Resource Aware Person Re-identification across Multiple Resolutions

Yan Wang*[1], Lequn Wang*[1], Yurong You*[2], Xu Zou[3], Vincent Chen[1],
Serena Li[1], Gao Huang[1], Bharath Hariharan[1], Kilian Q. Weinberger[1]
Cornell University[1], Shanghai Jiao Tong University[2], Tsinghua University[3]
{yw763, lw633}@cornell.edu, yurongyou@sjtu.edu.cn, zoux14@mails.tsinghua.edu.cn
{zc346, sl2327, gh349}@cornell.edu, bharathh@cs.cornell.edu, kqw4@cornell.edu

Abstract

Not all people are equally easy to identify: color statistics might be enough for some cases while others might require careful reasoning about high- and low-level details. However, prevailing person re-identification(re-ID) methods use one-size-fits-all high-level embeddings from deep convolutional networks for all cases. This might limit their accuracy on difficult examples or makes them needlessly expensive for the easy ones. To remedy this, we present a new person re-ID model that combines effective embeddings built on multiple convolutional network layers, trained with deep-supervision. On traditional re-ID benchmarks, our method improves substantially over the previous state-of-the-art results on all five datasets that we evaluate on. We then propose two new formulations of the person re-ID problem under resource-constraints, and show how our model can be used to effectively trade off accuracy and computation in the presence of resource constraints.

1. Introduction

Consider the two men shown in Figure 1. The man on the left is easier to identify: even from far away, or on a low-resolution photograph, one can easily recognize the brightly colored attire with medals of various kinds. By contrast, the man on the right has a nondescript appearance. One might need to look closely at the set of the eyes, the facial hair, the kind of briefcase he is holding or other such subtle and fine-grained properties to identify him correctly.

Current person re-identification(re-ID) systems treat both persons the same way. Both images would be run through deep convolutional neural networks (CNNs). Coarse-resolution and semantic embeddings from the last layer would be used to look the image up in the database. However, this kind of an architecture causes two major problems: first, for the hard cases such as the man on the right in Figure 1, these embeddings are too coarse and discard too much information. Features from the last layer of a CNN mostly encode semantic features, like object presence [15], but lose all information about the fine spatial details such as the pattern of one's facial hair or the particular shape of one's body. Instead, to tackle both cases, ideally we would want to reason *jointly* across multiple levels of semantic abstraction, taking into account both high-resolution (shape and color), as well as highly semantic details (objects or object parts).

Figure 1. Some people have distinctive appearance and are easy to identify (left), while others have nondescript appearance and require sophisticated reasoning to identify correctly (right).

In contrast, for the easy cases such as the man on the left in Figure 1, using a 50-layer network is overkill. A color histogram or the low-level statistics computed in the early layers of the network might work just as well. This may not be a problem if all we are interested in is the final accuracy. However, sometimes we need to be more resource efficient in terms of time, memory, or power. For example, a robot might need to make decisions within a time limit, or it may have a limited battery supply that precludes the running of a massive CNN on every frame.

Thus standard CNN-based person re-ID systems are only *one* point on a spectrum. On one end, early layers of the CNN can be used to identify people quickly under some resource constraints, but might sacrifice accuracy on hard images. On the other end of the spectrum, highly accurate person re-ID might require reasoning across multiple layers of the CNN. Ideally, we want a *single* model that encapsu-

*Authors contributed equally.

lates the entire spectrum. This can allow downstream applications to choose the right trade-off between accuracy and computation.

In this paper we present such a person re-ID system. Our model has a simple architecture, consisting of a standard base network with two straightforward modifications. First, embeddings across multiple layers are combined into a single embedding. Second, embeddings at each stage are trained in a supervised manner for the end task. While both ideas have appeared before in various forms for object detection and segmentation [8, 15, 53], we show for the first time the benefit of these ideas for person re-ID problems, and connect these ideas to the goal of performance under resource constraints.

We evaluate our approach on five well-known person re-ID benchmark datasets. Not only does our method outperform all previous approaches across all datasets, it is also to our knowledge the first person re-ID algorithm applicable to the resource budget settings in test time.

2. Related Work

We briefly review prior work on person re-ID and deep supervision.

2.1. Person re-ID

Traditional person re-ID methods first extract discriminative hand-crafted features that are robust to illumination and viewpoint changes [9,13,24,32,40,41,59], and then use metric learning [2,6,12,18,22,31,32,33,36,43,54,58,63] to ensure that features from the same person are close to each other while from different people are far away in the embedding space. Meanwhile, researchers have worked on creating ever more complex person re-ID datasets [28,44,60,61] to imitate real-world challenges.

Inspired by the success of CNNs [25] on a variety of vision tasks, recent papers have employed deep learning in person re-ID [1, 5, 28, 29, 34, 38, 50, 52, 64, 65]. CNN-based models are on the top of the scoreboard. This paper belongs to this large family of CNN-based person re-ID approaches.

There are three types of deep person re-ID models: classification, verification, and distance metric learning. Classification models consider each identity as a separate class, converting re-ID into a multi-class recognition task [48, 52, 62]. Verification models [28, 49, 55] take a pair of images as input to output a similarity score determining whether they are the same person. A related class of models learns distance metrics [3, 5, 7, 17, 46] in the embedding space directly in an expressive way. Hermans et al. [17] propose a variant of these models that uses the triplet loss with batch hard negative and positive mining to map images into a space where images with the same identity are closer than those of different identities. We also utilize the triplet loss to train our network, but focus on improvements to the architecture. Combinations of these loss functions have also been explored [4, 11, 38].

Instead of tuning the loss function, other researchers have worked on improving the training procedure, the network architecture, and the pre-processing. In order to alleviate problems due to occlusion, Zhong et al. [67] propose to randomly erase some parts of the input images as the antidote. Treating re-ID as a retrieval problem, re-ranking approaches [66] aim to get robust ranking by lifting up the k-reciprocal nearest neighbors. Under the assumption that correlated weight vectors damp the retrieval performance, Sun et al. [48] attempt to de-correlate the weights of the last layer. These improvements are orthogonal to our proposed approach. In fact, we integrate random erasing and re-ranking into our approach for better performance.

Some works explicitly consider local features or multi-scale features in the neural networks [11, 27, 30, 37, 47, 56, 57]. By contrast, we implicitly combine features across scale and abstraction by tapping into the different stages of the convolutional network.

2.2. Deep supervision and skip connections

The idea of using multiple layers of a CNN has been explored before. Combining features across multiple layers using *skip connections* has proved to be extremely beneficial for segmentation [8,15,39] and object detection [35]. In addition, prior work has found that injecting supervision by making predictions at intermediate layers improves performance. This *deep supervision* improves both image classification [26] and segmentation [53]. We show that the combination of deep supervision with distance metric learning leads to significant improvements in solving person re-ID problems.

We also present that, under limited resource, accurate prediction is still possible with deep supervision and skip connections. In spite of the key role that efficiency of inference plays in real-world applications, there is very little work incorporating such resource constraints, not even in general image classification setting (exception: [19]).

3. Deep supervision for person re-ID

We first consider the traditional person re-ID setting. Here, the system has a *gallery* $\mathcal{G}$ of images from different people with known identities. It is then given a *query*/*probe* image q of an unidentified person, which can also be multiple images. The objective of the system is to match the probe with image(s) in the gallery to identify that person.

Previous approaches to person re-ID only use the most high level features to encode an image, e.g., outputs of the last convolution layer form the ResNet-50 [16]. Although high-level features are indeed useful in forming abstract concepts for object recognition, they might discard low-level signals like color and texture, which are important clues for person re-ID. Furthermore, later layers in CNNs

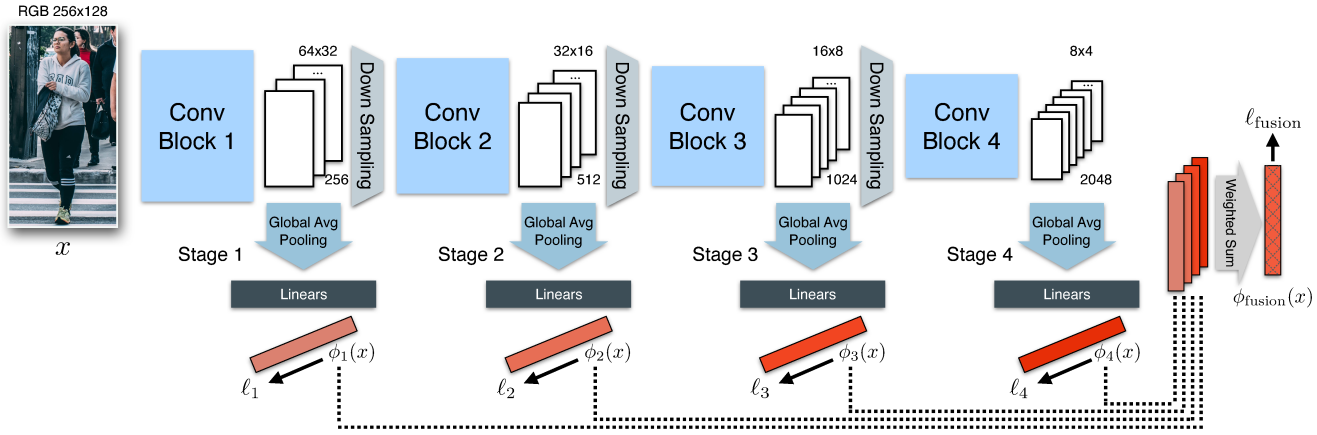

Figure 2. Illustration of Deep Anytime Re-ID (DaRe) for person re-ID. The model is based on ResNet-50 [16] which consists of four *stages*, each with decreasing resolution. DaRe adds extra global average pooling and fully connected layers right after each stage starting from stage 1 (corresponding to conv_2-5x in [16]). Different parts are trained jointly with loss $\ell_{all} = \sum_{s=1}^{4} \ell_s + \ell_{\text{fusion}}$. When inferring under constrained-resource settings, DaRe will output the most recent available embeddings from intermediate stages (and the ensemble embedding when computation resource is enough for a full pass of the network). (Example image copyright Kaique Rocha (CC0 License)).

are at a coarser resolution, and may not see fine-level details such as patterns on clothes, facial features, subtle pose differences etc. This suggests that person re-ID will benefit from *fusing* information across multiple layers.

However, such fusion of multiple features will only be useful if each individual feature vector is discriminative enough for the task at hand. Otherwise, adding in uninformative features might end up adding noise and degrade task performance.

With this intuition in mind, we introduce a novel architecture for person re-ID, which we refer to as *Deep Anytime Re-ID (DaRe)*, as illustrated in Figure 2. Compared to prior work on person re-ID, the architecture a) fuses information from multiple layers [8, 15], and b) has intermediate losses that train the embeddings from different layers (*deep supervision* [53]) for person re-ID directly with a variant of the triplet loss.

3.1. Network architecture

Our base network is a residual network (ResNet50) [16]. This network has four *stages*, each halves the resolution of the previous. Each stage contains multiple convolutional layers operating on feature maps of the same resolution. At the end of each stage, the feature maps are down-sampled and fed into the next layer.

We take the feature map at the end of *each* stage and use global average pooling followed by two fully connected layers to produce an embedding at each stage. The first fully connected layer has 1204 units including batch normalization and ReLU and the second layer has 128 units. The function of the fully connected layers is only to bring all embeddings to the same dimension.

Given an image x, denote by $\phi_s(x)$ the embedding produced at stage s. We fuse these embeddings using a simple weighted sum:

$$\phi_{\text{fusion}}(x) = \sum_{s=1}^{4} w_s \phi_s(x), \tag{1}$$

where the weights w_s are learnable parameters.

3.2. Loss function

The loss function we use to train our network is the sum of *per-stage* loss functions ℓ_s operating on the embedding $\phi_s(x)$ from every stage s and a loss function on the final fused embedding $\phi_{\text{fusion}}(x)$: $\ell_{\text{all}} = \sum_{s=1}^{4} \ell_s + \ell_{\text{fusion}}$.

For each loss function, we use the the triplet loss. The triplet loss is commonly used in metric learning [45,51] and recently introduced to person re-ID [5, 17].

The reason for using triplet loss is threefold: 1) It minimizes the nearest neighbor loss via expressive embeddings. 2) The triplet loss does not require more parameters as the number of identities in the training set increases. 3) Since it uses simple Euclidean distances, it can leverage well-engineered fast approximate nearest neighbor search (as opposed to the verification models, which construct feature vectors of pairs [42]).

Specifically, we adopt the triplet loss with batch hard mining and soft margin as proposed in [17], which reduces uninformative triplets and accelerates training. Given a batch of images X, of P individuals, the triplet loss takes K images per person and their corresponding identities Y in the following form:

$$\ell = \sum_{p=1}^{P} \sum_{k=1}^{K} \ln\Big(1 + \exp\Big(\overbrace{\max_{a=1,\dots,K} D\left(\phi(x_p^k), \phi(x_p^a)\right)}^{\text{furthest positive}} - \underbrace{\min_{\substack{q=1,\dots,P \\ b=1,\dots,K \\ q\neq p}} D\left(\phi(x_p^k), \phi(x_q^b)\right)}_{\text{nearest negative}}\Big)\Big), \tag{2}$$

where $\phi(x_p^k)$ is the feature embedding of person p image k and $D(\cdot,\cdot)$ is the L2 distance between two embeddings. The loss function encourages the distance to the furthest positive example to be smaller than to the nearest negative example.

4. Resource-constrained person re-ID

The availability of multiple embeddings from different stages makes our model especially suitable for re-ID applications under resource constraints. In this section, we consider the person re-ID problem with limited computational resources and illustrate how DaRe can be applied under these scenarios.

4.1. Anytime person re-ID

In the anytime prediction setting [14, 19], the computational budget for a test example is *unknown a priori*, and the re-ID inference process is subject to running out of computation budget at any time. Although the anytime setting has hardly been studied for person re-ID, it is a common scenario in many settings. For example, imagine a person re-ID app for mobile Android devices that is supposed to perform at a fixed frame-rate. There exist over $24,093$ distinct Android devices [19] and it is infeasible to ship different versions of an application for each hardware configuration — instead one may want to ship a single network that can guarantee a given frame rate on all hardware configurations.

Here, a traditional re-ID system is all or nothing: it can only return any result if the budget allows for the evaluation of the full model.

Ideally, we would expect the system to have the anytime property, i.e., it is able to produce predictions early-on, but can keep refining the results when the budget allows. This mechanism can be easily achieved with DaRe: we propagate the input image through the network, and use the most recent intermediate embedding that was computed when the budget ran out to do the identification.

4.2. Budgeted person re-ID

In the budgeted person re-ID problem, the system runs in an online manner, but it is constrained to only use a budget B *in expectation* to compute the answer. The system needs to decide how much computation to spend on each example as it is observing them one by one. Because it only has to adhere to the budget in expectation, it can choose to spend more time on the hard examples as long as it can process easier samples more quickly.

We formalize the problem as following: let S be the number of exits (4 in our case), and $C_s > 0$ the amount of computational cost needed to obtain embedding $\phi_s(q)$ at stage s for a single query q ($C_s \leq C_{s+1}, \forall s = 1,\ldots,S-1$). At any stage s for a given query, we can decide to "exit": stop computation and use the s-th embedding to identify the query q. Let us denote the proportion of queries that exit at stage s as p_s, where $\sum_{s=1}^{S} p_s = 1$. Thus the expected average computation cost for a single query is $\bar{C} = \sum_{s=1}^{S} p_s C_s$.

Exit thresholds. Given the total number of queries M and the total computation budgets B, the parameters $\{p_s\}$ can be chosen such that $\bar{C} \leq B/M$, which represents the computation budget for each query. There are various ways to determine $\{p_s\}$. In practice we define

$$p_s = \frac{1}{Z} a^{s-1}, \tag{3}$$

where Z is the normalization constant and $a \in [0, \inf)$ a fixed constant. Given the costs $C_1,\ldots,C_S$, there is a one-to-one mapping between the budget B and a. If there were infinitely many stages, eq. (3) would imply that a fraction of a samples is exited at each stage. In the presence of finitely many exit stages it encourages an even number of early-exits across all stages. Given p_s, we can compute the conditional probability that an input which has traversed all the way to stage s will exit at stage s and not traverse any further as $q_1 = p_1$ and $q_s = \frac{p_s}{1-\sum_{i=1}^{s-1} p_i}$.

Once we have solved for q_s, we need to decide which queries exit where. As discussed in the introduction, query images are not equally difficult. If the system can make full use of this property and route the "easier" queries through earlier stages and "harder" ones through latter stages, it will yield a better budget-accuracy trade-off. We solidify this intuition using a simple distance based routing strategy to decide at which stage each query should exit.

Query easiness. During testing, at stage s, we would like to exit the top q_s percent of "easiest" samples. We approximate how "easy" a query q is by considering the distance d_q to its nearest neighbor between the query embedding $\phi_s(q)$ and its nearest neighbor in the gallery of the current stage s. A small distance d_q means that we have likely found a match and thus successfully identified the person correctly. During testing time we keep track of all previous distances $d_{q'}$ for all prior queries q'. For a given query q we check if its distance d_q falls into the fraction q_s of smallest nearest neighbor distances, and if it does exit the query at stage s.

If labels are available for the gallery at test time, one can perform a better margin based proxy of uncertainty. For a query q one computes the distance d_q to the nearest neighbor, and d'_q, the distance to the second nearest neighbor (with a different class membership than the nearest neighbor). The difference $d'_q - d_q$ describes the "margin of certainty". If it is large, then the nearest neighbor is sufficiently closer than the second nearest neighbor and there is little uncertainty. If it is small, then the first and second nearest neighbors are close in distance, leaving a fair amount of ambiguity. If labels are available, we use this difference $d'_q - d_q$ as our measure of uncertainty, and remove the top q_s most certain queries at each stage.

5. Experiments

We evaluate our method on multiple large scale person re-ID datasets, and compare with the state-of-the-art.

Method	Dataset									
	Market		MARS		CUHK03(L)		CUHK03(D)		Duke	
	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
CNN+DCGAN(R) [65]	56.2	78.1	-	-	-	-	-	-	67.7	47.1
ST-RNN(C) [68]	-	-	70.6	50.7	-	-	-	-	-	-
MSCAN(C) [27]	80.3	57.5	71.8	56.1	-	-	-	-	-	-
PAN(R) [64]	82.2	63.3	-	-	36.9	35.0	36.3	34.0	71.6	51.5
SVDNet(R) [48]	82.3	62.1	-	-	40.9	37.8	41.5	37.2	76.7	56.8
TriNet(R) [17]	84.9	69.1	79.8	67.7	-	-	-	-	-	-
TriNet(R)+RE* [67]	-	-	-	-	64.3	59.8	61.8	57.6	-	-
SVDNet(R)+RE [67]	87.1	71.3	-	-	-	-	-	-	79.3	62.4
DaRe(R)	86.4	69.3	83.0	69.7	58.1	53.7	55.1	51.3	75.2	57.4
DaRe(R)+RE	88.5	74.2	82.6	71.7	64.5	60.2	61.6	58.1	79.1	63.0
DaRe(De)	86.0	69.9	84.2	72.1	56.4	52.2	54.3	50.1	74.5	56.3
DaRe(De)+RE	**89.0**	**76.0**	**85.5**	**74.0**	**66.1**	**61.6**	**63.3**	**59.0**	**80.2**	**64.5**
IDE(C)+ML+RR [66]	61.8	46.8	67.9	58.0	25.9	27.8	26.4	26.9	-	-
IDE(R)+ML+RR [66]	77.1	63.6	73.9	68.5	38.1	40.3	34.7	37.4	-	-
TriNet(R)+RR [17]	86.7	81.1	81.2	77.4	-	-	-	-	-	-
TriNet(R)+RE+RR* [67]	-	-	-	-	70.9	71.7	68.9	69.36	-	-
SVDNet(R)+RE+RR [67]	89.1	83.9	-	-	-	-	-	-	84.0	78.3
DaRe(R)+RR	88.3	82.0	83.0	79.3	66.0	66.7	62.8	63.6	80.4	74.5
DaRe(R)+RE+RR	90.8	85.9	83.9	80.6	72.9	73.7	69.8	71.2	**84.4**	79.6
DaRe(De)+RR	88.6	82.2	84.8	80.3	63.4	64.1	60.2	61.6	79.7	73.3
DaRe(De)+RE+RR	**90.9**	**86.7**	**85.1**	**81.9**	**73.8**	**74.7**	**70.6**	**71.6**	**84.4**	**80.0**

Table 1. Rank-1 and mAP comparison of DaRe with other state-of-the-art methods on the Market-1501 (*Market*), MARS, CUHK03 and DukeMTMC-ReID (*Duke*) datasets. Results that surpass all competing methods are **bold**. For convenience of the description, we abbreviate CaffeNet to C, ResNet-50 to R, DenseNet-201 to De, Random erasing to RE and Re-ranking to RR. For CUHK03 dataset, we use the new evaluation protocol shown in [66], where L stands for hand labeled and D for DPM detected. * denotes that the result was obtained by our own re-implementation, which yields higher accuracy than the original result.

Datasets and evaluation metrics: Table 2 describes the datasets used in our experiments. The images in both Market-1501 [61] and MARS [60] are collected by 6 cameras (with overlapping fields of view) in front of a supermarket. Person bounding boxes are obtained from a DPM detector [10]. Each person is captured by two to six cameras. The images in CUHK03 [28] are also collected by 6 cameras, but without overlapping. The bounding boxes are either manually labeled or automatically generated. The DukeMTMC-reID [65] contains 36,411 images of 1,812 identities from 8 high-resolution cameras. Among them, 1,404 identities appear in more than two cameras, while 408 identities appear in only one camera. On all datasets, we use two standard evaluation metrics: rank-1 *Cumulative Matching Characteristic* accuracy (Rank-1) and mean average precision (mAP) [61]. On the CUHK03 dataset, we use the new protocol to split the training and test data as suggested by Zhong et al. [66]. For all datasets, we use the officially provided evaluation code to obtain the results. Our only modification is to use *mean* pooling on the embeddings of a tracklet instead of *max* pooling on MARS.

Dataset	Market [61]	MARS [60]	CUHK03 [28]	Duke [65]
Format	Image	Video	Image	Image
Identities	1,501	1,261	1360	1,812
BBoxes	32,668	1,191,003	13,164	36,411
Cameras	6	6	6	8
Label method	DPM	DPM+GMMCP	Hand/DPM	Hand
Train # imgs	12,936	509,914	7,368/7,365	16,522
Train # ids	751	625	767	702
Test # imgs	19,732	681,089	1,400	2,228
Test # ids	750	635	700	702

Table 2. The person re-ID datasets used in our experiments. All datasets include realistic challenges, amongst other things due to occlusion, changes in lighting and viewpoint, or mis-localized bounding boxes from object detectors.

Implementation details: We use the same settings as in [17], except that we train the network for 60,000 iterations instead of 25,000 to ensure a more thorough convergence for our joint loss function (we confirm that training the models in [17] for more iterations does not help).

Each image is first resized to 256×128, amplified by a factor 1.125, followed by a 256×128 crop and a ran-

dom horizontal flip. DaRe is built upon a ResNet-50 [16] or DenseNet-201 [20] model, pre-trained on ImageNet [23] (both have similar number of parameters). We refer to the two versions as *DaRe(R)* and *DaRe(D)*, respectively. To allow an easier comparison to the TriNet architecture, we performed all experiments in the ablation studies with the ResNet architecture. For notational simplicity we will sometimes drop the (R) in the name and assume that $DaRe$, without specification refers to the ResNet architecture. We train both versions of DaRe with Adam [21] and a batch size of 72, which contains 18 different people, 4 different images each. The learning rate α is adjusted similarly as in [17], starting from $\alpha_0 = 3 \times 10^{-4}$

$$\alpha(t) = \begin{cases} \alpha_0 & \text{if } t \leq t_0, \\ \alpha_0 \times 0.001^{\frac{t-t_0}{t_1-t_0}} & \text{if } t_0 \leq t \leq t_1, \end{cases} \quad (4)$$

where we set $t_0 = 30,000$ and $t_1 = 60,000$. β_1 in Adam will reduce to 0.5 from 0.9 after t_0 as well. Following [17], the final feature vector of each image during inference is the average over embedding vectors of five crops and their flips [23]. All hyperparameters were taken from [17], optimized for Market and MARS. Potentially we could improve the results of DaRe even further through proper hyperparameter tuning.

Pre-/post-processing: There are two model-agnostic pre- and post-processing steps that increase the accuracy of the person re-ID systems. *Random erasing* (RE) [67] involves randomly masking parts of the input during training to increase its robustness to occlusion. *Reranking* (RR) [66] uses the nearest neighbor graph of multiple probe and gallery images to rerank matches. In our experiments, we evaluate our model both with and without these processing steps.

5.1. Results on standard person-reID

We compare DaRe to existing state-of-the-art methods, and find that DaRe is competitive even without any random erasing or reranking. The results are shown in Table 1. We ran several of the experiments four times and found all standard deviations to be less than 0.5. In particular, DaRe(R) is uniformly better than TriNet [17], which uses the same base network, has a similar number of parameters, and is trained using the same version of triplet loss but without deep supervision or skip connections. Incorporating random erasing and re-ranking further boosts the results, giving DaRe(R) state-of-the-art performance on all four datasets. DaRe is further improved significantly when the base network is changed from ResNet-50 to DenseNet-201. Our model works well not only on small datasets like Market, CUHK and Duke, but also on large scale datasets like MARS (which contains 1 million images). Not surprisingly, random erasing tends to improve the performance substantially on the former, where overfitting can become an issue.

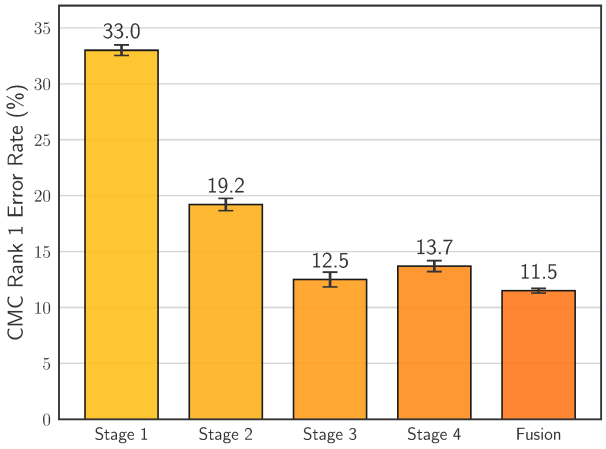

Figure 3. The Rank-1 error on the Market-1501 dataset without re-ranking across the different stages of DaRe and the final ensemble. Mean and std are estimated over four runs. DaRe is trained with Random Erasing.

5.2. Ablation

The results from Table 1 indicate that DaRe outperforms TriNet significantly. There are two possible factors behind this improvement: a) the fusion of information from multiple layers, and b) deep supervision. In the following we analyze the contribution of both factors on the ResNet-50 version of DaRe.

The impact of fusion: Figure 3 shows the performance of the different stages of DaRe, trained with random erasing and evaluated without re-ranking, on the Market-1501 dataset. As expected, the error rate decreases as one goes deeper into the network and the *fusion* of features from different stages actually achieves the lowest error rate.

However, note that stage 4 achieves lower error rate than stage 3. It is possible that features of the last stage are too "high level" and lose too much information due to an extra pooling layer. We further analyze the weights w_s for different stages. When trained with Random Erasing on Market1501, the learnt weights of the four stages are $-0.54, -0.73, -0.77, -0.51$. As expected, the absolute values of the weights of the third stage (the most accurate) is the largest.

Incidentally, note that the early layers of the network also achieve reasonable performance, with even stage 1 reaching a 33.3% error rate. This can probably be attributed to deep supervision, which we evaluate next.

The impact of deep supervision: We retrain DaRe without any deep supervision, i.e. we remove all intermediate losses except the loss on the fused feature vector. The results are presented in Figure 4. Without deep supervision, the error rates increase by 2%, suggesting that deep supervision is indeed required to make sure that each stage learns a good representation. Intuitively, without deep supervision,

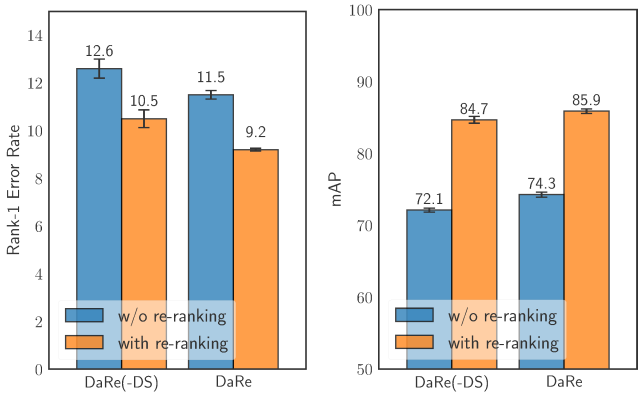

Figure 4. Rank-1 error rates (lower is better) and mAP (higher is better) of DaRe with and without deep supervision on Market1501 dataset. We show means and error bars from four trials.

the gradients from the loss on the fused feature vector are not informative of how each stage fares individually.

5.3. Results on resource-constrained person re-ID

We now show results on person re-ID under resource-constrained scenarios described in Section 4. All experiments in this section are conducted on Market-1501 dataset.

5.3.1 Anytime person re-ID

Baselines: We compare our DaRe model against a sequential ensemble of three ResNets (SE-ResNets) [16], consisting of a ResNet18, a ResNet34 and a ResNet50. All three ResNet models are trained separately using the same triplet loss as in [17]. At test time, the networks are evaluated sequentially in ascending order of size, and are forced to output the most recent re-ID result after surpassing the budget limit.

Anytime re-ID results: Figure 5 summarizes the results of the anytime setting. The computational cost is reported with respect to the cumulative number of multiplications and additions (Mul-Add). We confirm that the actual running timing is consistent with the Mul-Add. Note that we cannot perform re-ranking [66] in this setting since we cannot assume all queries are available at once. So we report the results from model "DaRe+RE" in Table 1.

Except for a narrow range of budgets, our DaRe model outperforms the SE-ResNets significantly. In particular, our model is able to achieve a high accuracy very quickly, achieving $3 \sim 5$ points higher performance for budgets higher than 2.5×10^9 Mul-Adds. This is because unlike the SE-ResNets, ours is a single model that *shares* computation between the "quick-and-dirty" and the slow-and-accurate predictions.

5.3.2 Budgeted stream person re-ID

Figure 6 shows the results in the budgeted streaming setting. We compare three variants of four stages DaRe. Each variant uses a different method to exit queries early. In the *random* setting, we interpolate between the individual stages (indicated as blue points). The straight lines between the blue dots are obtained by randomly deciding to exit queries at either one of the two corresponding stages, which yields a smooth interpolation between the budgets of the two stages. In the *distance* variant, the top q_s queries with the shortest distance to their nearest gallery neighbor are exited at each stage. In the *margin* variant inputs are exited based on their margin of certainty between the nearest neighbor and the second nearest neighbor of a different class. This latter version assumes the knowledge of class labels of gallery data points. We observe that our choice of thresholds are able to route queries effectively, allowing us to achieve higher accuracy at lower cost compared to prior state-of-the-art models (which are only single points without the ability to trade-off accuracy for computational cost). When gallery labels are available the *margin* selection method is clearly

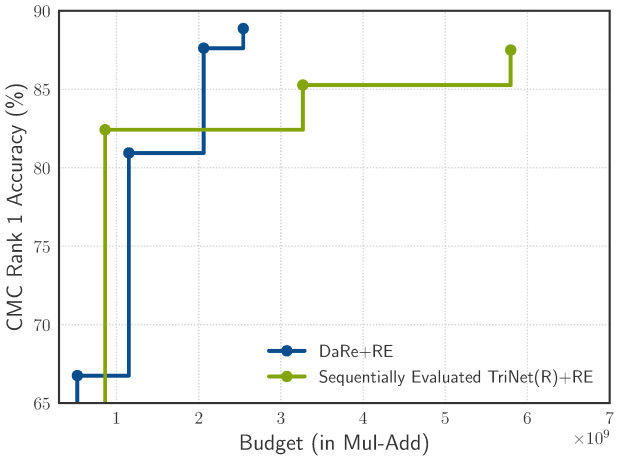

Figure 5. CMC rank 1 accuracy under *anytime re-ID* setting as a function of the computational budget on the Market-1501 dataset.

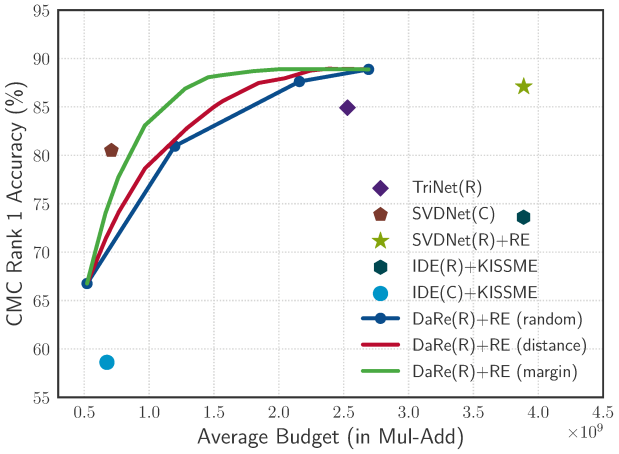

Figure 6. Results on the Market data set under the budgeted streaming setting. The graph shows the CMC Rank 1 accuracy as a function of the average budget in terms of Mul-Add.

preferred over the *distance* based method. Both methods outperform random interpolation.

Figure 7. Visualization of person re-ID results using features from different stages of our model. We purposely selected samples where the fused feature representation yields the correct results to show where it improves over earlier stages.

5.4. Qualitative results

To gain a better understanding of how the features from various stages differ in identifying people, and how the fusion helps, we visualize the retrieved images from four cases in Figure 7 for which the fused representation classifies the images correctly. The query images are shown in the left most column, and the retrieved images using the features from the four stages of ResNet-50 and the fused embedding are shown in column 2 to 6, respectively. Images with red boxes correspond to wrong identifications, while those with green boxes are correctly identified.

In the four cases, the fused features correctly identify the people from the query image, while low-level (e.g., from State 1) and high-level (e.g., from State 4) features may agree (Case 1, 2) or disagree (Case 3, 4) with each other. In Case 1, the low-level features are more helpful as the stripes on the clothes are important; while in Case 2, they overly emphasize the color signal and produces a wrong identification. In Case 3 and 4, although both low level and high level features yield consistent prediction, they appear to rely on very different information: the former uses more color and texture clues, while the latter seems to use higher level concepts to deal with large variations in pose and view angle. In all cases, the fused feature combines the advantages of both low-level and high-level features and appears to be more reliable than others.

Figure 8 shows a number of typical query images (together with their matched images from the gallery; green = correct) that are considered to have different difficulties for the network under the budgeted stream re-ID setting. Specifically, the query images (without boxes) in the top row are those exited from the first stage of our model, which we denote as "easy". The bottom row shows the "hard examples", which are not correctly identified until the last stage of the network. Generally, the separation between easy and hard by the network conforms to our intuitions.

Figure 8. Visualization of "easy" examples, which are confidently classified at the first stage, and "hard" examples, which never reach sufficient confidence until the very last stage.

6. Conclusion

We introduced a novel deeply supervised approach for person re-ID. Our model fuses embeddings at both lower (higher resolution) and higher (more semantics) layers of the network. This combination yields achieves state-of-the-art results throughout all our benchmark data sets. The availability of multiple embeddings with different computation cost also enables trading off performance for computation for the sake of efficiency. As the first work approaching the re-ID problem on a budget efficiency perspective, we show the solutions empirically on the two resource-constrained scenarios using DaRe of person re-ID.

7. Acknowledgements

The authors are supported in part by the National Science Foundation Grants III-1525919, IIS-1550179, IIS-1618134, S&AS 1724282, and CCF-1740822, the Office of Naval Research Grant N00014-17-1-2175, and the Bill and Melinda Gates Foundation.

References

[1] E. Ahmed, M. Jones, and T. K. Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015.

[2] S. Bai, X. Bai, and Q. Tian. Scalable person re-identification on supervised smoothed manifold. In *CVPR*, 2017.

[3] W. Chen, X. Chen, J. Zhang, and K. Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, 2017.

[4] W. Chen, X. Chen, J. Zhang, and K. Huang. A multi-task deep network for person re-identification. In *AAAI*, 2017.

[5] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, 2016.

[6] M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja. Pedestrian recognition with a learned metric. In *Asian Conference on Computer Vision*, 2010.

[7] S. Ding, L. Lin, G. Wang, and H. Chao. Deep feature learning with relative distance comparison for person re-identification. In *Pattern Recognition*, 2015.

[8] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. In *TPAMI*, 2013.

[9] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, 2010.

[10] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[11] X. Q. Y. Fu, Y.-G. Jiang, and T. X. X. Xue. Multi-scale deep learning architectures for person re-identification. In *ICCV*, 2017.

[12] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *IEEE International Workshop on Performance Evaluation for Tracking and Surveillance*, 2007.

[13] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*, 2008.

[14] A. Grubb and D. Bagnell. Speedboost: Anytime prediction with uniform near-optimality. In *AISTATS*, 2012.

[15] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[17] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[18] M. Hirzer, P. M. Roth, M. Köstinger, and H. Bischof. Relaxed pairwise learned metric for person re-identification. In *ECCV*, 2012.

[19] G. Huang, D. Chen, T. Li, F. Wu, L. van der Maaten, and K. Q. Weinberger. Multi-scale dense convolutional networks for efficient prediction. *arXiv preprint arXiv:1703.09844*, 2017.

[20] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[22] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *CVPR*, 2012.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[24] I. Kviatkovsky, A. Adam, and E. Rivlin. Color invariants for person reidentification. In *TPAMI*, 2013.

[25] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel. Handwritten digit recognition with a back-propagation network. In *NIPS*, 1990.

[26] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *AISTATS*, 2015.

[27] D. Li, X. Chen, Z. Zhang, and K. Huang. Learning deep context-aware features over body and latent parts for person re-identification. In *CVPR*, 2017.

[28] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014.

[29] W. Li, X. Zhu, and S. Gong. Person re-identification by deep joint learning of multi-loss classification. In *IJCAI*, 2017.

[30] X. Li, W.-S. Zheng, X. Wang, T. Xiang, and S. Gong. Multi-scale learning for low-resolution person re-identification. In *ICCV*, 2015.

[31] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith. Learning locally-adaptive decision functions for person verification. In *CVPR*, 2013.

[32] S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, 2015.

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

[34] J. Lin, L. Ren, J. Lu, J. Feng, and J. Zhou. Consistent-aware deep learning for person re-identification in a camera network. In *CVPR*, 2017.

[35] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[36] G. Lisanti, I. Masi, A. D. Bagdanov, and A. Del Bimbo. Person re-identification by iterative re-weighted sparse ranking. In *TPAMI*, 2015.

[37] J. Liu, Z.-J. Zha, Q. Tian, D. Liu, T. Yao, Q. Ling, and T. Mei. Multi-scale triplet cnn for person re-identification. In *ACM Multimedia*, 2016.

[38] Y. Liu, Y. Junjie, and W. Ouyang. Quality aware network for set to set recognition. In *CVPR*, 2017.

[39] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[40] B. Ma, Y. Su, and F. Jurie. Local descriptors encoded by fisher vectors for person re-identification. In *ECCV Workshops and Demonstrations*, 2012.

[41] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato. Hierarchical gaussian descriptor for person re-identification. In *CVPR*, 2016.

[42] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP*, 2009.

[43] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian. Local fisher discriminant analysis for pedestrian re-identification. In *CVPR*, 2013.

[44] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, 2016.

[45] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[46] H. Shi, Y. Yang, X. Zhu, S. Liao, Z. Lei, W. Zheng, and S. Z. Li. Embedding deep metric for person re-identification: A study against large variations. In *ECCV*, 2016.

[47] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian. Pose-driven deep convolutional model for person re-identification. In *ICCV*, 2017.

[48] Y. Sun, L. Zheng, W. Deng, and S. Wang. Svdnet for pedestrian retrieval. In *ICCV*, 2017.

[49] R. R. Varior, M. Haloi, and G. Wang. Gated siamese convolutional neural network architecture for human re-identification. In *ECCV*, 2016.

[50] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *CVPR*, 2016.

[51] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *JMLR*, 2009.

[52] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *CVPR*, 2016.

[53] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, 2015.

[54] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person re-identification using kernel-based metric learning methods. In *ECCV*, 2014.

[55] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Deep metric learning for person re-identification. In *ICPR*, 2014.

[56] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017.

[57] L. Zhao, X. Li, J. Wang, and Y. Zhuang. Deeply-learned part-aligned representations for person re-identification. In *ICCV*, 2017.

[58] R. Zhao, W. Ouyang, and X. Wang. Person re-identification by salience matching. In *ICCV*, 2013.

[59] R. Zhao, W. Ouyang, and X. Wang. Learning mid-level filters for person re-identification. In *CVPR*, 2014.

[60] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian. Mars: A video benchmark for large-scale person re-identification. In *ECCV*, 2016.

[61] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015.

[62] L. Zheng, Y. Yang, and A. G. Hauptmann. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*, 2016.

[63] W.-S. Zheng, S. Gong, and T. Xiang. Reidentification by relative distance comparison. In *TPAMI*, 2013.

[64] Z. Zheng, L. Zheng, and Y. Yang. Pedestrian alignment network for large-scale person re-identification. *arXiv preprint arXiv:1707.00408*, 2017.

[65] Z. Zheng, L. Zheng, and Y. Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *ICCV*, 2017.

[66] Z. Zhong, L. Zheng, D. Cao, and S. Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017.

[67] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.

[68] Z. Zhou, Y. Huang, W. Wang, L. Wang, and T. Tan. See the forest for the trees: Joint spatial and temporal recurrent neural networks for video-based person re-identification. In *CVPR*, 2017.