algorithms

MDPI

Article

KDAS-ReID: Architecture Search for Person Re-Identification via Distilled Knowledge with Dynamic Temperature

Zhou Lei [1,2], Kangkang Yang [1,2], Kai Jiang [1,2] and Shengbo Chen [1,2,*]

[1] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China; leiz@shu.edu.cn (Z.L.); ykk_9678@shu.edu.cn (K.Y.); sherlock_ss@shu.edu.cn (K.J.)
[2] Shanghai Key Laboratory of Computer Software Testing and Evaluating, Shanghai 201112, China
* Correspondence: schen@shu.edu.cn; Tel.: +86-021-66135378

Abstract: Person re-Identification(Re-ID) based on deep convolutional neural networks (CNNs) achieves remarkable success with its fast speed. However, prevailing Re-ID models are usually built upon backbones that manually design for classification. In order to automatically design an effective Re-ID architecture, we propose a pedestrian re-identification algorithm based on knowledge distillation, called KDAS-ReID. When the knowledge of the teacher model is transferred to the student model, the importance of knowledge in the teacher model will gradually decrease with the improvement of the performance of the student model. Therefore, instead of applying the distillation loss function directly, we consider using dynamic temperatures during the search stage and training stage. Specifically, we start searching and training at a high temperature and gradually reduce the temperature to 1 so that the student model can better learn from the teacher model through soft targets. Extensive experiments demonstrate that KDAS-ReID performs not only better than other state-of-the-art Re-ID models on three benchmarks, but also better than the teacher model based on the ResNet-50 backbone.

Keywords: architecture search; knowledge distillation; person re-identification; dynamic temperature; convolutional network

Citation: Lei, Z.; Yang, K.; Jiang, K.; Cheng, S. KDAS-ReID: Architecture Search for Person Re-Identification via Distilled Knowledge with Dynamic Temperature. *Algorithms* **2021**, *14*, 137. https://doi.org/10.3390/a14050137

Academic Editor: Mircea-Bogdan Radac

Received: 27 March 2021
Accepted: 23 April 2021
Published: 26 April 2021

1. Introduction

With the rapid development of Deep Learning, person re-Identification based on deep convolutional neural networks has achieved remarkable success in the last few years. The goal of Re-ID is to retrieve images of person-of-interest. Most researchers of the Re-ID community focus on modifying some pre-trained convolutional neural network backbones such as ResNet [1] and its variants in order to improve the performance of Re-ID. The Re-ID is a challenging task result from the presence of different viewpoints, varying low-image resolutions, illumination changes, unconstrained poses, occlusions, heterogeneous modalities, etc. Ref. [2]. Prevailing CNN models for Re-ID has greatly improved the rank-1 accuracy on the Market-1501 [3] and other benchmarks due to the excellent representation capability of features. However, manually designing an efficient convolutional neural network backbone usually needs a group of human experts and costs a lot of time. Another problem is that these handcraft models were oriented, designed for image classification tasks, and the Re-ID task is mainly for retrieval, so there is still a lot of room for improvement in Re-ID tasks.

As an important branch of Automated Machine Learning (AutoML), Neural Architecture Search (NAS) expects to design an efficient architecture for Re-ID tasks automatically, rather than designing the convolutional neural network backbone for Re-ID manually. With the growth of NAS, the time for a well-performing architecture of a particular task can be reduced from months to hours. Re-ID is essentially a retrieval task, but current NAS algorithms are merely designed for classification.

Manually designing a CNN network that is suitable for Re-ID task is not only time-consuming but also heavily relies on professional knowledge. In light of this, we propose to automatically search for a CNN architecture that is specifically suitable for the Re-ID task. Due to the fact that current NAS algorithms are only for classification purposes, while Re-ID is essentially a retrieval task. A retrieval based search algorithm over a specifically designed Re-ID search space is necessary since Re-ID is different from classification. Auto-ReID [4] took the first step to automate the Re-ID model design. They have removed some redundant operations and designed a new operation called a part-aware module that can retain body structural information of humans to improve the performance of the searched Re-ID model. Another crucial contribution is that they introduced retrieval loss in order to fit into the Re-ID task. Different from Auto-ReID, we proposed a novel deep-based Re-ID algorithm called KDAS-ReID that can automate the design of Re-ID architectures based on knowledge distillation.

Figure 1 showed that knowledge distillation aims at transferring knowledge from a pre-trained teacher model to a student model. Hinton et al. [5] defined knowledge distillation as training a smaller and faster student model to approach the teacher's outputs after softmax. Motivated by this, we considered an approach to apply the NAS algorithm in Re-ID tasks and make the final architecture that can be easily deployed through taking a pre-trained network with high performance on Re-ID as the teacher model during the search stage. Our contributions can be summarized as follows:

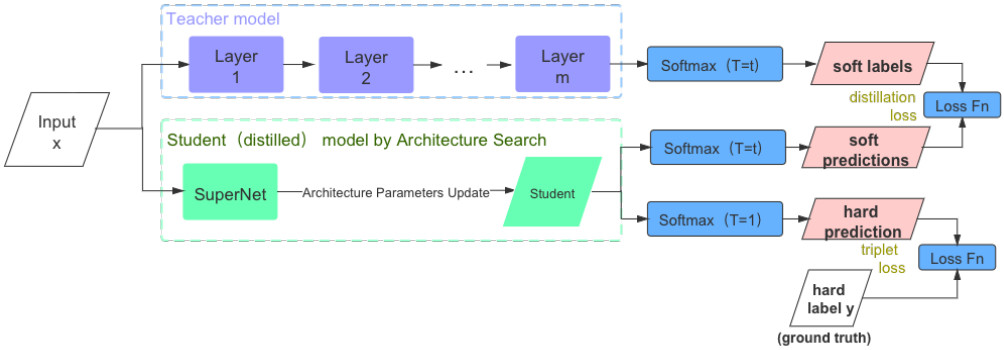

Figure 1. Knowledge distillation, which aims to transfer knowledge from the pre-trained teacher model to the student model by training a smaller, faster student model and making its output close to that of the teacher model after sofmax.

- Person re-identification is essentially a retrieval problem that aims at retrieving images of persons-of-interest. Previous Re-ID convolutional neural networks are usually built upon backbones for image classification, such as ResNet [1], Inception [6], and VGG [7]. In this paper, the NAS algorithm is recommended based on knowledge distillation that aims at designing a convolutional neural network backbone that performed well in the Re-ID task. More concretely, we first trained a teacher model on Market-1501 which built upon ResNet backbones and pre-trained on the ImageNet. Cross-entropy loss is replaced by distillation loss, which is computed by the soft targets and the outputs of the candidates model after softmax;
- With the success of deep learning, the demand for architecture engineering is also growing, and an increasing number of researchers focus on CNN architecture because the performance of CNNs depend on their architecture. Because of the features of NAS, we considered a distillation loss with dynamic temperature in hopes of breaking the limit of the teacher model. We started searching and training with a high temperature and gradually reduced to 1 to control the importance of the knowledge of teacher model;

- Extensive experiments showed that our work achieves a competitive accuracy compared with the teacher model, while the searched model has less than 50% parameters of the teacher model. In addition, rank-1 with 94.6% accuracy is achieved on market-1501 by using the search space of Auto-ReID.

2. Related Works

2.1. Person Re-ID

The great success of deep learning has brought tremendous progress in person Re-ID [8–11]. Xiao et al. [8] presented a pipeline that can learn generic feature representation from multiple datasets, and proposed Domain Guided Dropout to improve the performance of CNN. Chen et al. [9] designed a quadruplet loss. Compared with the triplet loss, this loss allows deep CNN to capture a larger inter-class variation and a smaller intra-class variation so that the model can have a better generalization ability. By directly integrating human joint information into a convolutional neural network, Saquib et al. [10] enable deep CNN to learn to pose sensitive representations. Sun et al. [11] leveraged a part-based CNN model and adaptive pooling method to learn part-level features. However, these deep-based Re-ID algorithms [8,10–13] rely heavily on the CNN backbones designed for classification problems, such as VGG [7], Inception [6], and ResNet [1]. However, these CNN backbones are all experimented on with classified datasets, which may be inconsistent with the Re-ID and limit the performance of the Re-ID algorithm. In our work, we not only inherit the advantages of previous Re-ID algorithms, but also overcome their disadvantages. KDAS-ReID will automatically search the CNN architecture suitable for Re-ID in its search space.

2.2. Neural Architecture Search

Although our work is based on the latest NAS research [14–20], most of these NAS are used for classification tasks, and a high-performance model that is suitable for Re-ID is required. Most NAS methods [15–18] search for CNN tasks in a small range, and then transfer the searched structures to a large specified target task. Zoph et al. [16,17] applied reinforcement learning to the search CNN architecture, but the search required hundreds of GPU days. Suganuma et al. [14] attempted to automatically construct CNN architectures for an image classification task based on Cartesian genetic programming (CGP). Real et al. [19] suggested associating each genotype with an age and bias the tournament selection to choose the younger CNN candidates. Brock et al. [18] and Bender et al. [20] tried to use a one-stop architecture search method. Lorenzo et al. [21] proposed a fully automatic method with the goal of optimizing deep neural network (DNN) topologies through memetic evolution. Xiao et al. [22] proposed to take advantage of a variable length genetic algorithm (GA) to systematically and automatically tune the hyperparameters of a CNN to improve its performance. In [23], particle swarm optimization (PSO) was used to select hyper-parameters. Liu et al. [15] relaxed the search space to be continuous, so that the architecture can be optimized with respect to its validation set performance by gradient descent. Furthermore, these NAS algorithms are primarily used for classification tasks, while the Re-ID task is primarily for retrieval, and the goals of retrieval and classification tasks are fundamentally different. Therefore, in order to apply NAS algorithm well in the Re-ID task, we took the pre-trained network with high performance on Re-ID as the teacher model in the search stage.

2.3. Knowledge Distillation

Transferring knowledge from a huge, cumbersome model to a smaller and simpler model without losing too much generalization ability has always been one of the classic problems studied by scholars in recent years. Knowledge distillation using neural networks was originally proposed by Hinton et al. [5], which aims at improving the training of student models by applying knowledge acquired from a powerful teacher model. Ba and Caruana et al. [24] demonstrated that shallow feed-forward nets can learn the complex

functions that were previously learned by deep nets and achieve accuracies that were only achievable with deep models. Romero et al. [25] guided the students' training process by introducing intermediate hints from the teacher's hidden layer, compressing wide and deep networks into thin and deep networks. In [26], the author proposes to use activation-based and gradient-based spatial attention maps to transfer attention as a method of knowledge transfer from one network to another. Yim et al. [27] used the extracted two-layer feature map to generate the flow of the solution procedure (FSP) matrix and trained the student DNN to make the FSP matrix similar to the teacher DNN. Sau and Balasubramanian [28] recommended to use a noise-based regularizer when training the student model to learn from the teacher model, which can significantly enhance the performance of the student network. Since knowledge distillation is characterized by transferring complex models to simple ones, the NAS algorithm can be applied in Re-ID task by knowledge distillation. At the same time, due to the characteristics of NAS, it is considered that the distillation loss at dynamic temperature breaks the constraint of the teacher model. A searching and training is started at a high temperature, and the temperature will gradually drop to 1 in order to control the importance of teacher model knowledge.

3. Methods

In this section, an approach that searches for a Re-ID Model based on distilled knowledge will be demonstrated. We will first introduce the preliminary background of DARTS in Section 3.1. Then, a novel NAS algorithm for Re-ID based on the knowledge distillation will be introduced in Section 3.2. Lastly, in Section 3.3, a comprehensive elaboration on how the distillation loss works and the way to break the limit of the teacher model will be provided.

3.1. Preliminaries of DARTS

Early existing architecture search algorithms are computationally like NAS [16] based on reinforcement learning that obtains a state-of-the-art architecture for CIFAR-10 required 2000 GPU days. In order to speed up the search process, DARTS [15] has proposed relaxing the discrete search space to a continuous search space so that the architecture parameters can be optimized by gradient descent.

Each node $x^{(i,j)}$ is a latent representation such as a feature map in convolutional networks and there are several edges (i, j) between every two nodes during the search phase. Each edge is an operation $o^{(i,j)}$ of the search space O (e.g., 3×3 dilated convolution, 3×3 max pooling) and a directed edge represent a transformation from $x^{(i)}$ to $x^{(j)}$ by applying operation $o^{(i,j)}$. Thus, each intermediate node is computed as:

$$x^{(j)} = \sum_{i<j} o^{(i,j)}(x^{(i)}) \tag{1}$$

The target of DARTS is to find the optimal operation (including a special zero operation that indicates a lack of connection between two nodes) between every two intermediate nodes, and then derive the final architecture that performs well in target datasets. Specifically, as showed in Figure 2, DARTS first initialize a supernet with random network weights and unknown operations on the edges. Then, there is continuous relaxation of the search space by placing a mixture of candidate operations on each edge to get a continuous search space that can optimize validation loss by using gradient descent. The process of making the search space continuous can be described as:

$$\bar{x}^{(i,j)} = \sum_{o \in O} \frac{exp(\alpha_o^{(i,j)})}{\sum_{o' \in O} exp(\alpha_{o'}^{(i,j)})} o(x) \tag{2}$$

where the α indicates the architecture parameters.

Let L_t and L_v denote the training loss and validation loss, after relaxing the categorical choice of a particular operation to a softmax over all possible operations, DARTS aims at

jointly learning architecture α and the network weights w by finding α^* that minimizes the validation loss. The goal of DARTS is to reduce such a bi-level optimization problem:

$$\min_{\alpha} \; L_v(w^*(\alpha), \alpha) \tag{3}$$

$$w^*(\alpha) = argmin_w \, L_t(w, \alpha^*) \tag{4}$$

where the w^* is obtained by minimizing the training loss.

Lastly, the learned α will be used to derive the final architecture.

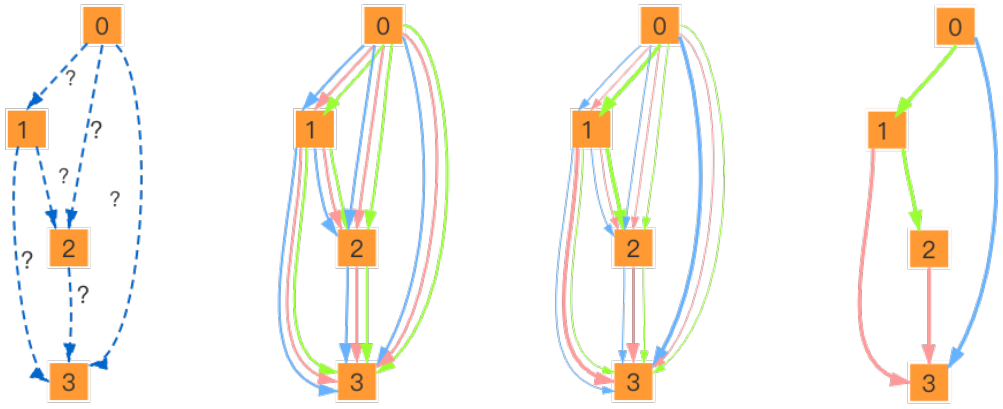

Figure 2. An overview of DARTS. First, initialize a supernet with random network weights w and unknown operations on the edges. Then, there is continuous relaxation of the search space by placing a mixture of candidate operations and jointly learning architecture α and the network weights w by finding α^* that minimizes the validation loss. Lastly, derive the final architecture based on the learned α.

3.2. ReID Search Algorithm with Knowledge Distillation

3.2.1. Search Space

Recent research suggested that the knowledge of a convolutional network not only relies on the network parameters but also depends on network architecture. The cell-based architecture search algorithms aim at designing the topological structure of a cell instead of the whole network structure. Therefore, the design of search space which represents a supernet determined the upper limit of performance. Auto-ReID [4] improves the search space based on DARTS to make the final architecture suitable to Re-ID tasks and our work draws on this search space:

- part-aware module
- 3×3 max pooling
- 3×3 average pooling
- 3×3 depth-wise separable convolution
- 3×3 dilated convolution
- zero operation
- identity mapping.

3.2.2. Network Structure

We use the ResNet-50 [1] which manually designs for classification as the teacher model in order to automatically design a convolutional network backbone that performs well in the Re-ID task. We also use the macrostructure of the teacher model for our student model backbone, where each residual layer is replaced by a cell. At the same time, we aim at searching a student network with much fewer parameters to compare with the teacher network by reducing the number of layers. The student backbone was stacked by eight cells (including *Normal Cell* and *Reduction Cell*), and we set *Reduction Cell* in the

3rd, 5th, and 7th place of cells to extend the dimension of features extracted by pre-cell. A convolutional operation is also adopted to process inputs before backbone, and, after backbone, an embedding layer and a classification model is used to transfer the features f extracted by the backbone into the logits l.

3.2.3. Search Algorithm

As our overall algorithm showed in Algorithm 1 and Figure 1, we adopted the differentiable search strategy of DARTS so that a mixed operation $\hat{o}^{(i,j)}$ parameterized by $\alpha^{(i,j)}$ for each edge (i, j) is created. For data preparation, $\mathbb{D}_T$ is split into the search training set $\mathbb{D}_{train}$ and the search validation set $\mathbb{D}_{val}$. During the iterative procedure, we first update the temperature T of the distillation loss in Equation (7) according to the current epoch. Secondly, the class-balance data sampler is used to get batch data from $\mathbb{D}_{train}$ at each epoch for network weights optimization. The objective loss in Equation (9) composed by the distillation loss in Equation (7) and the triplet loss in Equation (8). After network weights' optimization, we use the class-balance data sampler to get batch data from $\mathbb{D}_{val}$ and update the architecture parameters α via the objective loss. Lastly, we obtain the final architecture derived from the chosen operations, and then we train and evaluate the final architecture on the $\mathbb{D}_T$ and the evaluation set $\mathbb{D}_E$ in a standard reID strategy.

Algorithm 1: The KDAS-ReID Algorithm

Input: the SuperNet weights w; the architecture parameter α; the training set $\mathbb{D}_T$ and the evaluation set $\mathbb{D}_E$; a class-balance data sampler;

Create a mixed operation $\hat{o}^{(i,j)}$ parameterized by $\alpha^{(i,j)}$ for each edge (i, j);
Split $\mathbb{D}_T$ into the search training set $\mathbb{D}_{train}$ and the search validation set $\mathbb{D}_{val}$;
while *not terminated* **do**
    Update the temperature T of the distillation loss in Equation (7).;
    Use the sampler to get batch data from $\mathbb{D}_{train}$;
    Update the network weights w via the distillation loss and the triplet loss in Equation (9);
    Use the sampler to get batch data from $\mathbb{D}_{val}$;
    Update the architecture parameters α via the distillation loss and the triplet loss in Equation (9) ;
Derive the final architecture based on chosen operations ;
Optimize the final architecture on the $\mathbb{D}_T$ by the standard reID training strategy ;
Evaluate the trained final architecture on the evaluation set $\mathbb{D}_E$

3.3. Evaluation Based on Knowledge Distillation

3.3.1. Knowledge Distillation

Knowledge distillation is wildly used in model compression because of the feature that it aims to transfer knowledge from a pre-trained teacher model to a smaller and faster student model. In the process of knowledge distillation, an appropriate soft target set was obtained by increasing the temperature parameter of the softmax layer of the teacher model. Then, for the student model to be trained, the same temperature parameter value is used to match the soft target set of the teacher model as part of the total objective function of the student model. Thus, it can induce the training of the student model and realize the transfer of knowledge. Hinton et al. [5] redefined knowledge distillation of training a student network to approach to the teacher's output after the softmax layer. As we know, neural networks typically use a softmax output layer that converts the logit, and the class probabilities q produced as follows:

$$q_i = \frac{exp(l_i/T)}{\sum_j exp(l_j/T)} \tag{5}$$

where q_i is the probability of a class, l_i is the logit, and l_j indicate other logits. The softmax output converges to a one-hot vector as the temperature T approaches 0. As the temperature T approaches infinity, the softmax output is softer. Therefore, when training the new model, a higher temperature T can be used to make the distribution produced by softmax soft enough, and then the softmax output of the new model (at the same temperature) is approximate to the original model. The normal temperature $T = 1$ was used to predict after the training. The normal temperature $T = 1$ was used to predict after the training. A softer probability distribution over classes computed by a higher value of T.

3.3.2. Architecture Search with Distilled Knowledge

Motivated by Hinton et al. [5], we proposed to introduce knowledge distillation to automatically design a convolutional network that is suitable for Re-ID tasks. More concretely, distillation loss is applied to guide the searching process to allow all candidates to learn the knowledge from a manual design teacher model that is pre-trained and has a good performance in Re-ID tasks. Let L_c denote cross-entropy loss, and L_c is widely used to train convolutional networks for classification tasks:

$$L_c = \sum_{i=1}^{B} -log \frac{exp(l_i[c])}{\sum_{c'=1}^{C} exp(exp(l_i[c']))} \tag{6}$$

where C denotes the number of training identities, $l_i[c]$ indicates the c-th element in l_i, and B indicates the number of samples in each batch during training. We introduced distillation loss L_d instead of L_c during the searching phase, L_d aims to compute the distance between teacher's outputs and student's outputs based on cross-entropy loss:

$$L_d = L_c(q_t, q_s) \tag{7}$$

where q_t is the outputs of the teacher model, and q_s is the outputs of the student model. Otherwise, following previous work [4], we also applied the triplet loss L_t to make the final architecture better suitable to Re-ID tasks:

$$L_t = \sum_{a,p,n} (margin, \max_{p \in B_p} ||f_a - f_p|| - \min_{n \in B_n} ||f_a - f_n||) \tag{8}$$

where B is a batch sample, B_p is a set of B that has the same ID with anchor a, and another set has different IDs with anchor denoted as B_n. The margin term indicates the margin of triplet loss. f_a is the feature of sample a, f_p is the feature of a positive sample p, and f_n indicates the feature of a negative sample n. Training data in each batch are sampled by a class-balance data sampler which first samples uniformly some sample identities, and then randomly samples the same number of images for each identity. Considering the above problem mentioned, our search objective L_{obj} can be summarized as:

$$L_{obj} = \lambda_1 L_d + \lambda_2 L_t \tag{9}$$

where λ_1 and λ_2 represent the weight ratio of L_d and L_t. In the experiment, we refer to Auto-ReID [4] to set the same weight for these two loss functions.

3.3.3. Knowledge Distillation with Dynamic Temperature

The target of our work is not only to automatically design an architecture that performs well in Re-ID tasks and has much fewer parameters compared to the teacher model but one that hopes to break the limit of the teacher model. In other words, we considered a distillation loss with dynamic temperature to train the searched CNN. We started searching and training with a high temperature so that the student model can better learn the knowledge from the teacher model through the soft target. In addition, then, temperature T will progressively reduce to 0 during the search process or training process. The importance of the knowledge from the teacher model will gradually decrease due to the improve-

ment of the student model. Thus, the student model should pay more attention to the labels. We control the importance of the knowledge from the teacher model via controlling temperature T. Finally, L_d equals to L_c when the T is reduced to 0.

4. Experiments

4.1. Datasets and Evaluation Metrics

4.1.1. Datasets

Market-1501 [3] is a high-quality pedestrian re-identification dataset composed of 1501 pedestrians captured by six cameras. Among them, the training set has 751 people and contains 12,936 images; the test set has 750 people and contains 19,732 images. It means that each person has 17.2 pieces of training data and 26.3 pieces of test data on average. After adding 500K irrelevant pictures, it is a large-scale pedestrian re-identification dataset with 32,668 + 500K bboxes and 3368 query pictures.

CUHK03 [29] contains 14,096 pictures from 1467 identities collected by five pairs of cameras. In [29], 100 identities are randomly selected as the test set, and the remaining identities are used as the training set. We use the new protocol proposed in [30] to divide the data set into a training set of 767 identities and a test set of 700 identities.

MSMT17 [31] is a dataset that is closer to real life scenarios. In four days with different weather, 12 outdoor cameras and three indoor cameras collect videos at three different time periods every day. Then, 126,441 bounding boxes of 4101 identities were obtained, of which the training set contained 32,621 bounding boxes of 1041 pedestrians, and the test set contained 93,820 bounding boxes of 3060 pedestrians. For the test set, 11,659 bounding boxes were randomly selected as the query, and the other 82,161 bounding boxes were used as the gallery. Therefore, this is a more challenging data set due to the fact that it has both indoor and outdoor scene information of different weather and time periods.

4.1.2. Evaluation Metrics

In order to better evaluate our KDAS-ReID and compare it with other general Re-ID methods, two general evaluation metrics are considered during the comparison. We use the rank-1, rank-5, and rank-10. The cumulative matching characteristics (CMC) are used as one of the evaluation indicators. The other evaluation indicators are to calculate the mean average precision (mAP) of the above three benchmarks according to common settings [3,31].

4.2. Implementation Details

4.2.1. Search Configurations

As mentioned in Algorithm 1, we equally split $\mathbb{D}_T$ into the search training set $\mathbb{D}_{train}$ and the search validation set $\mathbb{D}_{val}$ by randomly selecting data from official training data on Market-1501 during the searching stage. In addition, we use the macro-struct of Auto-ReID which is stacked by eight cells and set the channel of the first convolutional layer $C = 32$. By default, the Reduction Cell is set in the 3rd, 5th, and 7th place of cells to double the number of channels.

During this period, the class-balance data-sampler will randomly select eight identities and sample two images for each identity in a batch with a batch size of 16 and the total epoch of 200. We use random horizontal flipping and cropping for data augmentation. For a momentum SGD optimizer with the initial learning rate of 0.1, decreasing it to 0.001 in a cosine scheduler is used to optimize the weights w of supernet. We use an Adam optimizer with an initial learning rate of 0.02 to optimize the architecture parameters α. The weight decay for both SGD and Adam is set as 0.0005. The margin is set to 0.3 and $\lambda_1 = \lambda_2 = 1$ in the objective loss in Equation (9). We start searching with a high value of temperature $T = 2$ and decrease evenly. The temperature T will eventually decrease to 0 at the 190th epoch.

4.2.2. Training Configurations

After obtaining the final architecture, a macro-struct stacked by eight cells and the first convolutional layer channels $C = 64$ is adopted. In addition, we use the same input size of 384×128 as the searching period and also use random horizontal flipping and cropping for data augmentation. For optimizer, only an Adam optimizer with a 0.00035 starting learning rate, a 0.0005 weight decay, and a 0.0005 weight decay bias is used to optimize final architecture weights w. The class-balance data sampler will randomly select eight identities and four images for each identity at each batch, and 16 identities will be selected when using ImageNet pre-trained models. Therefore, the total training epoch is 200 and both of the objective loss settings in Equation (9) and the temperature T setting are identical to the searching period.

4.3. Ablation Study

In this section, we will investigate the effect of each component of the KDAS-ReID algorithm after an extensive experiment on the Market-1501. In addition, we show the results in Table 1.

As shown in Table 1, compared to traditional manual architecture and backbones, which are searched on CIFAR-10 such as DARTS [15] and GDAS, our work has competitive performance. In our ablation experiment, we make several observations as follows:

- Baseline. Directly searching on the Re-ID dataset with differentiable architecture search strategy and adopted the search space of Auto-ReID [4]. The loss is computed by the cross-entropy function and the triplet function.
- Baseline + Distillation. Based on Baseline, we use the objective loss (Equation (9)) which is composed of triplet loss and distillation loss instead of the original loss function. Considering the effect of the distillation loss heavily relying on a well-performance teacher model, we use the official ResNet-50 network which pre-trains on the ImageNet as the backbone of the teacher model. In addition, the temperature $T = 2$ is retained during the whole search stage.
- Distillation + Dynamic Temperature. To break the limit of the teacher model, we introduce the distillation loss with dynamic temperature T. We start searching with a high value of temperature $T = 2$ and decrease evenly; finally, the temperature T will decrease to 0 at the 190th epoch.

All of the above experiments ran four times under the same training configuration to ensure fairness. Although we can find that the CNNs searched with distillation loss has a better performance than the CNNs searched only with triplet loss, there is still room for improvement. We have found the best architecture through the method "Distillation + Dynamic Temperature" called KDAS-ReID. In addition, extensive experiments in Section 4.4 showed that the KDAS-ReID not only has the best performance but also broke the limit of the teacher model.

4.4. Architecture Evaluation

To enable the model to obtain the best performance, the latest Re-ID algorithm will be pre-trained on ImageNet first. In order to ensure a fair comparison with other state-of-the-art algorithms, we also put our algorithm on ImageNet for pre-training.

Results on Market-1501. We compare our method with the state-of-the-art Re-ID model in Table 2. The CNN found by our KDAS-ReID reached a rank-1 of 94.7% and a mAP of 85.3%, which is better than other state-of-the-art Re-ID models. Our KDAS-ReID obtains higher accuracy and mAP than the teacher model based on ResNet-50, and reduces the parameters of the teacher model by more than 43%. In addition, after using the same augmentation technology, our KDAS-ReID also outperforms other Re-ID models.

Results on CUHK03. Table 3 shows the comparison between KDAS-ReID and other models. Whether in the bounding box of manually marked or automatically detected people, our KDAS-ReID has shown better performance than other models.

Results on MSMT17 in Table 4. On MSMT17, the rank-1 accuracy of our KDAS-ReID achieves 78.4%, and the mAP achieves 53.2%. In addition, our model outperforms the teacher model and the previous state-of-the-art method.

Table 1. We investigated the effect of each component of the KDAS-ReID. All candidates are trained in the same strategy. In addition, no candidates used pre-training on ImageNet except the teacher model. We use the official ResNet-50 network which pre-trains on the ImageNet as the backbone of the teacher model. We have boldfaced the best results in the table.

Architectures	mAP	Rank-1	Rank-5	Rank-10	Params(M)
ResNet-18 [1]	66.0	85.2	94.6	96.5	11.6
ResNet-34 [1]	68.0	86.7	94.8	96.6	21.7
ResNet-50 [1]	68.5	87.2	95.5	97.1	25.1
DARTS [15]	65.2	85.6	94.3	96.4	**9.1**
GDAS [32]	66.8	86.5	94.7	96.9	13.5
Baseline 1	71.7	87.9	95.9	97.4	11.2
Baseline 2	71.5	87.0	95.7	97.3	9.8
Baseline 3	72.3	89.0	96.5	97.9	12.0
Baseline 4	72.1	88.6	96.4	97.9	11.4
Baseline + Distillation 1	74.7	89.7	96.2	97.6	16.5
Baseline + Distillation 2	75.8	91.0	96.6	98.0	15.0
Baseline + Distillation 3	75.3	89.9	96.5	**98.1**	14.3
Baseline + Distillation 4	74.9	89.3	96.5	97.7	12.1
Distillation + Dynamic Temperature 1	**77.0**	**91.1**	**97.0**	97.9	14.3
Distillation + Dynamic Temperature 2	76.9	**91.1**	96.7	97.9	15.7
Distillation + Dynamic Temperature 3	76.0	90.0	96.6	98.0	16.4
Distillation + Dynamic Temperature 4	75.7	90.1	96.5	98.0	15.4

Table 2. The current state-of-the-art model is compared with our model on the Market-1501 dataset. We compare each model in terms of parameters, R-1 (Rank-1) accuracy, and mAP. We have boldfaced the best results in the table.

Methods	Backbone	Params(M)	Market-1501	
			R-1	mAP
PAN [33]	ResNet50	>25.1	82.8	63.3
TriNet [34]	ResNet50	25.1	84.9	69.1
AOS [35]	ResNet50	>25.1	86.4	70.4
MLFN [36]	ResNeXt-50	>25.0	90.0	74.3
DuATM [37]	DenseNet-121	>8.0	91.4	76.6
PCB [11]	ResNet50	27.2	93.8	81.6
Mancs [38]	ResNet50	>25.1	93.1	82.3
HPM [39]	ResNet50	25.1	94.2	82.7
Auto-ReID [4]	-	13.1	94.5	85.1
Teacher Model	ResNet50	>25.1	93.7	83.2
KDAS-ReID	-	14.3	94.7	85.3
Using the re-ranking technique [30]				
TriNet [34]	ResNet50	25.1	86.7	81.1
AOS [35]	ResNet50	>25.1	88.7	83.3
AACN [40]	GoogleNet	>8.0	88.7	83.0
PSE+ECN [10]	ResNet50	>25.1	90.3	84.0
PCB [11]	ResNet50	27.2	95.1	91.9
Auto-ReID [4]	-	13.1	95.4	94.2
Teacher Model	ResNet50	>25.1	94.7	93.1
KDAS-ReID	-	14.3	**95.6**	**94.7**

Table 3. We use the new evaluation protocol in [30] to evaluate on the data set CUHK03. We compare with the most advanced Re-ID model in terms of Rank-1 accuracy and mAP. We have boldfaced the best results in the table.

Methods	Labeled		Detected	
	Rank-1	mAP	Rank-1	mAP
PAN [30]	36.9	35.0	36.3	34.0
SVDNet [41]	40.9	37.8	41.5	37.3
HA-CNN [42]	44.4	41.0	41.7	38.6
AOS [35]	-	-	47.7	43.3
MLFN [36]	54.7	49.2	52.8	47.8
PCB [11]	-	-	63.7	57.5
Mancs [38]	69.0	63.9	65.5	60.5
DG-Net [43]	-	-	65.6	61.1
Auto-ReID [4]	77.9	73.0	73.3	69.3
Teacher Model	76.3	71.5	71.9	68.0
KDAS-ReID	**78.0**	**73.2**	**73.4**	**70.0**

Table 4. Comparison of accuracy and mAP with the state-of-art reID models on MSMT17. We have boldfaced the best results in the table.

Methods	Rank-1	Rank-5	Rank-10	mAP
GoogleNet [6]	47.6	65.0	71.8	23.0
PDC [12]	58.0	73.6	79.4	29.7
GLAD [44]	61.4	76.8	81.6	34.0
PCB [11]	68.2	81.2	85.5	40.4
Auto-ReID [4]	78.2	88.2	91.1	52.5
Teacher Model	77.1	86.6	90.1	51.2
KDAS-ReID	**78.4**	**88.3**	**91.1**	**53.2**

5. Conclusions

In this paper, we propose a novel algorithm that automated neural architecture search for the Re-ID tasks based on Knowledge Distillation, and we name our method as KDAS-ReID. We have made progress based on Auto-ReID via transferring the knowledge from the teacher model to the student model. Furthermore, we introduced distillation loss with a dynamic temperature in order to break the limit of the teacher model. In our experiments, the KDAS-ReID outperforms other state-of-the-art Re-ID models on Market-1501, CUHK03, and MSMT17. In addition, the KDAS-ReID also outperforms the teacher model which was built upon the ResNet-50 backbone and using pre-training on the ImageNet. In the future, we consider the other ways of knowledge distillation such as employing the internal representation of the teacher to guide the searching and training of the student.

Author Contributions: Conceptualization, K.Y.; Formal analysis, K.Y.; Project administration, Z.L.; Software, K.J.; Validation, S.C. and Z.L.; Writing—original draft, K.Y. and K.J.; Writing—review & editing, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61572306 and Grant 61502294, in part by the National Key Research and Development Program of China under Grant 2017YFB0701600, and in part by the Shanghai Engineering Research Center of Intelligent Computing System under Grant 19DZ2252600.

Data Availability Statement: Data is contained within the article.

Acknowledgments: The authors thank the High-Performance Computing Center of Shanghai University for providing computing resources.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
2. Ye, M.; Shen, J.; Lin, G.; Xiang, T.; Shao, L.; Hoi, S.C. Deep learning for person re-identification: A survey and outlook. *arXiv* **2020**, arXiv:2001.04193.
3. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; Tian, Q. Scalable person re-identification: A benchmark. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1116–1124.
4. Quan, R.; Dong, X.; Wu, Y.; Zhu, L.; Yang, Y. Auto-reid: Searching for a part-aware convnet for person re-identification. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3750–3759.
5. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
6. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
7. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
8. Xiao, T.; Li, H.; Ouyang, W.; Wang, X. Learning deep feature representations with domain guided dropout for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1249–1258.
9. Chen, W.; Chen, X.; Zhang, J.; Huang, K. Beyond triplet loss: A deep quadruplet network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 403–412.
10. Sarfraz, M.S.; Schumann, A.; Eberle, A.; Stiefelhagen, R. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 420–429.
11. Sun, Y.; Zheng, L.; Yang, Y.; Tian, Q.; Wang, S. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 480–496.
12. Su, C.; Li, J.; Zhang, S.; Xing, J.; Gao, W.; Tian, Q. Pose-driven deep convolutional model for person re-identification. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3960–3969.
13. Suh, Y.; Wang, J.; Tang, S.; Mei, T.; Mu Lee, K. Part-aligned bilinear representations for person re-identification. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 402–419.
14. Suganuma, M.; Shirakawa, S.; Nagao, T. A genetic programming approach to designing convolutional neural network architectures. In Proceedings of the Genetic and Evolutionary Computation Conference, Berlin, Germany, 15–19 July 2017; pp. 497–504.
15. Liu, H.; Simonyan, K.; Yang, Y. Darts: Differentiable architecture search. *arXiv* **2018**, arXiv:1806.09055.
16. Zoph, B.; Le, Q.V. Neural architecture search with reinforcement learning. *arXiv* **2016**, arXiv:1611.01578.
17. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8697–8710.
18. Brock, A.; Lim, T.; Ritchie, J.M.; Weston, N. Smash: One-shot model architecture search through hypernetworks. *arXiv* **2017**, arXiv:1708.05344.
19. Real, E.; Aggarwal, A.; Huang, Y.; Le, Q.V. Regularized evolution for image classifier architecture search. In Proceedings of the 2019 AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 4780–4789.
20. Bender, G.; Kindermans, P.J.; Zoph, B.; Vasudevan, V.; Le, Q. Understanding and simplifying one-shot architecture search. In Proceedings of the International Conference on Machine Learning PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 550–559.
21. Lorenzo, P.R.; Nalepa, J. Memetic evolution of deep neural networks. In Proceedings of the Genetic and Evolutionary Computation Conference, Kyoto, Japan, 15–19 July 2018; pp. 505–512.
22. Xiao, X.; Yan, M.; Basodi, S.; Ji, C.; Pan, Y. Efficient hyperparameter optimization in deep learning using a variable length genetic algorithm. *arXiv* **2020**, arXiv:2006.12703.
23. Lorenzo, P.R.; Nalepa, J.; Kawulok, M.; Ramos, L.S.; Pastor, J.R. Particle swarm optimization for hyper-parameter selection in deep neural networks. In Proceedings of the Genetic and Evolutionary Computation Conference, Berlin, Germany, 15–19 July 2017; pp. 481–488.
24. Ba, L.J.; Caruana, R. Do deep nets really need to be deep? *arXiv* **2013**, arXiv:1312.6184.
25. Romero, A.; Ballas, N.; Kahou, S.E.; Chassang, A.; Gatta, C.; Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv* **2014**, arXiv:1412.6550.
26. Zagoruyko, S.; Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv* **2016**, arXiv:1612.03928.
27. Yim, J.; Joo, D.; Bae, J.; Kim, J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4133–4141.
28. Sau, B.B.; Balasubramanian, V.N. Deep model compression: Distilling knowledge from noisy teachers. *arXiv* **2016**, arXiv:1610.09650.

29. Li, W.; Zhao, R.; Xiao, T.; Wang, X. Deepreid: Deep filter pairing neural network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 152–159.
30. Zhong, Z.; Zheng, L.; Cao, D.; Li, S. Re-ranking person re-identification with k-reciprocal encoding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1318–1327.
31. Wei, L.; Zhang, S.; Gao, W.; Tian, Q. Person transfer gan to bridge domain gap for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 79–88.
32. Dong, X.; Yang, Y. Searching for a robust neural architecture in four gpu hours. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1761–1770.
33. Zheng, Z.; Zheng, L.; Yang, Y. Pedestrian alignment network for large-scale person re-identification. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 3037–3045. [CrossRef]
34. Hermans, A.; Beyer, L.; Leibe, B. In defense of the triplet loss for person re-identification. *arXiv* **2017**, arXiv:1703.07737.
35. Huang, H.; Li, D.; Zhang, Z.; Chen, X.; Huang, K. Adversarially occluded samples for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5098–5107.
36. Chang, X.; Hospedales, T.M.; Xiang, T. Multi-level factorisation net for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2109–2118.
37. Si, J.; Zhang, H.; Li, C.G.; Kuen, J.; Kong, X.; Kot, A.C.; Wang, G. Dual attention matching network for context-aware feature sequence based person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5363–5372.
38. Wang, C.; Zhang, Q.; Huang, C.; Liu, W.; Wang, X. Mancs: A multi-task attentional network with curriculum sampling for person re-identification. In Proceedings of the European Conference on Computer Vision (ECCV), Berlin, Germany, 27 March 2018; pp. 365–381.
39. Fu, Y.; Wei, Y.; Zhou, Y.; Shi, H.; Huang, G.; Wang, X.; Yao, Z.; Huang, T. Horizontal pyramid matching for person re-identification. In Proceedings of the AAAI Conference on Artificial Intelligence, Dublin, Ireland, 17–19 October 2019; Volume 33, pp. 8295–8302.
40. Xu, J.; Zhao, R.; Zhu, F.; Wang, H.; Ouyang, W. Attention-aware compositional network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2119–2128.
41. Sun, Y.; Zheng, L.; Deng, W.; Wang, S. Svdnet for pedestrian retrieval. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3800–3808.
42. Li, W.; Zhu, X.; Gong, S. Harmonious attention network for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2285–2294.
43. Zheng, Z.; Yang, X.; Yu, Z.; Zheng, L.; Yang, Y.; Kautz, J. Joint discriminative and generative learning for person re-identification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2138–2147.
44. Wei, L.; Zhang, S.; Yao, H.; Gao, W.; Tian, Q. Glad: Global-local-alignment descriptor for pedestrian retrieval. In Proceedings of the 25th ACM international conference on Multimedia, Aires, Argentina, 22–23 May 2017; pp. 420–428.