

Optimizing Speed/Accuracy Trade-Off for Person Re-identification via Knowledge Distillation

Idoia Ruiz^{a,1,*}, Bogdan Raducanu^b, Rakesh Mehta^a, Jaume Amores^a

^a*United Technology Research Centre Ireland, 4th Floor, Penrose Business Center, Penrose Wharf, Cork City, Co. Cork, Republic of Ireland*

^b*Computer Vision Center, Universitat Autònoma de Barcelona, Edifici O, 08193 Bellaterra, Spain.*

Abstract

Finding a person across a camera network plays an important role in video surveillance. For a real-world person re-identification application, in order to guarantee an optimal time response, it is crucial to find the balance between accuracy and speed. We analyse this trade-off, comparing a classical method, that comprises hand-crafted feature description and metric learning, in particular, LOMO and XQDA, to deep learning based techniques, using image classification networks, ResNet and MobileNets. Additionally, we propose and analyse network distillation as a learning strategy to reduce the computational cost of the deep learning approach at test time. We evaluate both methods on the Market-1501 and DukeMTMC-reID large-scale datasets, showing that distillation helps reducing the computational cost at inference time while even increasing the accuracy performance.

Keywords: Person re-identification, Network Distillation, Image Retrieval, Model Compression, Surveillance

*Corresponding author

Email addresses: iruiz@cvc.uab.es (Idoia Ruiz), bogdan@cvc.uab.es (Bogdan Raducanu), mehtar1@utrc.utc.com (Rakesh Mehta), amoresj@utrc.utc.com (Jaume Amores)

¹Present address: Computer Vision Center, Universitat Autònoma de Barcelona, Edifici O, 08193 Bellaterra, Spain.

1. Introduction

Person re-identification refers to the problem of identifying a person of interest across a camera network [1, 2]. This task is specially important in surveillance applications, since nowadays the security systems in public areas such as airports, train stations or crowded city areas, are continuously improving to ensure the population's welfare. In big cities, there are extensive networks of cameras in the most sensitive locations. Identifying an individual requires finding it among all the instances that are present on the collection of images captured by the cameras. These images show usually complex crowded scenes, thus increasing even more the computational complexity of the problem. Therefore, the automation of this task that involves large-scale data becomes essential, as otherwise it would be a laborious task to be performed by humans.

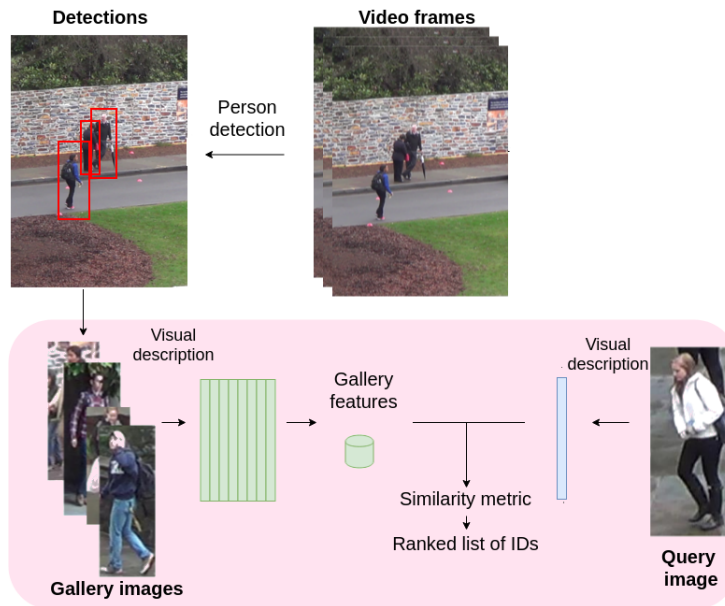


Figure 1: Pipeline of an end-to-end person re-identification system. The pink shaded region delimits the person re-identification module.

The aim of person re-identification is to find a person of interest, also referred as *query*, across a *gallery* of images. The difficulty of this problem lies

in the fact that the images are subject to variations in the point of view, person pose, light conditions and occlusions. Fig. 4 shows examples of gallery images for identities with such kind of variability. Fig. 1 shows the full person re-identification system, including the previous person detection stage. In the person re-identification module, a *query* image of a person of interest is compared against the *gallery*, retrieving the images that correspond to the same identity. To compare them, the system first extracts a feature representation that describes every image, either by using a hand-crafted descriptor or a deep neural network. Usually the features of the *gallery* are previously computed offline and stored, so that at test time we only have to extract the features for the query image. Once the features are extracted, they can be compared with the features of the *gallery* by computing a similarity measure. Finally, all the gallery images are ordered by the degree of similarity, obtaining a ranked list of the most similar images in the gallery to the person of interest [3].

In real scenarios, in order to have a feasible application that is able to work with large-scale datasets in an efficient and effective way, we have to address the problem of optimizing the computational cost of the system at test time, without decreasing drastically its accuracy. For that purpose, we consider both classical and deep learning based person re-identification methods. Although deep learning based techniques outperform significantly hand-crafted methods in terms of accuracy, their drawback is that they require dedicated hardware, *i.e.* GPUs, and big amounts of data for training, which takes usually long periods of time, *i.e.* weeks, in order to be effective.

To make deep learning approaches computationally efficient several works use model compression [4, 5]. The idea behind model compression is to discard *non-informative* weights in the deep networks and perform a fine-tuning to further improve performance. Although these methods make the architecture more efficient in terms of computational complexity, they also result in a drop of the accuracy on the compressed models. This drop is specially prominent when the dataset is large or the number of classes is higher, which is often the case in the person re-identification problem. In contrast, network distillation works

have shown that the smaller or compressed model trained with the support of a much bigger/deeper network is able to achieve very similar accuracy as the deeper network but having a much lower complexity [6, 7, 5]. Therefore, in this work we explore network distillation in the context of efficient person-re-identification.

Contribution. The goal of this work is first, to provide an analysis of the trade-off between accuracy and computational cost at test time in a person re-identification problem, considering the most suitable configuration for a real-world application conditions, and second, to propose an improvement to optimize this trade-off. The contribution of this work is, first, to provide such trade-off analysis on two challenging large-scale person re-identification benchmarks, that are Market-1501 [8] and DukeMTMC-reID [9], and finally to introduce and analyse network distillation [10] for optimizing this trade-off for the deep learning approach. For this purpose, we use ResNet-50 [11], acting as teacher, to transfer the knowledge to a more compact model represented by MobileNet [12], acting as student.

The paper is structured as follows. In Section 2, we review the literature related with person re-identification and distillation. In Section 3, we review the distillation approach. The experimental results are reported in Section 4. Finally, in Section 5, we present our conclusions and provide some guidelines for future work.

2. Related Work

2.1. Person re-identification

Classical methods for person re-identification consider it as two independent problems, that are feature representation and metric learning. For the first task, visual description, popular frameworks like Bag of Words [13] or Fisher Vectors [14] were initially used to encode the local features. Later, the LOMO [15] descriptor was introduced and commonly used on the person re-identification problem [16] [3] [2]. In the exhaustive comparison performed by

Karanam et al. [17], LOMO is the second hand-crafted feature descriptor that performs best across several datasets. The GOG [18] features are superior in terms of accuracy, but computing them is more computationally expensive, since it requires modeling each subregion in which the image is divided, by a set of Gaussian distributions. Indeed, in [18], LOMO features are extracted in 0.016 seconds/image, while GOG features are extracted in 1.34 second/image.

Metric learning consists in learning a distance function that maps from the feature space to a new space in which the vectors of features that correspond to the same identity are close, while those that correspond to different identities are not, being the distance a measure of the similarity. Once learnt, this mapping function is used to measure the similarity between features of the person of interest and the gallery images.

One of the most popular metrics is KISSME [19], that uses the Mahalanobis distance. Later, XQDA[15] was introduced as an extension of KISSME to cross-view metric learning, but doing the mapping function from the feature space to a lower dimensionality space, in which the similarity metric is computed. More recently, [20] proposed a novel metric learning method that address the small sample size problem, which is due to the high dimensionality of the features on person re-identification. According to this metric, the samples of distinct classes are separated with maximum margin while keeping the samples of same class collapsed to a single point, to maximize the separability in terms of Fisher criterion.

Nowadays, deep learning based methods are outperforming hand-crafted techniques. Some approaches used deep learning to compute better image representations, then computing the similarity metric as usual. Considering each identity as a different class, the features are extracted from a classification Convolutional Neural Network (CNN), that is trained on the target dataset. Then the features, that we denote as *deep features*, are the logits, *i.e.* the output of the network before the classification layer. Some works that use this approach are [21], [22] and [23]. A more complex framework is proposed in [24], where using a multi-scale context-aware network, they compute features that contain

both global and local part-based information.

In a different line of work, siamese models were used to learn jointly the representations, computing the similarity between the inputs, that are image pairs. The similarity measure provided by the output of the network, determines whether the input images correspond to the same identity or not. This architecture was first introduced by Bromley et al. [25] for signature verification, where the features for two signature images were extracted and compared by computing the cosine of the angle between the two feature vectors as a measure of the similarity. Similarly, in person re-identification, siamese networks take as an input two person images. This original approach is followed in [26]. Other architectures such as [27] or [28] use the softmax layer to provide a binary output. A siamese framework is also used in [29], where the authors propose an architecture with an enhanced attention mechanism, in order to increase the robustness for cross-view matching. Closely related to siamese networks, triplet networks, which were introduced in [30] for face recognition, take triplets of images as inputs, corresponding only two of them to the same person [31, 32, 33]. Similarly, a quadruplet loss was proposed in [34].

Recent approaches aim at increasing the robustness of person re-identification systems. Some address the problem of domain adaptation, *i.e.* applying to an unseen dataset a model is trained on a set of source domains without any model updating [35, 36]. To this end, image synthesis [37, 38] or domain alignment [39, 40, 41] are used. Other works propose generative approaches for data augmentation. In [42] the synthesized images help learning view-point invariant features by normalizing across a set of generated enhanced pose variations, while in [43] they compose high-quality cross-identities images.

2.2. Network Distillation

Network distillation approaches appeared as a computational effective solution to transfer the knowledge from a large, complex neural network (often called *teacher network*) to a more compact one (referred as *student network*), with significantly less number of parameters. This idea was originally proposed

in [10]. On their approach, the student network was penalized based on a softened version of the teacher network’s output. The student was trained to predict the output of the teacher, as well as the true classification labels. In [6], they proposed an idea to train a student network which is deeper and thinner than the teacher network. They do not only use the outputs, but also the intermediate representations learned by the teacher as hints to improve the training process and final performance of the student. A different approach was proposed in [44], where the knowledge to be transferred from the teacher to the student is obtained from the neurons in the top hidden layer, which preserve as much information as the softened label probabilities, but being more compact.

Network distillation approaches have also been applied recently to the person re-identification problem. In [45], the authors propose using a pair of students to learn collaboratively and teach each other throughout the training process. Each student is trained with two losses: a conventional supervised learning loss, and a mimicry loss that aligns each student’s class posterior with the class probabilities of other students. This way, each student learns better in such peer-teaching scenario than when learning alone. In [46], feature distillation is used to learn identity-related and pose-unrelated representations. They adopt a siamese architecture, consisting each branch of an image encoder/decoder pair, for feature learning with multiple novel discriminators on human poses and identities. The recent work in [47] resembles ours in some aspects, although their scope is semi-supervised and unsupervised person re-identification, in contrast to our fully-supervised formulation. Similarly to us, they consider lightweight models to reduce testing computation as well as network distillation as a strategy of knowledge transfer. However, their distillation approach is not probability based, but similarity based. They propose the Log-Euclidean Similarity Distillation Loss that imitates the pairwise similarity of the teacher instead of using soft labels as we do. They explore a multiple teacher-single student setting and propose an adaptive knowledge aggregator to weight the contributions of the teachers.

3. Reviewing Distillation

Besides improving the performance of the person re-identification pipeline in terms of computational cost at test time, we also aim at maximising the performance of a small network to be as accurate as possible.

As discussed in [10], the simplest way to transfer the knowledge is to use the output of the teacher network as soft targets for the student network, additionally to the hard targets provided by the ground truth. However, when the soft targets have high entropy, they provide more information to learn from. Then, a network that is very confident about its prediction, will generate a probability distribution similar to a Dirac delta function, in which the correct class has a very high probability and the rest of classes have almost zero probability, having a very low entropy and consequently providing less information than a less confident network. While a less confident network will assign higher probabilities to the incorrect classes, as shown graphically in Fig. 2. The intuition behind high entropy distributions help the distillation, is that by learning from the probabilities assigned to incorrect classes, the student network is learning how the teacher model generalizes.

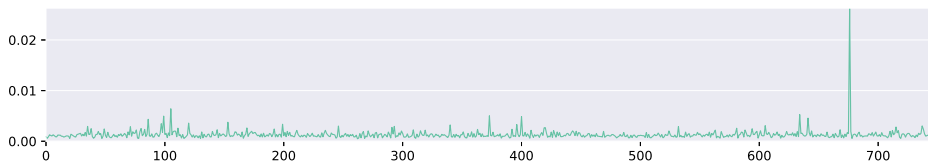


Figure 2: Example of a high entropy probability distribution, generated by the softmax layer of the teacher network for the Market-1501 dataset (751 classes).

Therefore, the authors propose to increase the entropy of the probability distribution generated by the teacher model, *i.e.* the output of the softmax layer, so that when the student network uses that output to learn from it, it can provide more information. In order to maximize the entropy, they propose to increase the *temperature* of this distribution.

The inputs of the softmax layer, that are the *logits*, denoted as z_i , are con-

verted to probabilities p_i by the softmax function, which expression is (1), where T is the temperature, that is a selected constant value in the distillation case, and it is equal to 1 when there is no distillation.

$$p_i = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \quad (1)$$

The knowledge transfer is performed via the loss function of the student model. The loss function for the k -th training example $L_{student_k}$ is defined as (2) and it is the weighted sum of two terms:

$$L_{student_k} = \underbrace{H(p_{teacher}(T = T_0), p_{student}(T = T_0))}_{\text{Distillation term}} + \underbrace{\lambda H(\text{hard targets}, p_{student}(T = 1))}_{\text{Cross-entropy loss}} \quad (2)$$

where $H(p, q)$ denotes the cross-entropy between two probability distributions p and q . The first term is the cross-entropy between the soft targets extracted from the teacher ($p_{teacher}(T = T_0)$), *i.e.* the softened probability distribution of the teacher that is obtained by applying the softmax function (1) to the logits of the teacher divided by a temperature T_0 , and the softened probability distribution of the student ($p_{student}(T = T_0)$) using the same value T_0 as for the teacher. The second term of the loss is the cross-entropy between the *hard targets*, that is the ground truth which has a value equal to 1 assigned to the correct class and 0 to the rest of them, and the probability distribution of the student ($p_{student}(T = 1)$), that is the output of the softmax using a $T = 1$. This second term is the cross-entropy loss function, which minimizes the cross-entropy between the prediction of the network and the ground truth. These two terms are balanced by a regularization parameter λ .

A graphical summary of the process is shown in Fig. 3. In the current framework of person re-identification, once the student network is trained via distillation, it is used to extract the features of the images at test time, to then measure their similarity using the Euclidean distance.

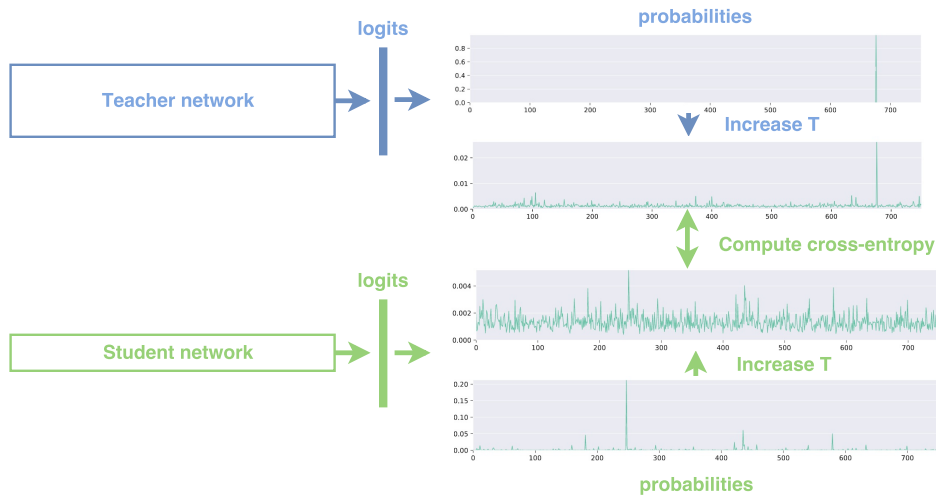


Figure 3: Distillation process. The cross-entropy between the softened distributions generated by the teacher and the student networks is computed in order to minimise it additionally to the cross-entropy with the ground truth.

4. Experiments

4.1. Datasets

In a real world application, there are often several cameras that can capture images of people from different points of view in different illumination conditions and even with occlusions. Thus, we choose datasets that simulate as much as possible a real scenario. Market-1501 [8] or DukeMTMC-reID [9] have these characteristics, providing images taken from 6 cameras in the case of Market-1501 and 8 in the case of DukeMTMC-reID, as shown in Fig. 4, that are captured in outdoor public areas, being also two of the largest-scale public datasets for person re-identification.

Market-1501 provides an average of 14.8 cross-camera ground truths for each query, containing in total 32,668 bounding boxes of 1,501 identities, from which 12,936 bounding boxes with 751 identities belong to the training set. The mean of images per identity is 17.2. All the bounding boxes are of size 128x64.

The DukeMTMC-reID dataset is an extension of the DukeMTMC tracking dataset. The bounding boxes are then extracted from the full frames provided

by the original dataset and therefore, their size is not fixed. It contains 36,441 bounding boxes that belong to 1,404 identities plus 408 distractor identities that only appear in a single camera. Among them, 16,522 bounding boxes with 702 identities are used for the training set. The mean number of images per identity is 20, with a maximum of 426 images for the identity with the largest amount of images.

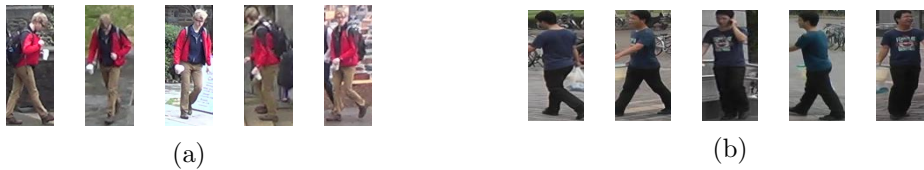


Figure 4: Subset of gallery images that correspond to 2 identities from the (a) DukeMTMC-reID and (b) Market-1501 datasets. Each identity can appear in different cameras and may present different points of view, pose, and illumination conditions.

4.2. Evaluation

In a re-identification task, the *query* is compared to all the *gallery*, computing a similarity metric that is used to rank the *gallery* images sorted by similarity. The rank-1 accuracy gives the probability of getting a true match from the gallery in the first position of the ranking. Similarly, the rank-5 accuracy evaluates whether we find a true match in the five first positions of the ranking. However, since the person of interest may appear many times in the gallery, we need an evaluation metric that also considers finding all the true matches that exist in the gallery, evaluating also the recall. The mean average precision (mAP) is suitable for evaluating on datasets in which an identity appears more than once in the gallery, such as Market-1501 and DukeMTMC-reID.

We also report the computational cost at test time of the algorithms proposed, by providing the time that feature extraction takes per image of a single individual. We do the feature extraction for all the *gallery* and compute the average time per image. We report as a computational cost metric, the number of images the system extracts the features from in a second, for the different considered architectures. Then, we report separately the computational cost for

the metric learning step.

4.3. Implementation details

To analyse the trade-off between accuracy and computational cost at test time, we evaluate both classical and deep learning based approaches. In a real world application, both of them can be considered depending on the scenario.

As a classical approach, we use the LOMO feature description and the XQDA metric learning algorithms [15], as they aim at being effective and computationally efficient. As we discussed in section 2.1, LOMO presents the best trade-off between accuracy and computational cost for all the methods considered in the exhaustive analysis performed in [17].

In a deep learning based approach, as described in section 2.1, the feature representations are extracted from a CNN considering the identities as classes and taking the output from the last layer before the softmax layer as the *deep features*. Our baseline is the one presented at [21] for the Market-1501 dataset, using the ResNet-50 [11] model. Since ResNet-50 might be too large for the datasets we consider, we also explore another smaller networks that can be more efficient and still perform well. In particular, we consider MobileNets [12] as an alternative architecture.

MobileNets are presented as efficient light weight models suitable for mobile applications. The MobileNets architecture can be adapted to particular requirements of the system. In order to decide the network size, two parameters are introduced to control its latency and accuracy: the width multiplier $\alpha \in (0, 1]$ and the resolution multiplier $\rho \in (0, 1]$. The width multiplier can make the model thinner, by multiplying the number of input and output channels on each layer by α . ρ is implicitly selected when determining the input size of the network, that can be 224, 192, 160 and 128. Finally, as the similarity metric to compare the features extracted from the gallery images, we use the Euclidean distance.

Hand-crafted features. To evaluate the LOMO features independently to XQDA, we compare the Euclidean distance, KISSME [19] and XQDA as similarity met-

rics. PCA is commonly applied previously to KISSME in order to reduce the dimensionality of the LOMO features, in our case from 26960 to 200. XQDA allows to select the dimensionality of its subspace. Thus, we also evaluate the performance of LOMO + XQDA depending on the XQDA dimensionality. The maximum value that we consider is the highest one with eigenvalues greater than 1. Following this criteria, we get a maximum dimensionality of 76 for the features extracted from the Market-1501 dataset. Therefore, we consider values of the XQDA dimensionality from 25 to 75. Finally, to evaluate the computational cost, we measure the inference time of the method, running these experiments on a laptop with a CPU Intel Core i5-6300U CPU @ 2.40GHz.

Deep features. Our deep learning based methods are implemented using the TensorFlow library. The training and validation splits used for deep features are the ones provided on the original baselines. For Market-1501, Zheng et al. [21] use a validation split of 1,294 images leaving 11,642 for training. The baseline for DukeMTMC-reID [48] uses the whole set of training images. Finally, to evaluate the computational cost, we measure the inference time, running the experiments on a NVIDIA GTX1070 GPU.

- *ResNet-50* The ResNet-50 network is fine-tuned from the weights pre-trained for ImageNet, considering the person identities as classes. The deep features are then extracted from the last layer before the softmax, which in the ResNet-50 architecture, corresponds to the output of the average pooling layer.

It is worth mentioning that because of the high number of classes in the datasets (751 and 702 identities for the training splits of Market-1501 and DukeMTMC-reID respectively), with few images per class (a mean of 20 for DukeMTMC-reID and 17.2 for Market-1501), it is hard to train the network, since a deep neural network needs a big amount of data to converge properly.

To train ResNet-50, we resize the input images to 224x224 and use horizontal flip for data augmentation. Using Stochastic Gradient Descent

(SGD), we initially set the learning rate to 0.001 with a decay of 0.1 every 20000 steps. Using a batch size of 16 and momentum of 0.9, we train the network for 21 epochs (15000 iterations) for the Market-1501 dataset. For DukeMTMC-reID, the learning rate is initially set to 0.01 and we use a batch size of 32, training it for 29 epoch (15000 iterations).

- *MobileNets* We choose an input size of 128 due to the size of the images of the datasets we use. Market-1501 images have a fixed size of 128x64 while DukeMTMC-reID images size vary. Then, we resize all the images to 128x128, applying horizontal flip for data augmentation. We evaluate the performance for width multipliers of $\alpha = 0.25, 0.5, 0.75, 1.0$, which are the values with ImageNet pre-trained weights being provided. We denote these networks as MobileNet 0.25, 0.5, 0.75 and 1.0 respectively. α also affects the dimensionality of the extracted features from the network, which are the output of the final average pooling. The features are of length 1024, 768, 512 and 256 for values of $\alpha = 1.0, 0.75, 0.5$ and 0.25 respectively.

The training hyperparameters we use, are those that perform best across several experiments. The results on Market-1501 are obtained by using SGD with a batch size of 32, an initial learning rate of 0.01 with a decay of 0.1 every 20000 steps, and momentum of 0.9. We train MobileNet 0.25 for 29 epochs and the rest of MobileNets for 39 epochs. On DukeMTMC-reID, we set the initial learning rate to 0.01 for MobileNet 0.25 and to 0.02 for MobileNet 1.0, training both of them for 39 epochs. For MobileNets 0.5 and 0.75, we use a batch size of 16, a starting learning rate of 0.005 and we train them for 39 epochs.

Network distillation. We propose ResNet-50 as teacher, but also MobileNet 1.0, which has the biggest capacity among the MobileNets configurations. The number of parameters for MobileNets are 4.24M, 2.59M, 1.34M and 0.47M for width multiplier values of 1.0, 0.75, 0.5 and 0.25 respectively, while ResNet-50 has 23.5M of parameters. Since we want an efficient network, the student is the

MobileNet with the smallest width multiplier (MobileNet 0.25). We analyse the effect of the hyperparameters of the distillation, that are the temperature T and the regularization weight λ for the distillation loss. We consider the range of temperatures $1 - 30$, being $T = 1$ the case in which the entropy of the soft targets is not modified and $T = 30$ a case of very high temperature. The highest temperature is selected based on the observed softened probability distribution that is generated by the teacher network for $T = 30$, as it is shown in Fig. 5. In that probability distribution, the difference between the probabilities assigned to the incorrect classes and the one assigned to the correct class is less than a 0.1%. This is due to a very high temperature with which the probability distribution is almost flat (which is the case of maximum entropy). To do the analysis for T in that range, we use intervals of 5, and 1 for the lowest values. For λ , we choose the values 0.0001, 0.001 and 0.01. They have been chosen by analysing the contribution of the loss terms while monitoring the training process, as shown in Fig. 6. When using a value of $\lambda = 0.1$, the cross-entropy loss leads the training and the distillation term barely affects, but we noted from our experiments that it makes the training harder to converge, resulting in a performance drop. Therefore, we do not consider $\lambda = 0.1$ and higher values for our analysis.

For each value of T , we evaluate both the Rank-1 accuracy and mAP with the features extracted from the student network. We try several combinations of the hyperparameters, *i.e.*, learning rate, batch size, number of epochs, etc. However, most of the experiments perform best using the same hyperparameters, *i.e.*, we obtain that the same optimum configuration of parameters for several values of T and λ . Then, all the Rank-1 and mAP values reported in section 5 for each value of T , are those that perform best among all the experiments performed. Most of the distillation experiments use SGD, with an initial learning rate of 0.02 that decays 0.1 every 20000 steps, and a momentum of 0.9, being trained for 39 epochs.

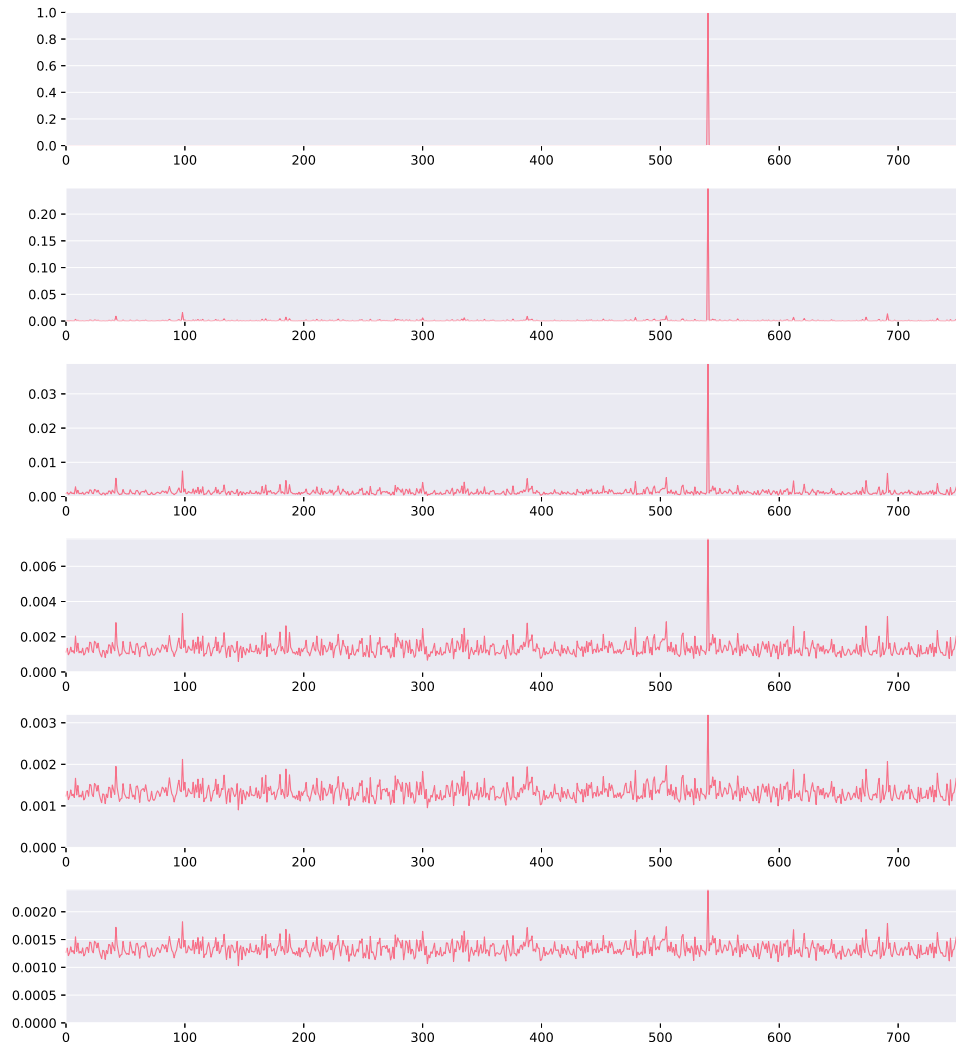


Figure 5: Original and softened probability distributions generated by the teacher network for temperature values of (from top to bottom) $T=1,3,5,10,20,30$.

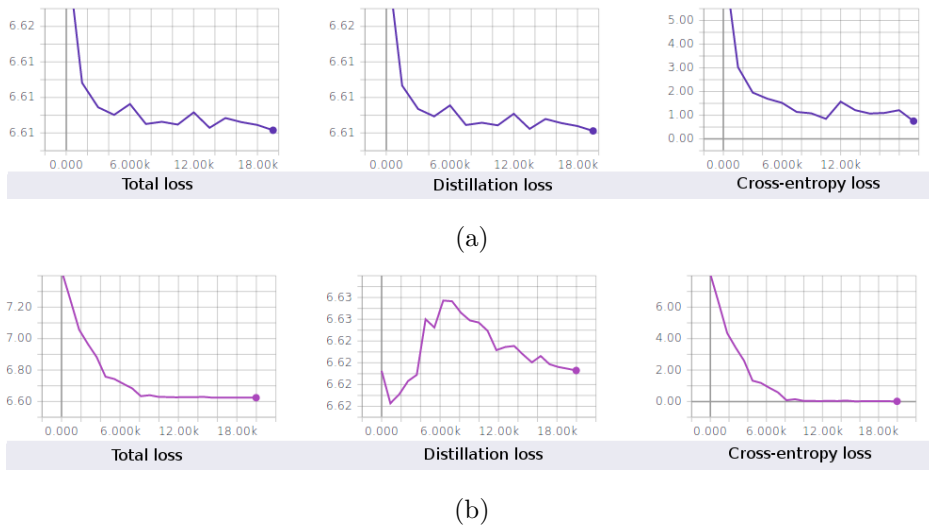


Figure 6: Training loss for the distillation with (a) low ($\lambda = 0.0001$) and (b) high ($\lambda = 0.1$) λ values. The distillation loss leads the training in case (a), while in case (b) it is done by the cross-entropy loss (see equation 2).

5. Results

The performance of the classical approach using LOMO and XQDA is shown in Table 1. We verify that the usage of metric learning algorithms such as KISSME or XQDA significantly improves the performance of hand-crafted features. However, we must consider that in this table, PCA is previously applied in the case of KISSME to reduce the dimensionality of the LOMO features to 200. The dimensionality in the XQDA space is 75, which is considerably smaller. Thus, XQDA performs better than KISSME even with a stronger dimensionality reduction.

However, both XQDA and KISSME require a metric learning step that increases the computational cost. In particular, the XQDA training, *i.e.* finding the projection matrix from the training set samples, takes 892 seconds for Market-1501, whose training set contains 12936 images. Also, comparing a query image against the gallery takes a mean time of 1,951 ms per image. Thus, using XQDA, the system compares the individuals' features at a rate of 0.5 images/s.

Regarding the computational cost for feature extraction with LOMO, the mean CPU time to extract the LOMO features per image is 17.5ms. Then, the system is able to get the descriptors for the images of the individuals at a rate of 57 images/s.

Table 1: LOMO and XQDA performance on Market-1501.

Features	Similarity metric	Rank-1 (%)	mAP (%)
LOMO	Euclidean distance	27.11	8.01
LOMO	KISSME [19]	41.83	19.37
LOMO	XQDA (dimensionality 75)	43.32	22.01

The performance of LOMO+XQDA reported in Table 1 corresponds to the highest dimensionality value for XQDA. We also show the dependency of the performance with the XQDA dimensionality on Fig. 7. The accuracy increases with the dimensionality of XQDA, since more information can be encoded in the feature vector with a higher dimensionality. Although we expect a saturation on the performance from a certain value, we do not reach such value. This is probably because the maximum dimensionality in our case is 75, which is considerably low. It is much lower than the dimensionality of the smallest feature vectors considered in this work that is 256 for MobileNet 0.25.

For the deep features baseline, Zheng et al. [21] get a 72.54% of rank-1 accuracy and 46% mean average precision on the Market-1501 dataset, with deep features extracted from ResNet-50. Following the same strategy, in [48] the baseline results for the DukeMTMC-reID dataset are a 65.22% of rank-1 accuracy and 44.99% of mean average precision.

Fine-tuning the ResNet-50 and MobileNets architectures to the datasets considered, we obtain the performance presented in Table 2. For Market-1501, the middle size MobileNets are the models that perform best, even slightly better than the biggest one and ResNet-50. However, MobileNet 0.25 presents a lower performance. The reason why the middle models perform so well, could be that

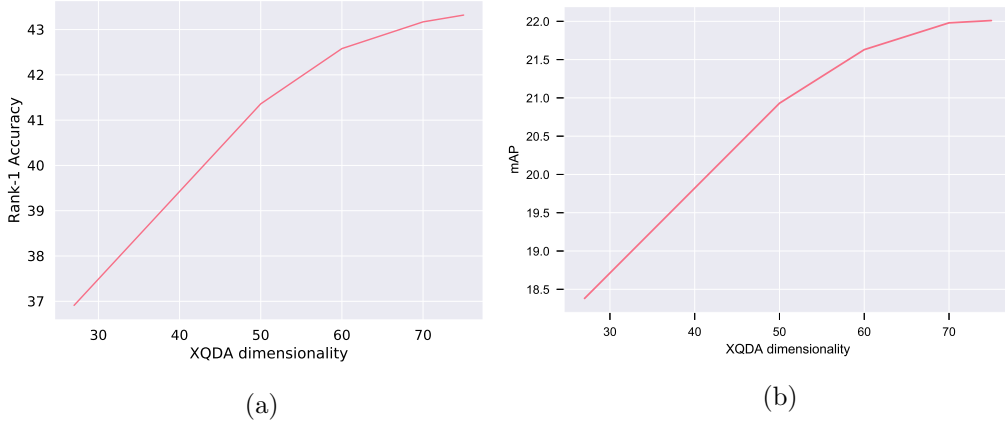


Figure 7: Performance for LOMO + XQDA on Market-1501 depending on the XQDA dimensionality. (a) Rank-1 accuracy and (b) Mean average precision.

all of them have enough capacity to solve the problem. Then, a bigger architecture, such as ResNet-50, would not involve an improvement. Moreover, as mentioned in section 4.3, training the networks on a dataset with a high number of classes and a small number of samples per class is not straightforward. The baseline achieved with ResNet-50 by Zheng et al. [21] suggests that a higher performance could be achieved for this network.

For the DukeMTMC-reID dataset, MobileNets do not perform as good as they do for Market-1501. The reason might be that this dataset is more challenging, and requires a higher capacity of the network to perform a good description of the identities. Since the size of the bounding boxes vary and all of them have to be resized to 128x128, losing thereby the aspect ratio, the input images have a higher variability.

We perform the network distillation experiments using pre-trained ResNet-50 and MobileNet 1.0 networks as teachers, whose performance is reported in Table 2. We show in Fig. 8 and Fig. 9 the Rank-1 accuracy and mAP dependency with the temperature in the distillation, for the Market-1501 and DukeMTMC-reID datasets respectively. The performance of the teacher and the student trained independently is also drawn in the previous figures to pro-

Table 2: Rank-1 accuracy, mean Average Precision (mAP) and computational cost of the inference for the deep features from the ResNet-50 and MobileNet architectures trained on the Market-1501 and DukeMTMC-reID datasets.

Market-1501	Rank-1 (%)	mAP (%)	# images/s
ResNet-50	64.46	38.95	128
MobileNet 0.25	59.74	34.13	613
MobileNet 0.5	68.11	41.52	607
MobileNet 0.75	67.34	40.44	574
MobileNet 1.0	67.37	39.54	545
DukeMTMC-reID	Rank-1 (%)	mAP (%)	# images/s
ResNet-50	67.1	44.59	128
MobileNet 0.25	49.69	28.67	613
MobileNet 0.5	54.62	32.17	607
MobileNet 0.75	57.32	34.69	574
MobileNet 1.0	57.41	34.86	545

vide the comparison with the baseline without distillation. All the experiments improve significantly the performance of the student, and even the performance of the teacher for low temperatures. The only case in which the student does not outperform the teacher is for the DukeMTMC-reID dataset for the distillation from ResNet-50 (Fig. 9 (a,b)). However, in this case, the difference of performance between the teacher and the student is higher than for the other experiments.

For a fixed value of λ , there is always a peak of performance in $T = 3$. The worst performance across all the values of the temperature T , is for $T=1$, which corresponds to the case in which the temperature is not increased, *i.e.* the original logits from the teacher models are used. This demonstrates the importance of raising the temperature to produce suitable soft targets. Also, from a certain value of T , the performance gets saturated, probably because the probabilities are already very softened and they do not change significantly

for those values of T , as Fig. 5 (e,f) shows for the values of $T = 20, 30$. The differences of probabilities among both distributions are less than a 0.1%.

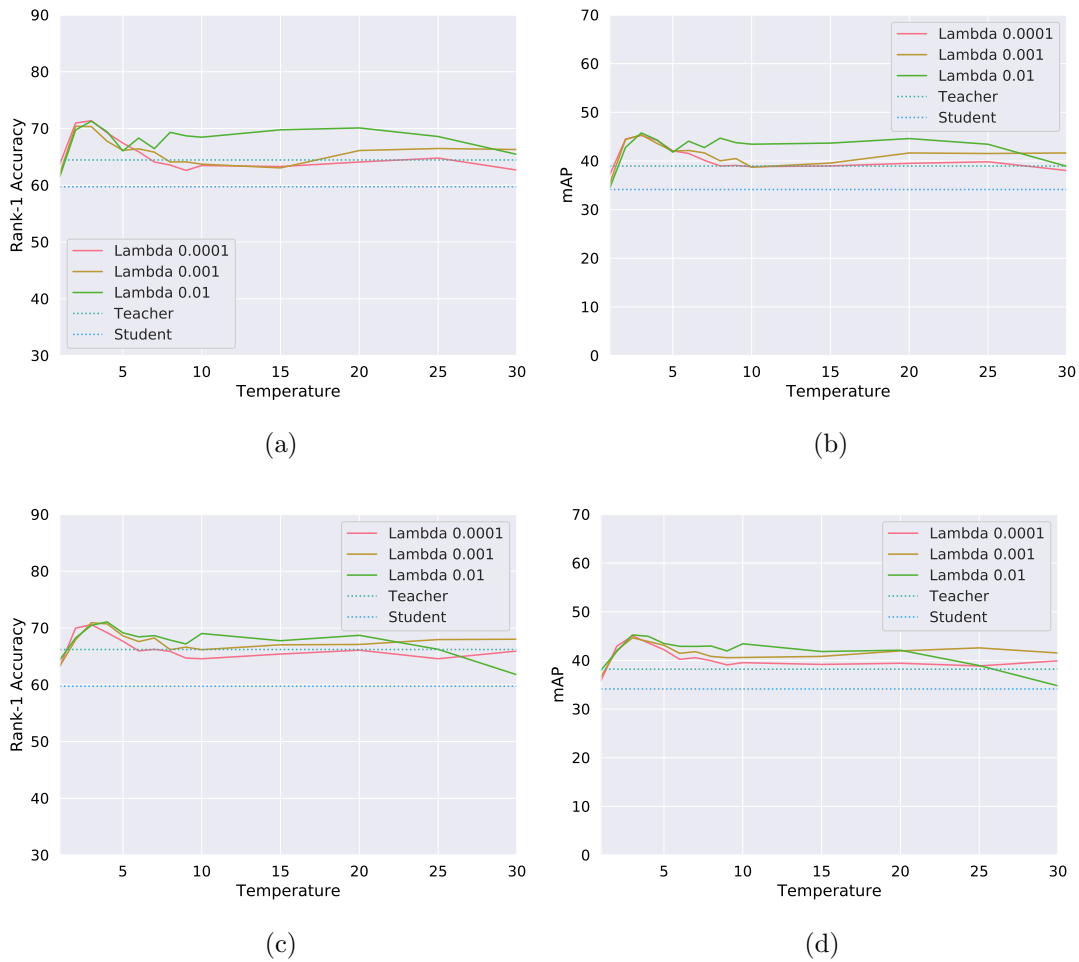
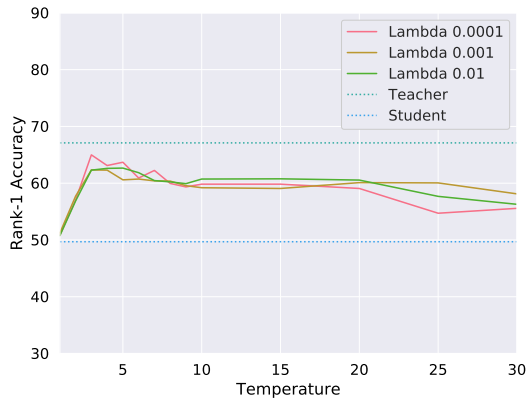
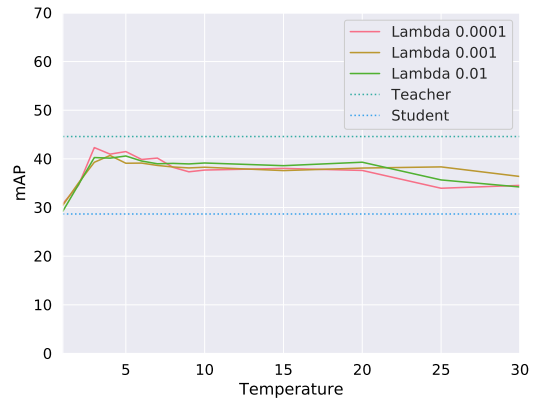


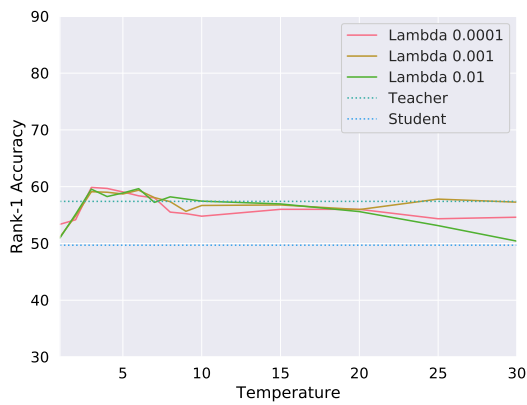
Figure 8: Distillation performance on Market-1501. (a) Rank-1 accuracy and (b) Mean average precision for student model MobileNet 0.25 with teacher model ResNet-50. (c) Rank-1 accuracy and (d) Mean average precision for student model MobileNet 0.25 with teacher model MobileNet 1.0. Best viewed in colour.



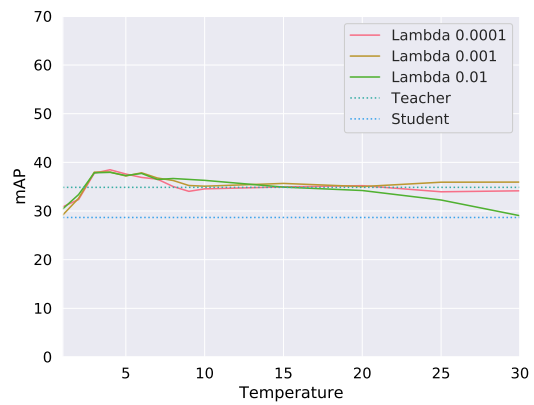
(a)



(b)



(c)



(d)

Figure 9: Distillation performance on DukeMTMC-reID. (a) Rank-1 accuracy and (b) Mean average precision for student model MobileNet 0.25 with teacher model ResNet-50. (c) Rank-1 accuracy and (d) Mean average precision for student model MobileNet 0.25 with teacher model MobileNet 1.0. Best viewed in colour.

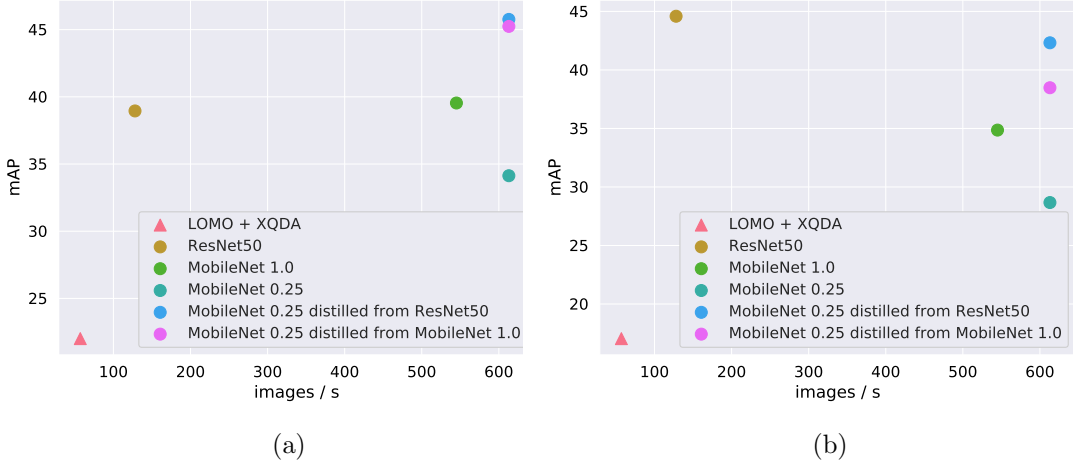


Figure 10: Trade-off between the mean average precision (mAP) and the feature extraction time for the proposed methods on the (a) Market-1501 and (b) DukeMTMC-reID datasets. Note that the feature extraction time for LOMO is measured as CPU time while the deep features experiments are run on a GPU. Best viewed in colour.

In Table 3, we compare our configuration with the highest performance for network distillation against the state-of-the-art. Although the accuracy achieved is not better than the state-of-the-art, our method is specifically designed to be efficient, which can compromise the accuracy.

Finally, to summarise all the considered methods, we show in Fig. 10 and in Table 4, the trade-off between computational cost and accuracy. In this table, we compare the performance of the classical approach (LOMO+XQDA), the deep features extracted from the MobileNets architectures trained with the cross-entropy loss as well as the deep features extracted from MobileNet 0.25 being distilled from the MobileNet 1.0 and ResNet-50 models, whose performance is reported in the table. On the Market-1501 dataset, we compute the LOMO features and then apply XQDA with dimensionality 75, while the results for the DukeMTMC-reID dataset is from [48].

Table 3: Rank-1 accuracy and mean Average Precision (mAP) for network distillation, taking MobileNet ($\alpha = 0.25$) as the student network, and MobileNet ($\alpha = 1.0$) and ResNet-50 as the teachers, compared against the state-of-the-art on the Market-1501 and DukeMTMC-reID benchmarks.

Market-1501	Rank-1 (%)	mAP (%)
MobileNet 0.25 distilled from ResNet-50	71.29	45.76
MobileNet 0.25 distilled from MobileNet 1.0	70.46	45.24
P2S [49]	70.72	44.27
CADL [50]	73.84	47.11
MSCAN Fusion [24]	80.31	57.53
SVDNet [51]	82.3	62.1
ACRN [52]	83.61	62.60
DML [45]	89.34	70.51
FD-GAN [46]	90.5	77.7
DukeMTMC-reID	Rank-1 (%)	mAP (%)
MobileNet 0.25 distilled from ResNet-50	64.99	42.32
MobileNet 0.25 distilled from MobileNet 1.0	59.69	38.48
Dataset baseline with ResNet-50 [48]	65.22	44.99
ACRN [52]	72.58	51.96
SVDNet [51]	76.7	56.8
FD-GAN [46]	80.0	64.5

Note that LOMO is measured in CPU time, while all the deep features methods are measured in GPU time. Therefore, the comparison for computational cost is not strictly fair. In terms of accuracy, the LOMO+XQDA accuracy is with a large margin the lowest, as expected for a hand-crafted method. Then, this kind of method would be suitable only for an application in which either a GPU, or a large amount of annotated data, is not available. The results show that distillation improves effectively the performance of efficient networks, providing the best accuracy among all the considered methods, as well as the

lowest inference time. It is also worth mentioning the gap of computational cost between ResNet-50 and MobileNets, while their performance in terms of accuracy is very similar. Then, it is important to choose a suitable architecture for the problem we want to solve. For the Market-1501 dataset, a network of the size of MobileNet can describe the features of the identities effectively. In the case of DukeMTMC-reID, ResNet-50 performs much better.

Table 4: Evaluation of the trade-off between Rank-1 accuracy, mean Average Precision (mAP) and computational time on the Market-1501 and DukeMTMC-reID datasets. *d.f.* stands for "distilled from".

Market-1501	Rank-1 (%)	mAP (%)	# images/s
LOMO + XQDA	43.32	22.01	57
ResNet-50	64.46	38.95	128
MobileNet 1.0 independent	67.37	39.54	545
MobileNet 0.25 independent	59.74	34.13	613
MobileNet 0.25 d.f. ResNet-50	71.29	45.76	613
MobileNet 0.25 d.f. MobileNet 1.0	70.46	45.24	613
DukeMTMC-reID	Rank-1 (%)	mAP (%)	# images/s
LOMO + XQDA [48]	30.75	17.04	57
ResNet-50	67.1	44.59	128
MobileNet 1.0 independent	57.41	34.86	545
MobileNet 0.25 independent	49.69	28.67	613
MobileNet 0.25 d.f. ResNet-50	64.99	42.32	613
MobileNet 0.25 d.f. MobileNet 1.0	59.69	38.48	613

6. Conclusions and Future Work

In this work, we have evaluated the trade-off between accuracy and computational cost for LOMO and XQDA as a classical approach, also for features extracted from the ResNet-50 and MobileNets networks, as a deep learning

based method. This evaluation was performed on large-scale datasets, aiming to simulate the scenario of a real-world application. In such scenario, the kind of images on which the re-identification is performed, frequently show crowded scenes, which justifies the necessity of having an efficient system that is able to identify as many individuals as possible in the shortest time.

We showed that using features from CNN outperforms by a large margin the accuracy achieved with a classical approach and it is also much faster, when using a GPU. However, this requirement as well as the large amount of annotated data that a network needs to be trained are the drawbacks to consider. Both ResNet-50 and MobileNets achieve a good performance being the second one 4 times faster at test time. Additionally, we proposed network distillation for improving the performance of MobileNets at test time, demonstrating its effectiveness. The student MobileNets networks even outperformed the teacher ResNet-50 model, achieving an accuracy that could not be achieved by training the student independently.

There are still research lines to explore for the the deep learning case applied to a real scenario. The problem of domain adaptation is still open. It refers to the situation when networks trained with labeled datasets can still perform well with new data recorded in different conditions. Also, the retrieval module in the person re-identification pipeline is a bottleneck since a brute-force search is needed in order to compare the person of interest against all the gallery. To solve this, some clustering and indexing approaches have been proposed to reduce the computational cost at test time too, but there is still room for improvement.

Acknowledgements

This work was supported by the SURVANT project which has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 720417. Bogdan Raducanu is supported by Grant No. TIN2016-79717-R, funded by MINECO, Spain.

References

References

- [1] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Q. Tian, Person Re-identification in the Wild, in: Proc. of IEEE International Conference on Computer Vision, 1367–1376, 2017.
- [2] R. Panda, A. Bhuiyan, V. Murino, A. K. Roy-Chowdhury, Unsupervised Adaptive Re-Identification in Open World Dynamic Camera Networks, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu (HI), 1377–1386, 2017.
- [3] Z. Zhong, L. Zheng, D. Cao, S. Li, Re-Ranking Person Re-Identification With k-Reciprocal Encoding, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu (HI), 1318–1327, 2017.
- [4] C. Bucilu, R. Caruana, A. Niculescu-Mizil, Model compression, in: Proc. of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, Philadelphia (PA), 535–541, 2006.
- [5] J. Ba, R. Caruana, Do deep nets really need to be deep?, in: Proc. of Proc. of Neural Information Processing Systems, Montreal (Quebec), Canada, 2654–2662, 2014.
- [6] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, Y. Bengio, Fitnets: Hints for thin deep nets, in: Proc. of International Conference on Learning and Representation, San Diego (CA), 1–10, 2015.
- [7] S. Zagoruyko, N. Komodakis, Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer, in: Proc. of International Conference on Representation and Learning, Toulon, France, 1–11, 2017.
- [8] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, Q. Tian, Scalable person re-identification: A benchmark, in: Proc. of IEEE International Conference on Computer Vision, Boston (MA), 1116–1124, 2015.

- [9] E. Ristani, F. Solera, R. Zou, R. Cucchiara, C. Tomasi, Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking, in: Proc. of European Conference on Computer Vision Workshops, Amsterdam, The Netherlands, 1–18, 2016.
- [10] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 .
- [11] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas (NV), 770–778, 2016.
- [12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861 .
- [13] W.-S. Zheng, S. Gong, T. Xiang, Associating Groups of People, in: Proc. of British Machine Vision Conference, Cardiff, UK, 1–11, 2009.
- [14] F. Perronnin, Y. Liu, J. Sánchez, H. Poirier, Large-scale image retrieval with compressed fisher vectors, in: Proc. of IEEE Computer Vision and Pattern Recognition, San Francisco (CA), 3384–3391, 2010.
- [15] S. Liao, Y. Hu, X. Zhu, S. Z. Li, Person re-identification by local maximal occurrence representation and metric learning, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Boston (MA), 2197–2206, 2015.
- [16] R. R. Varior, B. Shuai, J. Lu, D. Xu, G. Wang, A siamese long short-term memory architecture for human re-identification, in: Proc. of European Conference on Computer Vision, Amsterdam, The Netherlands, 135–153, 2016.
- [17] S. Karanam, M. Gou, Z. Wu, A. Rates-Borras, O. Camps, R. J. Radke, A Systematic Evaluation and Benchmark for Person Re-Identification: Fea-

- tures, Metrics, and Datasets, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41 (3) (2018) 523–536.
- [18] T. Matsukawa, T. Okabe, E. Suzuki, Y. Sato, Hierarchical Gaussian Descriptor for Person Re-Identification, in: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas (NV), 1363–1372, 2016.
- [19] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, H. Bischof, Large scale metric learning from equivalence constraints, in: *Proc. of IEEE Computer Vision and Pattern Recognition*, Providence (RI), 2288–2295, 2012.
- [20] T. F. Ali, S. Chaudhuri, Maximum Margin Metric Learning Over Discriminative Nullspace for Person Re-identification, in: *Proc. of European Conference on Computer Vision*, Munich, Germany, 123–141, 2018.
- [21] L. Zheng, Y. Yang, A. G. Hauptmann, Person re-identification: Past, present and future, *arXiv preprint arXiv:1610.02984* .
- [22] T. Xiao, H. Li, W. Ouyang, X. Wang, Learning deep feature representations with domain guided dropout for person re-identification, in: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas (NV), 1249–1258, 2016.
- [23] S. Bak, P. Carr, One-Shot Metric Learning for Person Re-Identification, in: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu (HI), 2990–2999, 2017.
- [24] D. Li, X. Chen, Z. Zhang, K. Huang, Learning Deep Context-Aware Features Over Body and Latent Parts for Person Re-Identification, in: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu (HI), 384–393, 2017.
- [25] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a "siamese" time delay neural network, in: *Proc. of Neural Information Processing Systems*, 737–744, 1994.

- [26] D. Yi, Z. Lei, S. Liao, S. Z. Li, Deep metric learning for person re-identification, in: Proc. of International Conference on Pattern Recognition, Stockholm, Sweden, 34–39, 2014.
- [27] W. Li, R. Zhao, T. Xiao, X. Wang, DeepReID: Deep Filter Pairing Neural Network for Person Re-identification, in: Proc. of IEEE Computer Vision and Pattern Recognition, Columbus (OH), 1–8, 2014.
- [28] E. Ahmed, M. Jones, T. K. Marks, An improved deep learning architecture for person re-identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Boston (MA), 3908–3916, 2015.
- [29] M. Zheng, S. Karanam, Z. Wu, R. J. Radke, Re-Identification with Consistent Attentive Siamese Networks, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 5735–5744, 2019.
- [30] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Boston (MA), 815–823, 2015.
- [31] D. Cheng, Y. Gong, S. Zhou, J. Wang, N. Zheng, Person re-identification by multi-channel parts-based cnn with improved triplet loss function, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas (NV), 1335–1344, 2016.
- [32] Y. Zhang, Q. Zhong, L. Ma, D. Xie, S. Pu, Learning Incremental Triplet Margin for Person Re-identification, in: Proc. of Association for the Advancement of Artificial Intelligence, Honolulu (HI), 1–8, 2019.
- [33] F. Zheng, C. Deng, X. Sun, X. Jiang, X. Guo, Z. Yu, F. Huang, R. Ji, Pyramidal Person Re-Identification via Multi-Loss Dynamic Training, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 8514–8522, 2019.
- [34] W. Chen, X. Chen, J. Zhang, K. Huang, Beyond Triplet Loss: A Deep Quadruplet Network for Person Re-Identification, in: Proc. of IEEE Con-

- ference on Computer Vision and Pattern Recognition, Honolulu (HI), 403–422, 2017.
- [35] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, T. M. Hospedales, Generalizable Person Re-identification by Domain-Invariant Mapping Network, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 719–728, 2019.
- [36] J. Liu, Z.-J. Zha, D. Chen, R. Hong, M. Wang, Adaptive Transfer Network for Cross-Domain Person Re-Identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 7202–7211, 2019.
- [37] W. Deng, L. Zheng, G. Kang, Y. Yang, Q. Ye, J. Jiao, Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person reidentification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City (UT), 994–1003, 2018.
- [38] Z. Zhong, L. Zheng, S. Li, Y. Yang, Generalizing a person retrieval model hetero-and homogeneously, in: Proc. of European Conference on Computer Vision, Munich, Germany, 176–192, 2018.
- [39] S. Lin, H. Li, C.-T. Li, A. C. Kot, Multi-task mid-level feature alignment network for unsupervised cross-dataset person re-identification, in: Proc. of British Machine Vision Conference, Newcastle, UK, 1–13, 2018.
- [40] J. Wang, X. Zhu, S. Gong, W. Li, Transferable joint attribute-identity deep learning for unsupervised person re-identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City (UT), 2275–2284, 2018.
- [41] L. Wei, S. Zhang, W. Gao, Q. Tian, Person Transfer GAN to Bridge Domain Gap for Person Re-Identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City (UT), 79–88, 2018.

- [42] X. Qian, Y. Fu, T. Xiang, W. Wang, J. Qiu, Y. Wu, Y.-G. Jiang, X. Xue, Pose-Normalized Image Generation for Person Re-identification, in: Proc. of European Conference on Computer Vision, Munich, Germany, 650–667, 2018.
- [43] Z. Zheng, X. Yang, Z. Yu, L. Zheng, Y. Yang, J. Kautz, Joint Discriminative and Generative Learning for Person Re-identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 2138–2147, 2019.
- [44] P. Luo, Z. Zhu, Z. Liu, X. Wang, X. Tang, Face Model Compression by Distilling Knowledge from Neurons, in: Proc. of Association for the Advancement of Artificial Intelligence, Phoenix (AZ), 3560–3566, 2016.
- [45] Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, Deep Mutual Learning, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City (UT), 4320–4328, 2018.
- [46] Y. Ge, Z. Li, H. Zhao, G. Yin, S. Yi, X. Wang, H. Li, FD-GAN: Pose-guided Feature Distilling GAN for Robust Person Re-identification, in: Proc. of Neural Information Processing Systems, Montreal, Canada, 1–12, 2018.
- [47] A. Wu, W.-S. Zheng, X. Guo, J.-H. Lai, Distilled Person Re-identification: Towards a More Scalable System, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach (CA), 1187–1196, 2019.
- [48] Z. Zheng, L. Zheng, Y. Yang, Unlabeled Samples Generated by GAN Improve the Person Re-identification Baseline in vitro, in: Proc. of IEEE International Conference on Computer Vision, Venice, Italy, 3754–3762, 2017.
- [49] S. Zhou, J. Wang, J. Wang, Y. Gong, N. Zheng, Point to Set Similarity Based Deep Feature Learning for Person Re-Identification, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu (HI), 5028–5037, 2017.

- [50] J. Lin, L. Ren, J. Lu, J. Feng, J. Zhou, Consistent-Aware Deep Learning for Person Re-Identification in a Camera Network, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 3396–3405, 2017.
- [51] Y. Sun, L. Zheng, W. Deng, S. Wang, Svdnet for pedestrian retrieval, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu (HI), 3800–3808, 2017.
- [52] A. Schumann, R. Stiefelhagen, Person Re-identification by Deep Learning Attribute-Complementary Information, in: Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu (HI), 1435–1443, 2017.