CM-NAS: Cross-Modality Neural Architecture Search
for Visible-Infrared Person Re-Identification

Chaoyou Fu[1,2*]   Yibo Hu[3*]   Xiang Wu[2]   Hailin Shi[3]   Tao Mei[3]   Ran He[1,2†]

[1] School of Artificial Intelligence, University of Chinese Academy of Sciences
[2] NLPR & CEBSIT & CRIPAC, CASIA   [3] JD AI Research

{chaoyou.fu, rhe}@nlpr.ia.ac.cn, {shihailin, tmei}@jd.com
{huyibo871079699,alfredxiangwu}@gmail.com

Abstract

Visible-Infrared person re-identification (VI-ReID) aims to match cross-modality pedestrian images, breaking through the limitation of single-modality person ReID in dark environment. In order to mitigate the impact of large modality discrepancy, existing works manually design various two-stream architectures to separately learn modality-specific and modality-sharable representations. Such a manual design routine, however, highly depends on massive experiments and empirical practice, which is time consuming and labor intensive. In this paper, we systematically study the manually designed architectures, and identify that appropriately separating Batch Normalization (BN) layers is the key to bring a great boost towards cross-modality matching. Based on this observation, the essential objective is to find the optimal separation scheme for each BN layer. To this end, we propose a novel method, named Cross-Modality Neural Architecture Search (CM-NAS). It consists of a BN-oriented search space in which the standard optimization can be fulfilled subject to the cross-modality task. Equipped with the searched architecture, our method outperforms state-of-the-art counterparts in both two benchmarks, improving the Rank-1/mAP by 6.70%/6.13% on SYSU-MM01 and by 12.17%/11.23% on RegDB. Code is released at https://github.com/JDAI-CV/CM-NAS.

1. Introduction

Person re-identification (ReID) refers to matching pedestrian images acquired from disjoint cameras [20, 53, 62]. In recent years, it has received substantial attention due to its significant practical value in video surveillance [47]. Conventional person ReID is only devoted to single-modality,

i.e. all the pedestrian images are taken by visible cameras during day time. Benefiting from the strenuous efforts of researchers, impressive achievements have been made on most benchmarks [57, 1, 54]. Nevertheless, the visible cameras cannot image clearly in dark environment, which impedes the popularization and application of person ReID [39]. To overcome this obstacle, in addition to the visible (VIS) cameras, infrared (IR) cameras that are robust to illumination variants are also equipped in many surveillance scenarios [5]. Therefore, in practice, we often need to match VIS and IR pedestrian images, raising the task of VI-ReID.

Modality discrepancy, caused by different wavelengths of VIS and IR images, is one of the most difficult challenges in VI-ReID. Existing works have manually designed various two-stream architectures [49, 50, 24] to mitigate the impact of the large modality discrepancy. Specifically, as exemplified in Fig. 2, some layers are separated into two branches to learn modality-specific representations for VIS an IR data respectively, while the remaining layers are shared to learn modality-sharable representations. The intuition behind this design is that VIS and IR images contain both modality-specific information, *e.g.* the spectrum, and modality-sharable information, *e.g.* the identity. At this point, an obvious problem is raised: which layers should be separated into two branches and which layers should be shared? Some methods separate the layers in the first one [47, 50] or five [52] blocks, while some others even share the whole network [42]. There is still no consensus on the optimal design of the neural architecture for VI-ReID.

In this paper, to investigate the impact of different separation schemes, we manually design a total of **195** different two-stream architectures. Given that Batch Normalization (BN) plays a crucial role in learning modality distributions [45], we also perform separation in units of BN layers, in addition to the entire block as usual. As illustrated in Section 3.1, after comprehensively comparing the performances of all the architectures, we obtain two major

observations: (1) only separating BN layers in the block is superior than separating the entire block; (2) separating two blocks of BN layers generally outperforms separating a single one. With these in mind, we arrive at a conclusion that **appropriately separating all BN layers is the key to bring a great boost towards cross-modality matching**. As a consequence, the essential objective is to find the optimal separation scheme for each BN layer in the backbone. However, there are a great deal of potential separation schemes. Specifically, the backbone ResNet50 [14] contains 53 BN layers, leading to a total of 2^{53} possible architectures. It is time consuming and labor intensive to manually traverse through all the possible architectures. To tackle this intractable problem, inspired by recently thriving Neural Architecture Search (NAS) technique [25, 16, 34, 6], we propose a novel Cross-Modality NAS (CM-NAS) to automatically determine whether each BN layer separates or not. A BN-oriented search space is elaborately built in which the standard optimization can be fulfilled subject to the cross-modality task. Note that it is infeasible to directly apply existing single-modality NAS methods like Auto-ReID [30], because its search space is powerless to bridge the modality discrepancy, as discussed in Section 4.2. In contrast, our designed search space supports to learn both modality-specific and modality-sharable representations via switching corresponding separating and sharing operations, which first opens the door of NAS to cross-modality matching.

Without bells and whistles, our method exceeds all state-of-the-art methods on both two VI-ReID benchmarks. On SYSU-MM01, our method achieves an improvement of **6.70%** and **6.13%** in terms of the Rank-1 accuracy and the mAP score. On RegDB, our method promotes the two indicators by **12.17%** and **11.23%**, respectively. Compared with the baseline ResNet50, our method increases the Rank-1/mAP by **7.50%/6.70%** on SYSU-MM01 and by **8.73%/8.35%** on RegDB, with a small additional parameters and no extra computational costs. We hope this simple yet effective method will be a solid foundation to facilitate future research in VI-ReID.

To sum up, we make the following three contributions:

- We systematically analyze **195** different manually designed architectures, and identify the significance of separating BN layers. This conclusion motivates us to develop a BN-oriented search algorithm.

- A novel CM-NAS is proposed to automatically search the optimal separation scheme for BN layers, which fills the blank of NAS in cross-modality matching.

- Our method significantly surpasses state-of-the-art competitors in both two benchmarks, improving the Rank-1/mAP by **6.70%/6.13%** on SYSU-MM01 and by **12.17%/11.23%** on RegDB. Code will be released to aid future research in VI-ReID.

2. Related Works

Single-Modality Person ReID. The goal of single-modality person ReID is to match pedestrian images across non-overlapping visible cameras [11, 59, 55, 10]. Existing works can be divided into three categories, including hand-crafted descriptors methods [22, 58], metric learning methods [60, 23] and deep learning methods [33, 61, 38, 15, 44]. However, the visible cameras cannot image clearly in dark environment, which impedes the popularization and application of the single-modality person ReID [56, 39].

Visible-Infrared Person ReID. Different from the aforementioned single-modality person ReID, VI-ReID matches pedestrian images belonging to different modalities. Due to the great practical value of VI-ReID, it has received substantial attention in recent years [9, 39, 26, 50]. The pioneer work [42] contributes a new VI-ReID dataset named SYSU-MM01 and introduces a deep zero-padding method, which explores modality-specific information in a one-stream network. [5] proposes a cross-modality generative adversarial framework to reduce the modality discrepancy. [52] learns discriminative cross-modality features via an elaborately designed dual-path network and a bi-directional dual-constrained top-ranking loss. [13] leverages a two-stream HSMEnet to map both representation learning and metric learning on to a hypersphere manifold.

Neural Architecture Search. Existing NAS works can be grouped into micro search methods and macro search methods [16]. The micro search methods aim to design robust cells and then stack these cells to constitute the neural architecture. Traditional methods mainly depend on evolutionary algorithms or reinforcement learning to tackle the discrete search problem [31, 63]. Recently, DARTS [25] first proposes a differentiable search strategy, which greatly reduces the computational overhead compared with the traditional methods [7, 46, 4]. The macro search methods search the whole neural architecture, which is thought to be more flexible than searching cells [3, 35]. [2] introduces Q-learning to select layers sequentially. [43] searches accurate and efficient architectures for mobile devices. However, these methods are all designed for single-modality tasks, with no need for considering the modality discrepancy.

3. Method

3.1. Analyses of Manually Designed Architectures

For fair comparisons with other VI-ReID methods, we employ ResNet50[1] pre-trained on ImageNet as the backbone, whose architecture is reported in Fig. 2. There are 5 stages in ResNet50, where each stage consists of blocks,
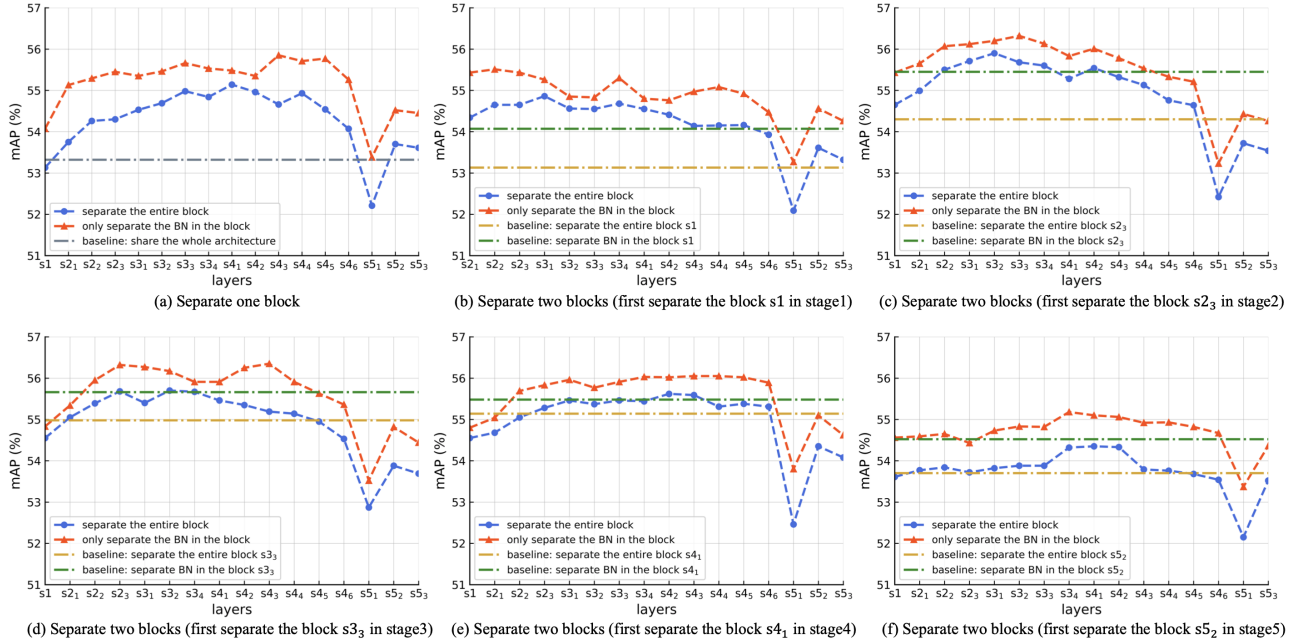
---

[1]Other networks are also applicable.

Figure 1. Comparisons of different ResNet50-based architecture designs on SYSU-MM01 under the challenging *single-shot&all-search* setting [42]. 's2$_1$' denotes the 1-st block in stage2 of ResNet50, whose architecture is depicted in Fig. 2. We design the architecture in units of the entire block (the blue line) or only the BN layers in the block (the red line). (a) shows the results of merely separating one block. (b)-(f) present the results of separating two blocks, where we first fixedly separate a block and then traverse through the remaining stages to separate the other one. Note that for (c), we choose to first fixedly separate the block 's2$_3$' rather than 's2$_1$' or 's2$_2$' in stage2, because 's2$_3$' performs better than the others in (a). (d)-(f) are also in the same way. It is obvious that separating BN layers significantly outperforms separating the entire block, which motivates us to explore more BN separation schemes.
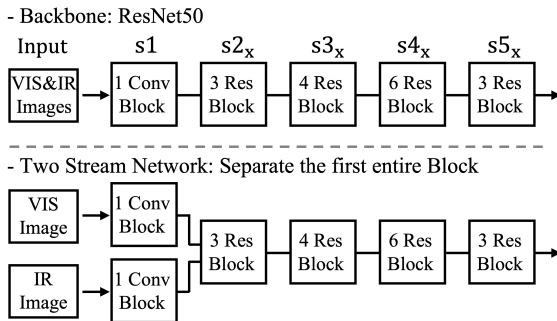

Figure 2. **Top:** The architecture of the backbone ResNet50 [14] that includes 5 stages. 's2$_x$' denotes the x-th ResBlock in stage2. A ConvBlock has one convolution layer, one BN layer [17] and one ReLU layer. A ResBlock contains a total of three convolution layers, three BN layers and three ReLU layers. Please refer to [14] for details. **Bottom:** An example of the two stream network, where the first entire block is separated into two branches for VIS and IR images, respectively.

including ConvBlock and ResBlock. As mentioned in Section 1, due to the absence of consensus on the optimal architecture, we perform systematic research to assess the impact of different architecture designs. In general, the architecture is designed in units of the entire block [49, 50], such as separating one or five blocks into two branches [52, 24].

Besides, we also design the architecture in units of the BN layers in the block, *i.e.* only separating the BN layers rather than all the layers in the block. This is inspired by [45], which reveals that separating BN layers for the data from different domains outperforms sharing the BN layers.

First of all, we evaluate the performance of separating one block, including separating all the layers in the block and solely separating the BN layers in the block. Concretely, we separate the blocks in ResNet50 one by one to learn modality-specific representations, and share the remaining to learn modality-sharable representations. Fig. 1 (a) depicts the results of all the potential architectures as well as the result of one baseline: sharing the whole architecture without separation. We have three observations from Fig. 1 (a): (1) the baseline generally performs worse than separating one block, suggesting the necessity of separating blocks to learn modality-specific representations; (2) separating different blocks yields much different performances. For example, when separating the entire block, 's4$_1$' and 's5$_1$' lead to the best and the worst results, respectively. This implies that we need to carefully treat each layer in the design process; (3) separating the BN layers in the block is more suitable than separating the entire block, since the former (the red line) gains much better results than the latter (the blue line) in all separation schemes. Subsequently,

we further separate two blocks for each time. Fig. 1 (b)-(f) display the results when we first fixedly separate a block in stage1, stage2, stage3, stage4 and stage5 respectively, and then traverse through the remaining stages to separate the other one. The performance of separating one block in the first step is also reported as a baseline result. A distinct observation from these results is that separating two blocks generally outperforms than separating a single one, especially when only separating the BN layers in the block.

With these observations, we arrive at a conclusion that appropriately separating BN layers can lead to better performances. Consequently, the essential objective is to find the optimal separation scheme for each BN layer in the backbone. As mentioned in Section 1, since it is intractable to manually traverse through all potential architectures, we develop a novel CM-NAS to automatically find the best one.

3.2. Cross-Modality NAS

Search Space. In view of the above analyses, our architecture design revolves around which BN layers should be separated and which BN layers should be shared. With this in mind, we design a BN-oriented search space that is depicted in Fig. 3. In our search space, all BN layers in the backbone are reshaped as searchable units, and each BN layer has two candidate operations: employing separate or shared parameters. If a BN layer chooses the former, this BN layer will have two separate parameters that are learned from VIS and IR data, respectively. Otherwise, this BN layer will share parameters that are learned from both the two modalities of data.

Formally, let o^1 and o^2 denote the above two candidate operations, respectively. In each BN layer l, we assign an architecture parameter $\alpha_{o^1}^l$ to the operation o^1 and the other architecture parameter $\alpha_{o^2}^l$ to the remaining operation o^2. When $\alpha_{o^1}^l = 1$ and $\alpha_{o^2}^l = 0$, it means that the BN layer l uses separate parameters. Otherwise, when $\alpha_{o^1}^l = 0$ and $\alpha_{o^2}^l = 1$, the BN layer l shares its parameters. In practice, instead of searching on such discrete candidate architectures, we relax the search space to make it can be optimized via gradient descent [25]. Concretely, we relax the binary architecture parameters $\alpha_{o^i}^l$ ($i \in \{1, 2\}$) to be continuous, and then obtain the probability of choosing the corresponding operation by computing a softmax over all architecture parameters:

$$p_{o^i}^l = \frac{\exp(\alpha_{o^i}^l)}{\exp(\alpha_{o^1}^l) + \exp(\alpha_{o^2}^l)}. \quad (1)$$

The larger the value of $p_{o^i}^l$, the more likely the BN layer l is to choose the operation o^i. The output of the BN layer l is a weighted sum of all possible operations:

$$x_{l+1} = p_{o^1}^l \cdot o_1(x_l) + p_{o^2}^l \cdot o_2(x_l), \quad (2)$$

where $o_i(x_l)$ denotes that the operation o_i is applied to the input x_l. In such a case, the search process is transformed
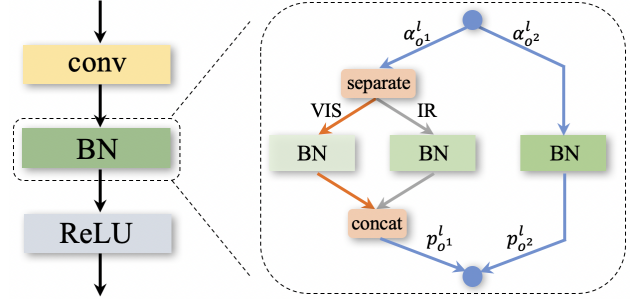

Figure 3. BN-oriented search space of CM-NAS. All BN layers in the backbone are reshaped as searchable units. Each BN layer has two candidate operations: employing two separate parameters for VIS and IR data respectively, or sharing its parameters for both the two modalities of data.

into the learning of a set of architecture parameters $\alpha = \{\alpha_{o^i}^l\}$. Furthermore, since the network weights w, such as the weights of convolution layers, also need to be learned, we are required to tackle the following bi-level optimization problem [25, 7, 43]:

$$\begin{aligned} &\min_\alpha \mathcal{L}_{\text{val}}(w^*, \alpha), \\ &s.t. \quad w^* = \arg\min_w \mathcal{L}_{\text{train}}(w, \alpha). \end{aligned} \quad (3)$$

The goal of Eq. (3) is to search architecture parameters α^* that minimize the validation loss $\mathcal{L}_{\text{val}}(w^*, \alpha^*)$, where the network weights w^* are obtained by minimizing the training loss $\mathcal{L}_{\text{train}}(w, \alpha)$. After training, for each BN layer l, we choose the operation with a larger probability and abandon the other one, yielding a discrete architecture. For instance, when $p_{o^1}^l > p_{o^2}^l$, we will choose the operation o_1, *i.e.* employing two separate parameters in the BN layer l. In addition, it is obvious that the training and the validation losses play critical roles in the search process, which are introduced in the following contents.

Objective Function. Up to now, the remaining part of our CM-NAS is to design an appropriate objective function to better guide the cross-modality search process. To begin with, Class-specific Maximum Mean Discrepancy (CMMD) [12, 19] is a commonly used measure for the modality discrepancy:

$$\mathcal{L}_{\text{CMMD}} = \frac{1}{C} \sum_{c=1}^{C} \left\| \frac{1}{m^c} \sum_{i=1}^{m^c} \psi(f_i^{c,vis}) - \frac{1}{n^c} \sum_{j=1}^{n^c} \psi(f_j^{c,ir}) \right\|_{\mathcal{H}}, \quad (4)$$

where $f^{c,vis}$ and $f^{c,ir}$ denote the features of VIS and IR images belonging to the c-th class, respectively. m^c and n^c are the numbers of the corresponding features. $\psi(\cdot)$ is a function that maps features into a universal Reproducing Kernel Hilbert Space (RKHS) [32]. We construct $\psi(\cdot)$ via a polynomial kernel $k(x, y) = \psi(x) \cdot \psi(y) = (x \cdot y)^2$.

Subsequently, given that the importance of feature correlations [29, 36], we also constrain the correlation consistency between the features of the VIS and IR modalities. Formally, let $\mathbf{F}_{vis}$ and $\mathbf{F}_{ir}$ be the set of embedding features of the VIS and the IR data respectively:

$$\begin{aligned} \mathbf{F}_{vis} &= \text{matrix}(f_1^{vis}, f_2^{vis}, ..., f_n^{vis}), \\ \mathbf{F}_{ir} &= \text{matrix}(f_1^{ir}, f_2^{ir}, ..., f_n^{ir}). \end{aligned} \quad (5)$$

In practice, we sample n VIS and n IR images for each time, because constraining the correlation consistency requires the same number of data [36]. Meanwhile, f_i^{vis} and f_i^{ir} ($i \in \{1, ..., n\}$) belong to the same identity. Correlation matrices $\tilde{\mathbf{G}}_{vis}$ and $\tilde{\mathbf{G}}_{ir}$ that reflect pairwise similarities among the features are given by:

$$\tilde{\mathbf{G}}_{vis} = \mathbf{F}_{vis} \cdot \mathbf{F}_{vis}^\top, \quad \tilde{\mathbf{G}}_{ir} = \mathbf{F}_{ir} \cdot \mathbf{F}_{ir}^\top. \quad (6)$$

Then, a row-wise L_2 normalization is applied on the two correlation matrices:

$$\mathbf{G}_{vis[i,:]} = \frac{\tilde{\mathbf{G}}_{vis[i,:]}}{||\tilde{\mathbf{G}}_{vis[i,:]}||_2}, \quad \mathbf{G}_{ir[i,:]} = \frac{\tilde{\mathbf{G}}_{ir[i,:]}}{||\tilde{\mathbf{G}}_{ir[i,:]}||_2}, \quad (7)$$

where the notation $[i,:]$ denotes the i-th row in a matrix. A correlation consistency loss $\mathcal{L}_{CC}$ is further developed to penalize the difference between $\mathbf{G}_{vis}$ and $\mathbf{G}_{ir}$:

$$\mathcal{L}_{CC} = \frac{1}{n^2} ||\mathbf{G}_{vis} - \mathbf{G}_{ir}||_F^2, \quad (8)$$

where $|| \cdot ||_F$ denotes a Frobenius norm.

The weighted sum of the above $\mathcal{L}_{CMMD}$ and $\mathcal{L}_{CC}$ is named as $\mathcal{L}_{C3MMD}$:

$$\mathcal{L}_{C3MMD} = \lambda_1 \mathcal{L}_{CMMD} + \lambda_2 \mathcal{L}_{CC}, \quad (9)$$

where λ_1 and λ_2 are trade-off parameters. Finally, in addition to $\mathcal{L}_{C3MMD}$, our objective function also includes a basic loss $\mathcal{L}_{basic}$ [40, 50] that consists of a classification loss $\mathcal{L}_{cls}$ and a triplet loss $\mathcal{L}_{triplet}$ to learn embedding features. As a result, the training loss $\mathcal{L}_{train}$ in Eq. (3) is the combination of $\mathcal{L}_{basic}$ and $\mathcal{L}_{C3MMD}$:

$$\mathcal{L}_{train} = \mathcal{L}_{basic} + \mathcal{L}_{C3MMD}. \quad (10)$$

The validation loss $\mathcal{L}_{val}$ in Eq. (3) has the same form as the training loss.

4. Experiment

4.1. Experimental Settings

Dataset. SYSU-MM01 [42] is a pioneer benchmark for the research of VI-ReID. This dataset consists of a total of 287,628 VIS images taken by 4 VIS cameras in the day-time, and 15,792 near-IR images taken by 2 near-IR cameras in the dark environment. These images are captured in both indoor and outdoor scenarios with abundant poses and viewpoints. The training set and the testing set have 395 and 96 person identities, respectively. Following [42], there are two testing models: *all-search* and *indoor-search*. For the former, the gallery set contains VIS images in both indoor and outdoor scenarios. For the latter, the gallery set merely contains VIS images in the indoor scenario. Besides, for both models, there are also two settings: *single-shot* and *multi-shot*. It means 1 or 10 VIS images of a person identity are randomly chosen to constitute the gallery set.

RegDB [28] is built by a dual camera acquisition system that includes a VIS camera and a thermal-IR camera. The two cameras are attached together to take photos at the same time, acquiring a total of 4,120 paired VIS-IR images from 412 person identities (each identity has 10 VIS images and 10 thermal-IR images). Following [48], 2,060 images from 206 person identities are randomly chosen as the training set and the remaining 2,060 images from 206 identities constitute the testing set. There are two evaluation settings: *Visible to Infrared* and *Infrared to Visible*. Take the former for example, it denotes leveraging the VIS images as the probe set and the IR images as the gallery set.

Evaluation Metrics. Following existing VI-ReID methods [39, 49], we adopt Cumulative Matching Characteristic (CMC) and mean Average Precision (mAP) as evaluation metrics. Moreover, the reported result on the SYSU-MM01 dataset is an average performance of 10 times repeated random probe/gallery splits [42], while that on the RegDB dataset is an average performance of 10 times random training/testing splits [48, 41].

Implementation Details. We adopt ResNet50 [14] pre-trained on ImageNet as the backbone. The input images are first padded with 10 and then randomly cropped to 256×128. Random horizontal flipping and random erasing are further imposed as data augmentation. Adam (β_1=0.5, β_2=0.999) is employed as an optimizer with 5e-4 weight decay. The initial learning rate is set to 0.01 and divided by 10 at the 40-th and the 70-th epochs. The training process is finished at the 120-th epoch. One training batch contains 8 identities and each identity has 4 VIS images as well as 4 IR images. These experimental settings mainly refer to previous works [26, 27]. The trade-off parameters λ_1 and λ_2 in Eq. (9) are set to 5.0 and 0.05, respectively. Following the standard process of NAS [25, 8, 16], there are two phases to train the network. In the first phase, all BN layers in the backbone are set to be searchable and Eq. (3) is utilized as the loss function, aiming to find the optimal neural architecture. Since the search process needs a validation set, we divide the original training set into a new training set and a validation set by a ratio of identities of 8:2. In the second phase, the architecture of the network is fixed to the

Table 1. Evaluations of CM-NAS on the SYSU-MM01 dataset under the *all-search* setting and the RegDB dataset. R1, R10 and R20 denote Rank-1, Rank-10 and Rank-20 accuracies (%), respectively. mAP denotes the mean average precision score (%). phase1 and phase2 correspond to the searching and the training phases of NAS, respectively. IN stands for Instance Normalization [37, 18].

Method	$\mathcal{L}_{\mathrm{C3MMD}}$	SYSU-MM01 (All-Search)								RegDB							
		Single-Shot				Multi-Shot				Visible to Infrared				Infrared to Visible			
		R1	R10	R20	mAP	R1	R10	R20	mAP	R1	R10	R20	mAP	R1	R10	R20	mAP
one-stream	✗	54.49	89.85	95.78	53.32	61.26	85.22	92.20	46.53	75.81	90.57	95.07	71.97	74.03	90.30	94.43	70.01
one-stream (IN)	✗	45.54	86.51	94.20	44.48	52.37	81.51	96.46	36.76	67.48	85.47	91.15	63.75	63.62	83.01	89.23	60.08
two-stream	✗	56.20	90.62	95.83	54.22	62.70	93.44	97.61	46.87	76.93	91.24	95.39	73.02	74.51	90.46	94.66	70.55
two-stream (IN)	✗	46.01	86.83	94.36	45.06	52.71	90.32	96.21	37.60	68.39	85.63	91.30	64.50	64.04	83.56	89.32	61.28
two-stream (BN)	✗	56.94	90.77	96.18	55.24	62.74	92.81	96.93	47.91	77.88	91.96	95.74	73.98	75.00	90.71	94.56	71.64
Auto-ReID [30]	✗	43.08	85.88	93.43	42.83	51.67	88.55	95.87	35.01	63.07	83.44	91.34	61.08	62.72	82.32	90.68	59.09
search	✗	59.56	91.43	96.24	56.79	66.12	94.21	97.90	49.89	78.77	92.94	96.29	76.05	77.55	92.44	95.87	74.86
one-stream	✓	56.18	90.80	96.05	54.47	63.36	93.15	97.53	47.60	77.42	91.46	95.32	74.57	75.45	90.67	94.66	71.84
one-stream (IN)	✓	46.93	87.10	94.59	45.83	54.43	90.23	97.11	38.51	70.96	88.34	93.08	66.35	68.24	86.18	91.92	62.76
two-stream	✓	57.86	90.72	96.21	55.97	64.28	93.66	97.97	48.41	78.03	91.79	95.37	74.96	75.52	90.77	94.68	72.04
two-stream (IN)	✓	47.38	88.66	95.74	46.78	55.18	92.65	97.75	39.57	71.35	88.62	93.35	66.94	68.94	86.30	92.05	63.72
two-stream (BN)	✓	57.94	91.70	96.76	56.30	65.05	93.58	97.88	49.33	78.48	92.65	96.27	75.53	75.87	91.10	94.64	73.10
Auto-ReID [30]	✓	44.71	86.10	93.91	44.58	53.41	88.85	96.31	36.63	65.52	83.80	91.47	62.70	64.64	83.13	90.94	60.56
search	phase1	59.55	91.83	96.93	57.62	66.72	94.52	97.63	50.73	79.71	93.25	96.38	76.65	78.20	92.70	96.04	75.28
search	phase2	59.95	91.74	96.80	57.72	66.93	94.55	97.96	51.02	82.36	94.16	97.27	78.83	81.20	93.63	96.54	77.14
search	phases1&2	**61.99**	**92.87**	**97.25**	**60.02**	**68.68**	**94.92**	**98.36**	**53.45**	**84.54**	**95.18**	**97.85**	**80.32**	**82.57**	**94.51**	**97.37**	**78.31**
search	w/o $\mathcal{L}_{\mathrm{CMMD}}$	60.77	91.73	96.46	58.74	67.56	94.69	98.14	51.93	82.18	94.41	97.69	78.92	81.53	93.98	96.86	77.45
search	w/o $\mathcal{L}_{\mathrm{CC}}$	60.27	91.96	96.77	58.45	67.25	94.50	98.09	51.39	83.75	94.86	97.84	79.72	81.87	94.29	97.09	77.45
search	$\mathcal{L}_{\mathrm{CMMD}} \rightarrow \mathcal{L}_{\mathrm{MMD}}$	60.83	92.14	96.79	58.92	67.99	94.76	97.90	52.37	82.79	95.06	97.74	79.25	81.68	94.06	96.91	77.58

best one and we retrain the network with Eq. (10) for the evaluation of VI-ReID. The training set in the second phase is the original one without splitting the validation set.

4.2. Experimental Analyses

Evaluation of the Search Space. We evaluate the search space proposed in Section 3.2 via a series of comparisons. Concretely, we only employ the basic loss $\mathcal{L}_{\mathrm{basic}}$ in Eq.(10) as the objective function. The compared six baselines are: (1) *one-stream* that shares the whole architecture for both VIS and IR modalities; (2) *one-stream (IN)* that replaces all BN in *one-stream* with IN, since the latter has the potential to reduce modality discrepancy [18]; (3) *two-stream* that separates blocks of stage1 and stage2 to learn modality-specific representations, and shares remaining blocks to learn modality-sharable representations; (4) *two-stream (IN)* that replaces all BN in *two-stream* with IN; (5) *two-stream (BN)* that only separates BN layers rather than all layers in the stage1 and stage2 blocks; (6) *Auto-ReID* [30] that searches the whole architecture rather than merely BN layers. To make a fair comparison, we constrain the searched architecture of Auto-ReID to have the same FLOPs as ResNet50.

The comparison results on the SYSU-MM01 and the RegDB datasets are reported Table 1, from which we have five observations. First, replacing BN with IN leads to performance degradation. This is because that although IN can minimize modality discrepancy by shifting style, it also brings the loss of discriminative information [18]. Second, the two-stream method performs better than the one-stream method, implying the necessity of separating blocks. Third, only separating the BN layers in the block is superior than separating the entire block, which indicates the significance of the BN layer in learning cross-modality representations.

The above two observations are consistent with those in Section 3.1, which motivate us to develop a BN-oriented search algorithm to automatically decide the separation of BN layers. Fourth, the performances of Auto-ReID are only comparable with the two stream method. This is because that the search space of Auto-ReID, which is specially designed for the single-modality task, is powerless to bridge the modality discrepancy in VI-ReID. Fifth, our search algorithm outperforms all the competitors. For instance, the search algorithm exceeds the BN-oriented two-stream method by 2.62% and 1.55% in terms of the Rank-1 and mAP on the SYSU-MM01 dataset under the *single-shot&all-search* setting. This demonstrates that our search algorithm can indeed find a more suitable neural architecture. Besides, compared with Auto-ReID [30], our method improves the Rank-1/mAP by 16.48%/13.96% on SYSU-MM01 and 15.70%/14.97% on RegDB. The unsatisfactory performances of Auto-ReID may be because that its cell-based search space is powerless to bridge the modality discrepancy. This further verifies the merit of our specially designed BN-oriented search space.

Evaluation of the Objective Function. In order to verify the effectiveness of $\mathcal{L}_{\mathrm{C3MMD}}$ in Eq. (9), we elaborately design several comparative experiments. Specifically, we add $\mathcal{L}_{\mathrm{C3MMD}}$ to the training of the aforementioned seven methods, including the six baselines as well as the search method. Since the search method has two training phases, there are three manners to add $\mathcal{L}_{\mathrm{C3MMD}}$: (1) only adding $\mathcal{L}_{\mathrm{C3MMD}}$ in the first phase, which means that the first phase is trained with both $\mathcal{L}_{\mathrm{basic}}$ and $\mathcal{L}_{\mathrm{C3MMD}}$ while the second phase is solely trained with $\mathcal{L}_{\mathrm{basic}}$; (2) only adding $\mathcal{L}_{\mathrm{C3MMD}}$ in the second phase; (3) adding $\mathcal{L}_{\mathrm{C3MMD}}$ in the both two phases. The results of the above methods are listed Table 1.

Regarding to the six baselines, compared with only employing $\mathcal{L}_{\text{basic}}$ (the first six rows in Table 1), it is observed that their performances are all improved after adding $\mathcal{L}_{\text{C3MMD}}$. For example, on the SYSU-MM01 dataset under the *single-shot&all-search* setting, the Rank-1 accuracy and the mAP score of the BN-oriented two-stream method increase by 1.0% and 1.06%, respectively. Such improvements adequately reveal the effectiveness of the proposed C3MMD loss. For the search method, we can see that adding $\mathcal{L}_{\text{C3MMD}}$ in the both two phases performs better than adding it in only one single phase.

Furthermore, we also investigate the effect of the two components in $\mathcal{L}_{\text{C3MMD}}$, *i.e.* $\mathcal{L}_{\text{CMMD}}$ and $\mathcal{L}_{\text{CC}}$. The results of the ablation study are shown in the bottom three rows of Table 1. We find that removing any component will lead to performance degradation, suggesting the importance of both $\mathcal{L}_{\text{CMMD}}$ and $\mathcal{L}_{\text{CC}}$. Furthermore, replacing $\mathcal{L}_{\text{CMMD}}$ with $\mathcal{L}_{\text{MMD}}$ [12] also results in inferior performances. This is because that compared with the latter, the former takes class labels into consideration and thus focuses on more specific modality distributions.

Parameter Analyses. We analyze the two trade-off parameters λ_1 and λ_2 in Eq. (9) on SYSU-MM01. The Rank-1 and the mAP results of CM-NAS with different λ_1 and λ_2 are exhibited in Fig. 5. We can see that our method is not sensitive to the parameters λ_1 and λ_2 in a large range. For instance, when λ_1 changes from 0.01 to 0.1, the Rank-1 accuracy only changes 0.67%. Moreover, the performance drops when the parameters are set too large, such as $\lambda_1 = 0.2$ or $\lambda_2 = 15$. The most suitable parameter setting is that $\lambda_1 = 0.05$ and $\lambda_2 = 5.0$.

Architecture Analyses. The searched architectures on SYSU-MM01 and RegDB are depicted in Fig. 4. First, it is observed that compared with low-level BN layers, high-level BN layers prefer to share parameters, which is in line with manual experience. Second, there are also some low-level BN layers choose sharing and some high-level BN layers choose separating. Compared with previous methods that directly separate the layers in the first one [47, 50] or five [52] blocks, our searched architecture is such complex and is hard to be manually designed, which suggests the

s1	s2			s3				s4						s5		
	$s2_1$	$s2_2$	$s2_3$	$s3_1$	$s3_2$	$s3_3$	$s3_4$	$s4_1$	$s4_2$	$s4_3$	$s4_4$	$s4_5$	$s4_6$	$s5_1$	$s5_2$	$s5_3$
0	0000	010	011	0001	000	111	101	1111	101	111	111	111	111	111	111	111
0	0000	000	110	1100	000	100	100	1111	111	111	111	111	111	111	111	111

Figure 4. Searched architectures on SYSU-MM01 (top) and RegDB (bottom). Each bit denotes the state of the corresponding BN layer, where '0' means separating and '1' means sharing. The symbol like 's2' denotes the architecture name of ResNet50, as shown in Fig. 2.
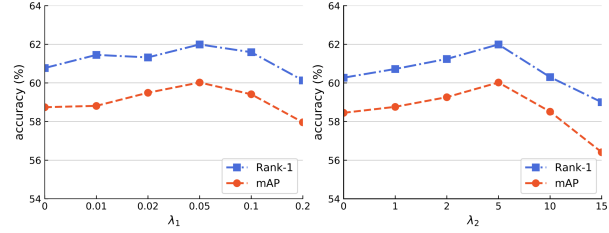

Figure 5. Parameter analyses of λ_1 and λ_2 in Eq. (9) on the SYSU-MM01 dataset under the *single-shot&all-search* setting. We first fix λ_2 to 5.0 and vary the value of λ_1 (left). Then, we fix λ_1 to 0.05 and vary the value of λ_2 (right). Rank-1 accuracies and mAP scores are reported for comparisons.

necessity of the search algorithm. Third, all BN layers in stage5 are shared, which is consistent with the observation in Fig. 1 (f). That is, separating the layers of stage5 always results in inferior performances.

4.3. Comparisons with State-of-the-Art Methods

We evaluate the proposed CM-NAS against current state-of-the-art VI-ReID methods on the SYSU-MM01 and the RegDB datasets. The compared methods include Zero-Pad [42], TONE [48], HCML [48], cmGAN [5], BDTR [49], eBDTR [49], HSME [13], D^{2}RL [41], MSR [9], AlignGAN [39], JSIA-ReID [40], Xmodal [21], MACE [47], cm-SSFT [26], DDAG [50] and HAT [51]. It should be noted that the multi-query setting of cm-SSFT uses multiple queries to constitute an auxiliary set to facilitate feature matching, which is infeasible in real scenarios and unfair to other methods. Therefore, we only report the results of cm-SSFT under the single-query setting [26].

Comparisons on SYSU-MM01. Table 2 displays the comparison results with state-of-the-art VI-ReID methods on the SYSU-MM01 dataset. From the perspective of evaluation settings, *i.e.* *all-search/indoor-search* and *single-shot/multi-shot*, we have two observations. First, the results of all the methods under the *indoor-search* setting are better than those under the *all-search* setting. This is because the images in the *indoor-search* setting only contain relatively brief indoor scenarios, while those in the *all-search* setting have more complex in-the-wild scenarios. Second, for a same method, the results under the *multi-shot* setting are better than those under the *single-shot* setting in the aspect of Rank-1 accuracies, while the phenomenon is opposite in the aspect of mAP scores. This is due to the fact that each person identity contains 10 gallery images in the *multi-shot* setting and only has 1 gallery image in the *single-shot* setting. As a result, for *multi-shot*, it is prone to match one right sample according to the ranking of similarity scores, but it is difficult to match all right samples. The following analyses are based on the *single-shot&all-search* setting, since it is more challenging as mentioned in [39, 13].

Table 2. Comparisons with state-of-the-art methods on the SYSU-MM01 dataset. cm-SSFT* denotes that we report the results of cm-SSFT under the single-query setting [26] for fair comparisons with other methods. CM-NAS† means first searching a neural architecture on the RegDB dataset and then training the searched neural network on the SYSU-MM01 dataset with the loss function Eq. (10).

Method	Venue	All-Search								Indoor-Search							
		Single-Shot				Multi-Shot				Single-Shot				Multi-Shot			
		R1	R10	R20	mAP	R1	R10	R20	mAP	R1	R10	R20	mAP	R1	R10	R20	mAP
Zero-Pad [42]	ICCV-17	14.80	54.12	71.33	15.95	19.13	61.40	78.41	10.89	20.58	68.38	85.79	26.92	24.43	75.86	91.32	18.64
TONE [48]	AAAI-18	12.52	50.72	68.60	14.42	-	-	-	-	20.82	68.86	84.46	26.38	-	-	-	-
HCML [48]	AAAI-18	14.32	53.16	69.17	16.16	-	-	-	-	24.52	73.25	86.73	30.08	-	-	-	-
cmGAN [5]	IJCAI-18	26.97	67.51	80.56	27.80	31.49	72.74	85.01	22.27	31.63	77.23	89.18	42.19	37.00	80.94	92.11	32.76
BDTR [49]	IJCAI-18	27.32	66.96	81.07	27.32	-	-	-	-	31.92	77.18	89.28	41.86	-	-	-	-
eBDTR [49]	TIFS-19	27.82	67.34	81.34	28.42	-	-	-	-	32.46	77.42	89.62	42.46	-	-	-	-
HSME [13]	AAAI-19	20.68	62.74	77.95	23.12	-	-	-	-	-	-	-	-	-	-	-	-
D^{2}RL [41]	CVPR-19	28.9	70.6	82.4	29.2	-	-	-	-	-	-	-	-	-	-	-	-
MSR [9]	TIP-19	37.35	83.40	93.34	38.11	43.86	86.94	95.68	30.48	39.64	89.29	97.66	50.88	46.56	93.57	98.80	40.08
AlignGAN [39]	ICCV-19	42.4	85.0	93.7	40.7	51.5	89.4	95.7	33.9	45.9	87.6	94.4	54.3	57.1	92.7	97.4	45.3
JSIA-ReID [40]	AAAI-20	38.1	80.7	89.9	36.9	45.1	85.7	93.8	29.5	43.8	86.2	94.2	52.9	52.7	91.1	96.4	42.7
Xmodal [21]	AAAI-20	49.92	89.79	95.96	50.73	-	-	-	-	-	-	-	-	-	-	-	-
cm-SSFT* [26]	CVPR-20	47.7	-	-	54.1	57.4	-	-	**59.1**	-	-	-	-	-	-	-	-
MACE [47]	TIP-20	51.64	87.25	94.44	50.11	-	-	-	-	57.35	93.02	97.47	64.79	-	-	-	-
DDAG [50]	ECCV-20	54.75	90.39	95.81	53.02	-	-	-	-	61.02	94.06	98.41	67.98	-	-	-	-
HAT [51]	TIFS-20	55.29	92.14	97.36	53.89	-	-	-	-	62.10	95.75	99.20	69.37	-	-	-	-
CM-NAS	-	**61.99**	**92.87**	**97.25**	**60.02**	**68.68**	**94.92**	**98.36**	53.45	**67.01**	**97.02**	**99.32**	**72.95**	**76.48**	**98.68**	**99.91**	**65.11**
CM-NAS†	-	60.42	91.86	96.76	58.62	67.87	94.49	98.11	52.07	66.74	96.14	99.27	72.37	75.69	97.93	99.76	64.27

Table 3. Comparisons with state-of-the-art methods on the RegDB dataset. CM-NAS† means first searching a neural architecture on the SYSU-MM01 dataset and then training the searched neural network on the RegDB dataset.

Method	Visible to Infrared				Infrared to Visible			
	R1	R10	R20	mAP	R1	R10	R20	mAP
Zero-Pad [42]	17.74	34.21	44.35	18.90	16.63	34.68	44.25	17.82
HCML [48]	24.44	47.53	56.78	20.08	21.70	45.02	55.58	22.24
BDTR [49]	33.56	58.61	67.43	32.76	32.92	58.46	68.43	31.96
eBDTR [49]	34.62	58.96	68.72	33.46	34.21	58.74	68.64	32.49
HSME [13]	50.85	73.36	81.66	47.00	50.15	72.40	81.07	46.16
D^{2}RL [41]	43.4	66.1	76.3	44.1	-	-	-	-
MAC [47]	36.43	62.36	71.63	37.03	36.20	61.68	70.99	36.63
MSR [9]	48.43	70.32	79.95	48.67	-	-	-	-
AlignGAN [39]	57.9	-	-	53.6	56.3	-	-	53.4
JSIA-ReID [40]	48.5	-	-	49.3	48.1	-	-	48.9
Xmodal [21]	62.21	83.13	91.72	60.18	-	-	-	-
cm-SSFT* [26]	65.4	-	-	65.6	63.8	-	-	64.2
MACE [47]	72.37	88.40	93.59	69.09	72.12	88.07	93.07	68.57
DDAG [50]	69.34	86.19	91.49	63.46	68.06	85.15	90.31	61.80
HAT [51]	71.83	87.16	92.16	67.56	70.02	86.45	91.61	66.30
CM-NAS	**84.54**	**95.18**	**97.85**	**80.32**	**82.57**	**94.51**	**97.37**	**78.31**
CM-NAS†	82.68	94.79	97.58	78.91	81.24	93.85	96.65	77.16

From the perspective of methods, our CM-NAS outperforms all the competitors by a large margin. For instance, compared with the state-of-the-art HAT, we improve the Rank-1 accuracy and the mAP score by **6.70%** and **6.13%**, respectively. Meanwhile, it is worth noting that the image size of the input of HAT is 288×144, while that of our method is 256×128. The significant improvements with a smaller image size fully reveal the superiority of our method. Furthermore, we also conduct a cross-dataset experiment to evaluate the generalization ability of the searched neural architecture. We first search a neural architecture on RegDB, and then train the searched neural architecture on SYSU-MM01 with the loss function Eq. (10). Such a cross-dataset method is denoted as CM-NAS† in Table 2. It is observed that CM-NAS† still significantly surpasses all counterparts, unfolding the great generalization ability of the searched cross-modality architecture.

Comparisons on RegDB. As listed in in Table 3, it can be seen that our CM-NAS has distinct advantages over state-of-the-art counterparts on RegDB. Under the *Visible to Infrared* setting, CM-NAS exceeds the state-of-the-art MACE by **12.17%** and **11.23%** in terms of the Rank-1 accuracy and the mAP score, respectively. When switching to the *Infrared to Visible* setting, compared with MACE, CM-NAS still improves the two indicators by **10.45%** and **9.74%**, respectively. Besides, we also investigate the aforementioned cross-dataset method CM-NAS†, in which the neural architecture is first searched on SYSU-MM01 and then trained on RegDB. Table 3 shows that CM-NAS† is always superior to the compared methods, which further verifies the generalization ability of the searched architecture.

5. Conclusion

This paper has presented a novel CM-NAS to tackle the challenging VI-ReID. We systematically investigate the manually designed neural architectures and find that appropriately separating BN layers can yield better performances. This motivates us to develop a BN-oriented NAS algorithm that has the ability to automatically decide the separation of BN layers, searching the optimal architecture. Extensive experiments on two popular datasets demonstrate the superiority of CM-NAS. We expect this simple yet effective method will serve as a solid foundation to facilitate future research in VI-ReID.

Acknowledgment

References

[1] Song Bai, Xiang Bai, and Qi Tian. Scalable person re-identification on supervised smoothed manifold. In *CVPR*, 2017.

[2] Bowen Baker, Otkrist Gupta, Nikhil Naik, and Ramesh Raskar. Designing neural network architectures using reinforcement learning. In *ICLR*, 2017.

[3] Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware. In *ICLR*, 2019.

[4] Xin Chen, Lingxi Xie, Jun Wu, and Qi Tian. Progressive differentiable architecture search: Bridging the depth gap between search and evaluation. In *ICCV*, 2019.

[5] Pingyang Dai, Rongrong Ji, Haibin Wang, Qiong Wu, and Yuyu Huang. Cross-modality person re-identification with generative adversarial training. In *IJCAI*, 2018.

[6] Xiaoliang Dai, Peizhao Zhang, Bichen Wu, Hongxu Yin, Fei Sun, Yanghan Wang, Marat Dukhan, Yunqing Hu, Yiming Wu, Yangqing Jia, Peter Vajda, Matt Uyttendaele, and Niraj K. Jha. Chamnet: Towards efficient network design through platform-aware model adaptation. In *CVPR*, 2019.

[7] Xuanyi Dong and Yi Yang. Searching for a robust neural architecture in four gpu hours. In *CVPR*, 2019.

[8] Jiemin Fang, Yuzhu Sun, Qian Zhang, Yuan Li, Wenyu Liu, and Xinggang Wang. Densely connected search space for more flexible neural architecture search. In *CVPR*, 2020.

[9] Zhanxiang Feng, Jianhuang Lai, and Xiaohua Xie. Learning modality-specific representations for visible-infrared person re-identification. *TIP*, 2019.

[10] Chen Gao, Si Liu, Defa Zhu, Quan Liu, Jie Cao, Haoqian He, Ran He, and Shuicheng Yan. Interactgan: Learning to generate human-object interaction. In *ACM MM*, 2020.

[11] Shaogang Gong, Marco Cristani, Shuicheng Yan, and Chen Change Loy. Person re-identification. 2016.

[12] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. In *NeurIPS*, 2007.

[13] Yi Hao, Nannan Wang, Jie Li, and Xinbo Gao. Hsme: Hypersphere manifold embedding for visible thermal person re-identification. In *AAAI*, 2019.

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

[15] Lingxiao He, Jian Liang, Haiqing Li, and Zhenan Sun. Deep spatial feature reconstruction for partial person re-identification: Alignment-free approach. In *CVPR*, 2018.

[16] Yibo Hu, Xiang Wu, and Ran He. Tf-nas: Rethinking three search freedoms of latency-constrained differentiable neural architecture search. In *ECCV*, 2020.

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*, 2015.

[18] Xin Jin, Cuiling Lan, Wenjun Zeng, Zhibo Chen, and Li Zhang. Style normalization and restitution for generalizable person re-identification. In *CVPR*, 2020.

[19] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*, 2019.

[20] Qingming Leng, Mang Ye, and Qi Tian. A survey of open-world person re-identification. *TCSVT*, 2019.

[21] Diangang Li, Xing Wei, Xiaopeng Hong, and Yihong Gong. Infrared-visible cross-modal person re-identification with an x modality. In *AAAI*, 2020.

[22] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, 2015.

[23] Shengcai Liao and Stan Z Li. Efficient psd constrained asymmetric metric learning for person re-identification. In *ICCV*, 2015.

[24] Haijun Liu, Jian Cheng, Wen Wang, Yanzhou Su, and Haiwei Bai. Enhancing the discriminative feature learning for visible-thermal cross-modality person re-identification. *Neurocomputing*, 2020.

[25] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. In *ICLR*, 2019.

[26] Yan Lu, Yue Wu, Bin Liu, Tianzhu Zhang, Baopu Li, Qi Chu, and Nenghai Yu. Cross-modality person re-identification with shared-specific feature transfer. In *CVPR*, 2020.

[27] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *CVPR Workshops*, 2019.

[28] Dat Tien Nguyen, Hyung Gil Hong, Ki Wan Kim, and Kang Ryoung Park. Person recognition system based on a combination of body images from visible light and thermal cameras. *Sensors*, 2017.

[29] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *ICCV*, 2019.

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

[31] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *AAAI*, 2019.

[32] Ingo Steinwart. On the influence of the kernel on the consistency of support vector machines. *JMLR*, 2001.

[33] 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.

[34] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *CVPR*, 2019.

[35] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019.

[36] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *ICCV*, 2019.

[37] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv:1607.08022*, 2016.

[38] 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.

[39] Guan-An Wang, Tianzhu Zhang, Jian Cheng, Si Liu, Yang Yang, and Zengguang Hou. Rgb-infrared cross-modality person re-identification via joint pixel and feature alignment. In *ICCV*, 2019.

[40] Guan-An Wang, Tianzhu Zhang, Yang Yang, Jian Cheng, Jianlong Chang, Xu Liang, and Zengguang Hou. Cross-modality paired-images generation for rgb-infrared person re-identification. In *AAAI*, 2020.

[41] Zhixiang Wang, Zheng Wang, Yinqiang Zheng, Yung-Yu Chuang, and Shin'ichi Satoh. Learning to reduce dual-level discrepancy for infrared-visible person re-identification. In *CVPR*, 2019.

[42] Ancong Wu, Wei-Shi Zheng, Hong-Xing Yu, Shaogang Gong, and Jianhuang Lai. Rgb-infrared cross-modality person re-identification. In *ICCV*, 2017.

[43] Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In *CVPR*, 2019.

[44] Xiang Wu, Ran He, Yibo Hu, and Zhenan Sun. Learning an evolutionary embedding via massive knowledge distillation. *IJCV*, 2020.

[45] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *CVPR*, 2020.

[46] Yuhui Xu, Lingxi Xie, Xiaopeng Zhang, Xin Chen, Guo-Jun Qi, Qi Tian, and Hongkai Xiong. Pc-darts: Partial channel connections for memory-efficient differentiable architecture search. In *ICLR*, 2020.

[47] Mang Ye, Xiangyuan Lan, Qingming Leng, and Jianbing Shen. Cross-modality person re-identification via modality-aware collaborative ensemble learning. *TIP*, 2020.

[48] Mang Ye, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Hierarchical discriminative learning for visible thermal person re-identification. In *AAAI*, 2018.

[49] Mang Ye, Xiangyuan Lan, Zheng Wang, and Pong C Yuen. Bi-directional center-constrained top-ranking for visible thermal person re-identification. *TIFS*, 2019.

[50] Mang Ye, Jianbing Shen, David J Crandall, Ling Shao, and Jiebo Luo. Dynamic dual-attentive aggregation learning for visible-infrared person re-identification. In *ECCV*, 2020.

[51] Mang Ye, Jianbing Shen, and Ling Shao. Visible-infrared person re-identification via homogeneous augmented tri-modal learning. *TIFS*, 2020.

[52] Mang Ye, Zheng Wang, Xiangyuan Lan, and Pong C Yuen. Visible thermal person re-identification via dual-constrained top-ranking. In *IJCAI*, 2018.

[53] Hong-Xing Yu, Ancong Wu, and Wei-Shi Zheng. Unsupervised person re-identification by deep asymmetric metric embedding. *TPAMI*, 2018.

[54] Yunpeng Zhai, Shijian Lu, Qixiang Ye, Xuebo Shan, Jie Chen, Rongrong Ji, and Yonghong Tian. Ad-cluster: Aug-

mented discriminative clustering for domain adaptive person re-identification. In *CVPR*, 2020.

[55] Hongwen Zhang, Jie Cao, Guo Lu, Wanli Ouyang, and Zhenan Sun. Danet: Decompose-and-aggregate network for 3d human shape and pose estimation. In *ACM MM*, 2019.

[56] Shizhen Zhao, Changxin Gao, Jun Zhang, Hao Cheng, Chuchu Han, Xinyang Jiang, Xiaowei Guo, Wei-Shi Zheng, Nong Sang, and Xing Sun. Do not disturb me: Person re-identification under the interference of other pedestrians. In *ECCV*, 2020.

[57] 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.

[58] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015.

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

[60] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Reidentification by relative distance comparison. *TPAMI*, 2012.

[61] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, 2019.

[62] Zhedong Zheng, Liang Zheng, and Yi Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *ICCV*, 2017.

[63] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *CVPR*, 2018.