, H. Li, G. Huang, and X. Zhu, "Autoloss-zero: Searching loss functions from scratch for generic tasks," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 1009–1018.

[29] K. Tian, C. Lin, M. Sun, L. Zhou, J. Yan, and W. Ouyang, "Improving auto-augment via augmentation-wise weight sharing," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2020, pp. 19 088–19 098.

[30] A. Liu, Z. Huang, Z. Huang, and N. Wang, "Direct differentiable augmentation search," in *Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 12 219–12 228.

[31] Y. Zheng, Z. Zhang, S. Yan, and M. Zhang, "Deep autoaugment," *arXiv preprint arXiv:2203.06172*, 2022.

[32] X. Zhang, Z. Huang, N. Wang, S. Xiang, and C. Pan, "You only search once: Single shot neural architecture search via direct sparse optimization," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 43, no. 9, pp. 2891–2904, 2021.

[33] Y. Wu, A. Liu, Z. Huang, S. Zhang, and L. Van Gool, "Neural architecture search as sparse supernet," in *AAAI Conf. Artif. Intell. (AAAI)*, 2021, pp. 10 379–10 387.

[34] X. Liu, J. Zhao, J. Li, B. Cao, and Z. Lv, "Federated neural architecture search for medical data security," *IEEE Trans. Ind. Inf. (TII)*, vol. 18, no. 8, pp. 5628–5636, 2022.

[35] P. Liu, G. Zhang, B. Wang, H. Xu, X. Liang, Y. Jiang, and Z. Li, "Loss function discovery for object detection via convergence-simulation driven search," in *Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–11.

[36] H. Li, C. Tao, X. Zhu, X. Wang, G. Huang, and J. Dai, "Auto seg-loss: Searching metric surrogates for semantic segmentation," in *Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–15.

[37] T. Chenxin, Z. Li, X. Zhu, G. Huang, Y. Liu *et al.*, "Searching parameterized ap loss for object detection," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2021, pp. 22 021–22 033.

[38] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Y. Yang, and Q. Tian, "Person re-identification in the wild," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 1367–1376.

[39] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2015, pp. 815–823.

[40] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 3754–3762.

[41] H. Luo, W. Jiang, Y. Gu, F. Liu, X. Liao, S. Lai, and J. Gu, "A strong baseline and batch normalization neck for deep person re-identification," *IEEE Trans. Multimedia (TMM)*, vol. 22, no. 10, pp. 2597–2609, 2019.

[42] L. Wu, Y. Wang, J. Gao, M. Wang, Z.-J. Zha, and D. Tao, "Deep coattention-based comparator for relative representation learning in person re-identification," *IEEE Trans. Neur. Net. Learn. Syst. (TNNLS)*, vol. 32, no. 2, pp. 722–735, 2021.

[43] L. Wu, D. Liu, W. Zhang, D. Chen, Z. Ge, F. Boussaid, M. Bennamoun, and J. Shen, "Pseudo-pair based self-similarity learning for unsupervised person re-identification," *IEEE Trans. Image Process. (TIP)*, vol. 31, pp. 4803–4816, 2022.

[44] L. Wu, Y. Wang, H. Yin, M. Wang, and L. Shao, "Few-shot deep adversarial learning for video-based person re-identification," *IEEE Trans. Image Process. (TIP)*, vol. 29, no. 1, pp. 1233–1245, 2020.

[45] D. Liu, L. Wu, R. Hong, Z. Ge, J. Shen, F. Boussaid, and M. Bennamoun, "Generative metric learning for adversarially robust open-world person re-identification," *ACM Trans. Multimedia Comput. Commun. Appl. (TOMM)*, pp. 1–20, 2022.

[46] X. Fan, W. Jiang, H. Luo, and M. Fei, "Spherereid: Deep hypersphere manifold embedding for person re-identification," *J. Visual Commun. Image Represent. (JVCIR)*, vol. 60, pp. 51–58, 2019.

[47] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020, pp. 6398–6407.

[48] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 403–412.

[49] K. Oksuz, B. C. Cam, E. Akbas, and S. Kalkan, "A ranking-based, balanced loss function unifying classification and localisation in object detection," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2020, pp. 15 534–15 545.

[50] J. Revaud, J. Almazan, R. S. Rezende, and C. R. d. Souza, "Learning with average precision: Training image retrieval with a listwise loss," in *Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 5107–5116.

[51] E. Ramzi, N. Thome, C. Rambour, N. Audebert, and X. Bitot, "Robust and decomposable average precision for image retrieval," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2021, pp. 23 569–23 581.

[52] K. He, F. Cakir, S. A. Bargal, and S. Sclaroff, "Hashing as tie-aware learning to rank," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 4023–4032.

[53] R. Benham, J. Mackenzie, A. Moffat, and J. S. Culpepper, "Boosting search performance using query variations," *ACM Trans. Inf. Syst. (TOIS)*, vol. 37, no. 4, pp. 1–25, 2019.

[54] M. Ranjbar, T. Lan, Y. Wang, S. N. Robinovitch, Z.-N. Li, and G. Mori, "Optimizing nondecomposable loss functions in structured prediction," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 35, no. 4, pp. 911–924, 2012.

[55] P. Mohapatra, C. Jawahar, and M. P. Kumar, "Efficient optimization for average precision svm," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2014, pp. 2312–2320.

[56] Y. Song, A. Schwing, R. Urtasun *et al.*, "Training deep neural networks via direct loss minimization," in *Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2169–2177.

[57] P. Mohapatra, M. Rolinek, C. Jawahar, V. Kolmogorov, and M. P. Kumar, "Efficient optimization for rank-based loss functions," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 3693–3701.

[58] M. Rolínek, V. Musil, A. Paulus, M. Vlastelica, C. Michaelis, and G. Martius, "Optimizing rank-based metrics with blackbox differentiation," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020, pp. 7620–7630.

[59] K. Oksuz, B. C. Cam, E. Akbas, and S. Kalkan, "Localization recall precision (lrp): A new performance metric for object detection," in *Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 504–519.

[60] K. Chen, W. Lin, J. See, J. Wang, J. Zou *et al.*, "Ap-loss for accurate one-stage object detection," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 43, no. 11, pp. 3782–3798, 2021.

[61] M. Engilberge, L. Chevallier, P. Pérez, and M. Cord, "Sodeep: a sorting deep net to learn ranking loss surrogates," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 10 792–10 801.

[62] R. Quan, X. Dong, Y. Wu, L. Zhu, and Y. Yang, "Auto-reid: Searching for a part-aware convnet for person re-identification," in *Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 3750–3759.

[63] H. Li, G. Wu, and W.-S. Zheng, "Combined depth space based architecture search for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2021, pp. 6729–6738.

[64] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res. (JMLR)*, vol. 13, no. 2, pp. 281–305, 2012.

[65] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2011, pp. 1–9.

[66] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *International Conference on Learning and Intelligent Optimization*, 2011, pp. 507–523.

[67] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2012, pp. 1–9.

[68] J. T. Springenberg, A. Klein, S. Falkner, and F. Hutter, "Bayesian optimization with robust bayesian neural networks," in *Annu. Conf. Neur. Inform. Process. Syst. (NeurIPS)*, 2016, pp. 1–9.

[69] E. Real, S. Moore, A. Selle, S. Saxena, Y. L. Suematsu, J. Tan, Q. V. Le, and A. Kurakin, "Large-scale evolution of image classifiers," in *Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 2902–2911.

[70] T. Elsken, J. H. Metzen, and F. Hutter, "Efficient multi-objective neural architecture search via lamarckian evolution," *arXiv preprint arXiv:1804.09081*, 2018.

[71] J. Degrave, F. Felici, J. Buchli, M. Neunert, B. Tracey, F. Carpanese, T. Ewalds, R. Hafner, A. Abdolmaleki, D. de Las Casas *et al.*, "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414–419, 2022.

[72] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2013, pp. 3610–3617.

[73] R. Kushwaha and N. Nain, "Pug-fb: Person-verification using geometric and haralick features of footprint biometric," *Multimedia Tools and Applications*, vol. 79, no. 3, pp. 2671–2701, 2020.

[74] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *PETS*, 2007, pp. 1–7.

[75] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1116–1124.

[76] K. Chen, J. Li, W. Lin, J. See, J. Wang, L. Duan, Z. Chen, C. He, and J. Zou, "Towards accurate one-stage object detection with ap-loss," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 5119–5127.

[77] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[78] J. Burkardt, "The truncated normal distribution," *Department of Scientific Computing Website, Florida State University*, vol. 1, p. 35, 2014.

[79] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 79–88.

[80] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2014, pp. 152–159.

[81] X. Liu, W. Liu, T. Mei, and H. Ma, "A deep learning-based approach to progressive vehicle re-identification for urban surveillance," in *Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 869–884.

[82] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016, pp. 2167–2175.

[83] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 1318–1327.

[84] K. Li, Z. Ding, K. Li, Y. Zhang, and Y. Fu, "Support neighbor loss for person re-identification," in *ACM Int. Conf. Multimedia (ACM MM)*, 2018, pp. 1492–1500.

[85] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 480–496.

[86] Y. Huang, Y. Huang, H. Hu, D. Chen, and T. Su, "Deeply associative two-stage representations learning based on labels interval extension loss and group loss for person re-identification," *IEEE Trans. Circ. Syst. Video Technol. (TCSVT)*, vol. 30, no. 12, pp. 4526–4539, 2019.

[87] Q. Zhou, B. Zhong, X. Lan, G. Sun, Y. Zhang, B. Zhang, and R. Ji, "Fine-grained spatial alignment model for person re-identification with focal triplet loss," *IEEE Trans. Image Process. (TIP)*, vol. 29, pp. 7578–7589, 2020.

[88] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020, pp. 6398–6407.

[89] X. Zhu, Z. Luo, P. Fu, and X. Ji, "Voc-reid: Vehicle re-identification based on vehicle-orientation-camera," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh. (CVPRW)*, 2020, pp. 602–603.

[90] R. Chu, Y. Sun, Y. Li, Z. Liu, C. Zhang, and Y. Wei, "Vehicle re-identification with viewpoint-aware metric learning," in *Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 8282–8291.

[91] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang, "Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 8797–8806.

[92] B. He, J. Li, Y. Zhao, and Y. Tian, "Part-regularized near-duplicate vehicle re-identification," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 3997–4005.

[93] X. Jin, C. Lan, W. Zeng, and Z. Chen, "Uncertainty-aware multi-shot knowledge distillation for image-based object re-identification," in *AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 11 165–11 172.

[94] X. Zhang, R. Zhang, J. Cao, D. Gong, M. You, and C. Shen, "Part-guided attention learning for vehicle instance retrieval," *IEEE Trans. Intell. Transp. Syst. (TITS)*, vol. 23, no. 4, pp. 3048–3060, 2020.

[95] P. Khorramshahi, N. Peri, J.-c. Chen, and R. Chellappa, "The devil is in the details: Self-supervised attention for vehicle re-identification," in *Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 369–386.

[96] Z. Sun, X. Nie, X. Xi, and Y. Yin, "Cfvmnet: A multi-branch network for vehicle re-identification based on common field of view," in *ACM Int. Conf. Multimedia (ACM MM)*, 2020, pp. 3523–3531.

[97] M. Li, X. Huang, and Z. Zhang, "Self-supervised geometric features discovery via interpretable attention for vehicle re-identification and beyond," in *Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 194–204.

[98] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2009, pp. 248–255.

[99] A. Sanakoyeu, V. Tschernezki, U. Buchler, and B. Ommer, "Divide and conquer the embedding space for metric learning," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 471–480.

[100] K. Roth, B. Brattoli, and B. Ommer, "Mic: Mining interclass characteristics for improved metric learning," in *Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 8000–8009.

[101] X. Wang, H. Zhang, W. Huang, and M. R. Scott, "Cross-batch memory for embedding learning," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020, pp. 6388–6397.

[102] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 44, no. 6, pp. 2872–2893, 2022.